

Gesture-based Exploration of 3D Medical Images in Virtual Reality

by

Andrey TITOV

MANUSCRIPT-BASED THESIS PRESENTED TO ÉCOLE DE
TECHNOLOGIE SUPÉRIEURE
IN PARTIAL FULFILLMENT FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
Ph.D.

MONTREAL, FEBRUARY 27, 2026

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC



Andrey Titov, 2026



This Creative Commons license allows readers to download this work and share it with others as long as the author is credited. The content of this work cannot be modified in any way or used commercially.

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED
BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Simon Drouin, Thesis supervisor
Department of Software and IT Engineering, École de Technologie Supérieure

Ms. Marta Kersten-Oertel, Thesis Co-Supervisor
Gina Cody School of Engineering and Computer Science, Concordia University

Ms. Rita Noumeir, Chair, Board of Examiners
Department of Electrical Engineering, École de Technologie Supérieure

Mr. Michael McGuffin, Member of the Jury
Department of Software and IT Engineering, École de Technologie Supérieure

Mr. François Rheault, External Independent Examiner
Département d'informatique, Université de Sherbrooke

THIS THESIS WAS PRESENTED AND DEFENDED
IN THE PRESENCE OF A BOARD OF EXAMINERS AND THE PUBLIC
ON FEBRUARY 20, 2026
AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my extraordinary mentors for making this rewarding journey possible. I warmly thank my primary PhD supervisor, Prof. Simon Drouin, for his continuous guidance, patience, and for consistently pushing me to produce my best work over the past five years. I am equally grateful to my co-supervisor, Prof. Marta Kersten-Oertel. Her invaluable insights and unwavering support, from believing in me at the beginning of my Master's to the completion of this doctoral degree, have been fundamental to my success.

I extend my sincere thanks to my examination committee, Prof. Rita Noumeir, Prof. Michael McGuffin, and Prof. François Rheault, for their time, engaging discussions, and constructive feedback. I am also grateful to Prof. Catherine Laporte, whose invaluable advice as a doctoral exam jury member significantly strengthened this thesis.

I am also thankful to all my colleagues from both the Laboratoire de Visualisation Interactive and the Applied Perception Lab, including Samuelle St-Onge, Étienne Léger, Alejandro Olivares, Naomi Catwell, and Liam O'Connor, for their help and insights during this project. I would especially like to thank Tina N. H. Nantenaina for our collaboration and for coordinating the recruitment of neurosurgeons for our user study.

I gratefully acknowledge the financial support that made this research possible. This work was funded by the Fonds de recherche du Québec - Nature et technologies (FRQNT; B2X Doctoral Scholarship, application no. 334501) and the internal ÉTS doctoral scholarship (2022-2023).

Finally, my deepest gratitude goes to my family, specifically my parents, Anatoly and Larissa, and my sister, Tatiana. Your unwavering support and encouragement over all these years made this milestone possible. I also owe a special thank you to my girlfriend, Marzie, for her immense patience, her companionship, and for being my biggest cheerleader during the most challenging parts of my PhD.

Exploration gestuelle d'images médicales 3D en réalité virtuelle

Andrey TITOV

RÉSUMÉ

La réalité virtuelle (RV) est une technologie qui permet une visualisation plus immersive et de nouvelles formes d'interaction que les postes informatiques classiques. Dans cette thèse, nous étudions comment la RV peut améliorer la visualisation de l'anatomie humaine en imagerie médicale. Le travail se concentre sur deux problèmes de visualisation anatomique, la perception de la profondeur et l'auto-occlusion, et examine comment la RV peut aider à dépasser les limites actuelles. La perception de la profondeur est étudiée lors de la visualisation de données angiographiques cérébrales représentant les vaisseaux sanguins du cerveau, ainsi que de volumes non segmentés. L'auto-occlusion apparaît lorsque l'on visualise des régions du corps densément organisées où les couches externes masquent les structures internes. Pour traiter ces deux problèmes, nous proposons des méthodes gestuelles interactives implémentées en RV.

Pour la visualisation angiographique, plusieurs techniques de visualisation de la profondeur ont été développées afin d'améliorer la lisibilité des données. Notre premier objectif est de déterminer comment des techniques initialement conçues pour un écran 2D se transposent en RV et comment les indices de profondeur fournis par les casques affectent la performance des usagers. À l'aide d'une étude utilisateur, nous avons constaté que les techniques évaluées présentent peu de différences lorsqu'elles sont mesurées avec des métriques courantes, ce qui suggère que la RV fournit déjà des indices de profondeur puissants. Toutefois, des métriques spécifiques à l'étude montrent que les variantes plus interactives, où les mouvements de la main modifient la visualisation, produisent de meilleurs résultats que les variantes moins interactives.

La visualisation d'ensembles de données présentant une auto-occlusion importante a été au centre de deux publications, dans lesquelles nous étudions la gestion interactive de l'occlusion en RV afin d'améliorer l'exploration des données. Dans la première, nous proposons Contextual Ambient Occlusion, une nouvelle implémentation d'occlusion ambiante volumétrique permettant un clipping en temps réel tout en recalculant l'occlusion à chaque image sans sacrifier la qualité visuelle. L'algorithme améliore la perception de la profondeur en exploitant des indices d'ombrage et clarifie la structure sous-jacente. Dans la seconde publication, nous introduisons AnatomyCarve, une technique interactive de découpage sensible aux segments qui facilite l'exploration de segments anatomiques préétiquetés. AnatomyCarve met en œuvre le principe de la dissection par étapes, en retirant couche par couche les tissus externes pour révéler l'anatomie interne et produire des visualisations proches des illustrations de manuels d'anatomie. Cette technique a été jugée facile à utiliser et utile pour la planification préopératoire par des neurochirurgiens et des résidents.

Nous montrons que la RV est une technologie prometteuse pour la visualisation médicale, où les indices de profondeur existants apportent déjà un soutien perceptif solide, mais où une interactivité accrue peut encore renforcer cette perception. Nos résultats indiquent en outre que le découpage interactif est un moyen efficace de visualiser des ensembles de données anatomiques

VIII

auto-occlus en permettant une inspection claire de structures segmentées et non segmentées. Globalement, la thèse illustre comment la RV rapproche fidélité visuelle et interaction intuitive et ouvre la voie à une exploration plus efficace de données anatomiques immersives.

Mots-clés: Imagerie médicale, réalité virtuelle, perception de la profondeur, gestion de l'occlusion, interactivité, interaction human-machine

Gesture-based Exploration of 3D Medical Images in Virtual Reality

Andrey TITOV

ABSTRACT

Virtual reality (VR) is a technology that enables more immersive visualization and new interaction possibilities compared to a standard desktop computer setup. In this thesis, we explore how VR can be used in the field of medical imaging to improve the visualization of human anatomy. This work focuses on two specific problems in anatomical visualization. For each problem, we examine how VR may help address current limitations or enhance the visualization process. The first problem is depth perception when visualizing cerebral angiographic data, which represents blood vessels inside the brain, or unsegmented data. The second problem is self-occlusion, which occurs when visualizing regions of the human body that are densely packed with organs and tissues, where outer layers of anatomy hide inner regions. We propose gesture-based interactive methods implemented in VR to tackle these two problems.

For angiographic visualization, multiple depth visualization techniques have been developed to improve the readability of such data. Our first goal is to determine how techniques originally designed for 2D screens translate to VR, and how the depth cues provided by VR headsets affect them. We found that the evaluated depth perception techniques showed very few differences when measured with commonly used metrics, which suggests that VR already provides powerful depth cues. However, using study-specific metrics, we observed that variants with greater interactivity, where hand movements modified the visualization, performed better than less interactive variants.

The visualization of datasets with substantial self-occlusion has been the focus of two publications. In both, we study interactive occlusion management in VR to improve data exploration. In the first publication, we propose Contextual Ambient Occlusion, a new implementation of an existing high-quality ambient occlusion technique. This method makes it possible to achieve real-time clipping with dynamically calculated volumetric ambient occlusion at every frame without sacrificing visual quality. The proposed algorithm improves depth perception in volume visualization by leveraging shading cues, thereby facilitating a clearer understanding of the underlying structure. In the second publication, we introduce AnatomyCarve, an interactive segment-aware clipping technique that facilitates exploration of specific segments, understood as pre-labeled anatomical structures obtained by image segmentation, in VR. AnatomyCarve employs the idea of stepwise dissection, making it possible to remove outer tissues layer by layer and reveal inner anatomy. It supports the creation of visualizations similar to anatomical book illustrations and shows high usability and positive ratings for preoperative planning from neurosurgeons and neurosurgical residents.

Our work demonstrates that VR is a promising technology for medical visualization. Our findings indicate that in VR, existing depth cues provide strong perceptual support, but that enhancing interactivity may further improve depth perception. Our results also show that interactive clipping in VR is an effective way to visualize anatomical datasets with strong

self-occlusion, enabling clear inspection of both segmented and unsegmented structures through precise control of clipping planes. Overall, this thesis highlights how VR can bridge the gap between visual fidelity and intuitive interaction in medical imaging, offering new pathways toward more effective and immersive exploration of complex anatomical data.

Keywords: Medical imaging, virtual reality, depth perception, occlusion management, interactivity, human-computer interaction

TABLE OF CONTENTS

	Page
INTRODUCTION	1
0.1 Depth Perception of Cerebrovascular Volumes	2
0.2 Occlusion Management in Medical Volumes	2
0.3 Context	3
0.4 Virtual Reality	4
0.5 Research scope and methodology	5
0.6 Original contributions	7
0.7 Thesis organization	8
 CHAPTER 1 BACKGROUND AND LITERATURE REVIEW	 9
1.1 Creation of Medical Images	9
1.1.1 X-Ray	9
1.1.2 Computed Tomography	12
1.1.3 Computed Tomography Angiography	13
1.1.4 Magnetic Resonance Imaging	14
1.2 Volume Rendering	16
1.2.1 Ray Setup	19
1.2.2 Sampling	20
1.2.3 Classification	22
1.2.4 Compositing	24
1.3 Human Visual Perception	25
1.4 Virtual reality	28
1.4.1 History of Virtual Reality	30
1.4.2 Modern Virtual Reality	31
1.5 Perceptual improvements in medical visualization	33
1.5.1 Depth Perception	34
1.5.2 Occlusion Management	41
1.5.2.1 Shading-Based Techniques	41
1.5.2.2 Slicing/Clipping-Based Interaction Techniques	41
1.5.2.3 Deformation-based interaction techniques	44
 CHAPTER 2 OBJECTIVES AND METHODOLOGICAL APPROACH	 51
 CHAPTER 3 THE EFFECT OF INTERACTIVE CUES ON THE PERCEPTION OF ANGIOGRAPHIC VOLUMES IN VIRTUAL REALITY	 53
3.1 Introduction	53
3.2 Related Work	54
3.3 Materials and Methods	56
3.3.1 Depth Cues	57
3.3.2 Experiment	60

3.4	Results	64
3.4.1	Objective Measures	64
3.4.2	Subjective Measures	67
3.5	Discussion	67
3.6	Conclusion	71
CHAPTER 4	CONTEXTUAL AMBIENT OCCLUSION: A VOLUMETRIC RENDERING TECHNIQUE THAT SUPPORTS REAL-TIME CLIPPING	73
4.1	Introduction	73
4.2	Related Work	76
4.3	Methods	79
4.3.1	Pipeline	80
4.3.2	Shadow test volume clipping	82
4.3.3	Local Update of LAO	84
4.3.4	Precomputation of the Dilation Shape Depth Maps	87
4.3.5	Steps performed each frame	90
4.3.6	Multiple simultaneous clipping meshes	92
4.4	Evaluation	94
4.5	Discussion	100
4.6	Conclusion	103
CHAPTER 5	ANATOMYCARVE: A VR OCCLUSION MANAGEMENT TECHNIQUE FOR MEDICAL IMAGES BASED ON SEGMENT- AWARE CLIPPING	105
5.1	Introduction	105
5.2	Background and Related Work	107
5.2.1	Anatomical Illustration	107
5.2.2	Occlusion Management Techniques	108
5.2.2.1	Virtual X-ray	108
5.2.2.2	Volumetric Probe	110
5.3	Methods	111
5.3.1	Core Concept	111
5.3.2	Interaction Paradigm	111
5.3.3	Implementation	114
5.4	User Studies	116
5.4.1	Novice Usability Evaluation	116
5.4.1.1	Data Preparation	117
5.4.1.2	Pre-tests Questionnaire and Tutorial	117
5.4.1.3	Task 1: Illustrative Recreation	118
5.4.1.4	Task 2: Illustration Ranking	118
5.4.1.5	Evaluation Measures	119
5.4.2	Expert Evaluation	120
5.4.2.1	Data Preparation	120

	5.4.2.2	Pre-test Questionnaire and Tutorial	124
	5.4.2.3	Task	124
	5.4.2.4	Evaluation Measures	124
	5.4.3	Apparatus	124
5.5	Results		125
	5.5.1	Novice Usability Evaluation	125
	5.5.1.1	Participants	125
	5.5.1.2	Task 1: Results	125
	5.5.1.3	Task 2: Illustration Ranking	126
	5.5.2	Expert Evaluation	128
	5.5.2.1	Participants	128
	5.5.2.2	Results	129
5.6	Discussion		130
5.7	Conclusion		131
CONCLUSION AND RECOMMENDATIONS			133
6.1	Impact and Contributions		137
6.2	Future Work		138
APPENDIX I MANUSCRIPT 2 ADDITIONAL NOTES			141
LIST OF REFERENCES			147

LIST OF TABLES

		Page
Table 4.1	Comparison of existing volumetric ambient occlusion methods	78
Table 4.2	Scenes tested during the evaluation of CAO	94
Table 4.3	Mean frames per second for each scene of CAO	96
Table 4.4	Performance comparison between using 1 and 2 clipping meshes in each scene for CAO	98
Table 4.5	Per-voxel difference between the ambient occlusion factor obtained with a different number of rays and the ground truth volume for CAO	99
Table 4.6	Per-pixel difference between the image pixel values obtained with a given number of rays and the ground truth image for CAO	100
Table 5.1	SUS Results for AnatomyCarve	126

LIST OF FIGURES

	Page
Figure 1.1	Electromagnetic spectrum with sources 10
Figure 1.2	X-ray image of a normal hand 11
Figure 1.3	CT scanner 12
Figure 1.4	CT angiography image and subtraction CTA 14
Figure 1.5	T1 and T2 MRI images 15
Figure 1.6	Medical imaging pipeline 17
Figure 1.7	Ray casting steps 18
Figure 1.8	Colored unit cube used for ray setup 19
Figure 1.9	Trilinear interpolation 20
Figure 1.10	Blinn-Phong shading model components 22
Figure 1.11	Size-based classification of an aneurysm 24
Figure 1.12	Comparison of the depth contrast provided by depth cues 26
Figure 1.13	Motion parallax depth cue 28
Figure 1.14	Binocular disparity depth cue 29
Figure 1.15	Modern VR headset Meta Quest 3 31
Figure 1.16	Transfer function example 35
Figure 1.17	Comparison between the Phong, pseudo-chromadpeth and an illustrative technique 37
Figure 1.18	Angiographic cerebral medical scan visualized using static and dynamic shading 38
Figure 1.19	Different visualizations of a cerebral medical scan 40
Figure 1.20	Volume carving with a cubic clipping shape on a cerebral medical scan . 44
Figure 1.21	Hinge Spreader deformation tool 45

Figure 1.22	The symmetrical bare-hands "Volume Cracker" tool	48
Figure 3.1	Static and dynamic shading	58
Figure 3.2	Static and dynamic chromadepth	59
Figure 3.3	Static and dynamic aerial perspective	60
Figure 3.4	Physical setup superposed with the image observed in the VR headset ..	63
Figure 3.5	Collected data for each metric	65
Figure 3.6	Collected subjective data for each metric	68
Figure 4.1	CAO applied on medical volumes	74
Figure 4.2	Ray cast performed for each voxel to calculate the LAO factor	75
Figure 4.3	Visual representations of the 6 steps necessary to obtain the volume shaded using CAO	81
Figure 4.4	Shadow test volume clipping	84
Figure 4.5	3 possible categories for the voxel in LAO	85
Figure 4.6	3 possible categories for the voxel in LAO with boundaries	86
Figure 4.7	Computation performed for every pixel during the Minkowski sum calculationl	89
Figure 4.8	Comparison between CAO images created with and without anti-aliasing	91
Figure 4.9	Scenes used for the evaluation of the CAO algorithm	95
Figure 4.10	Runtime computation cost breakdown for a single frame using the LAO and CAO algorithms	97
Figure 4.11	Modified Engine scene with two clipping meshes	99
Figure 4.12	The 2D error between the image rendered with the 6-ray configuration and the ground truth image	102
Figure 5.1	Comparison between anatomical book illustrations and their recreation with AnatomyCarve	106
Figure 5.2	A simulation of the Joshi et al. clipping algorithm	109

Figure 5.3	A carving configuration with two clipping spheres	112
Figure 5.4	The basic functionality of AnatomyCarve	113
Figure 5.5	Volume rendering pipeline of AnatomyCarve	116
Figure 5.6	The first and the last volumes that participants had to copy in the AnatomyCarve usability user study	120
Figure 5.7	The volume presented in the 5th trial in the AnatomyCarve usability user study	121
Figure 5.8	The volume presented in the 7th trial in the AnatomyCarve usability user study	122
Figure 5.9	The volume presented in the 3rd trial in the AnatomyCarve usability user study	123
Figure 5.10	NASA TLX score of novice participants in the AnatomyCarve user study	127
Figure 5.11	Average MAE of the first encountered segment for each pixel in the AnatomyCarve user study	128

LIST OF ALGORITHMS

	Page
Algorithm 4.1	Calculation of a dilated depth value executed for every pixel 90
Algorithm 4.2	Anti-aliased shadow test clipping calculation executed for every voxel 93

LIST OF ABBREVIATIONS

2D	Two-Dimensional
3D	Three-Dimensional
AAS	Axis-Aligned Slicing
ANOVA	Analysis of Variance
AO	Ambient Occlusion
AR	Augmented Reality
AS	Arbitrary Slicing
BHVC	Bare-Hands Volume Cracker
CAO	Contextual Ambient Occlusion
CI	Confidence Interval
CPU	Central Processing Unit
CR	Computed Tomography
CT	Computed Tomography
CTA	Computed Tomography Angiography
DoF	Degree of Freedom
DVR	Direct Volume Rendering
FBO	Framebuffer Object
FXAA	Fast Approximate Anti-Aliasing
GitHub	Online platform for code repository hosting

GPU	Graphics Processing Unit
HLSL	High-Level Shading Language
HMD	Head-Mounted Display
LAO	Local Ambient Occlusion
LCD	Liquid Crystal Display
MAE	Mean Absolute Error
MRI	Magnetic Resonance Imaging
MSAA	Multisample Anti-Aliasing
PDF	Portable Document Format
PRISM	Platform for Real-time Interactive Shader Modification
RAM	Random Access Memory
RGB	Red, Green, and Blue colors
RGB α	Red, Green, and Blue colors with an alpha channel
RMS	Root Mean Square
RMSE	Root Mean Squared Error
SAT	Summed Area Table
SR	Surface Rendering
SSAO	Screen-Space Ambient Occlusion
SUS	System Usability Scale
TF	Transfer Function

TLX	Task Load Index
Unity3D	A cross-platform game engine
URL	Uniform Resource Locator
VC	Volume Cracker
VR	Virtual Reality
VRAM	Video Random Access Memory
VSS	Void Space Surfaces
XR	Extended Reality

LIST OF SYMBOLS AND UNITS OF MEASUREMENTS

CI	Confidence interval
cm	Centimeter
F	F-statistic in ANOVA
FPS	Frames per second
GB	Gigabyte
Hz	Hertz
M	Arithmetic Mean
mm	Millimeter
ms	Millisecond
p	P-value in ANOVA
px	Pixel
R^2	Coefficient of determination
SD	Standard deviation
χ^2	Chi-square statistic
°	Degree
$\wedge$	Conjunction
$\vee$	Iterated disjunction

INTRODUCTION

Medical image visualization enables researchers and medical professionals to explore anatomical datasets without a predefined hypothesis, to evaluate and refine existing hypotheses by visualizing measurement results, and to communicate findings clearly and effectively (Preim & Bartz, 2007, p. 1). It plays an important role in surgical training, diagnosis, preoperative planning, and intraoperative guidance (Serra *et al.*, 1998; Kersten-Oertel, Chen & Collins, 2014). For instance, during preoperative planning for brain tumor surgery, neurosurgeons analyze patient imaging data (e.g., CT or MRI) to develop a detailed understanding of patient-specific anatomy and to determine an appropriate surgical approach. In this context, it is important to understand the spatial relationships among anatomical structures in the tumor region to assess resectability and to plan the optimal resection strategy (Preim, Baer, Cunningham, Isenberg & Ropinski, 2016).

Clinical imaging increasingly relies on volumetric data to support spatial reasoning and treatment planning. A key challenge in visualizing three-dimensional medical images is enabling viewers to interpret spatial relationships between anatomical structures (depth perception) while maintaining visibility of relevant regions despite overlapping tissues (occlusion management). Because the eyes receive only two-dimensional projections of the world, the visual system must reconstruct depth from a set of perceptual indicators known as depth cues. In densely structured anatomy, however, many tissues lie along the same line of sight, so superficial structures can obscure deeper ones and leave important detail concealed even across multiple viewpoints (Joshi *et al.*, 2008; Weiskopf, Engel & Ertl, 2003). To situate the contributions of this thesis, the following sections outline the specific perceptual and visualization challenges associated with depth perception and occlusion in volumetric medical data.

0.1 Depth Perception of Cerebrovascular Volumes

In the context of this thesis, depth perception is particularly relevant for cerebrovascular imaging, where thin, branching vessels intersect and overlap. On a two-dimensional monitor, the visual system must infer depth from depth cues, that is, image regularities that signal relative distance and spatial ordering (Serra *et al.*, 1998; McGuffin, Tancau & Balakrishnan, 2003). Occlusion is a primary cue: a nearer structure partially hides a farther one, and the shared boundary indicates which is closer to the viewer. Shading is another key cue: systematic changes in light and dark produced by illumination and local orientation help recover local shape and relative depth (Serra *et al.*, 1998; McGuffin *et al.*, 2003). In angiographic volumes, these signals are often weak. Frequent crossings, small vessel diameters, and translucent direct volume rendering diminish both occlusion boundaries and shading gradients, so depth ordering becomes ambiguous unless additional cues are introduced (Chu *et al.*, 2018). The problem is amplified in neuroanatomy, where tight spatial relationships require precise judgments of connectivity, and patient-specific variability limits the transfer of prior experience to new cases (Kersten-Oertel *et al.*, 2014). These perceptual limitations motivate the investigation of how interactive depth cues in virtual reality can enhance users' ability to interpret complex three-dimensional vascular structures.

0.2 Occlusion Management in Medical Volumes

Occlusion management becomes difficult when visualizing volumetric data containing densely packed anatomical structures. When many layers align along the same line of sight, outer tissues and nearby structures block deeper ones; therefore, large portions of the target anatomy remain hidden within the surrounding context even when the viewpoint is changed (Joshi *et al.*, 2008; Weiskopf *et al.*, 2003). This self-occlusion prevents seeing all structures at once. In segmented data, the volume is partitioned into labeled regions such as organs or vessels. In this scenario, preserving correct spatial relationships between identified parts is a central

challenge (Pham, Xu & Prince, 2000). On the other hand, in unsegmented data, the volume consists of raw intensity values without labels, so boundaries and shapes must be inferred from shading cues (Levoy, 1988). For unsegmented volumes, stronger depth cues are often needed (Stewart, 2003). Ambient occlusion can improve the sense of shape and layering, yet simple variants are costly to update during interactive clipping (Hernell, Ljung & Ynnerman, 2010) and screen-space approaches may create small inconsistencies between the two eyes in stereo displays (Shi, Billeter & Eisemann, 2022). These challenges motivate investigating how rendering and interaction techniques can better support viewers in revealing, exploring, and maintaining context for occluded anatomical structures in volumetric datasets.

0.3 Context

To address these challenges, numerous rendering and interaction techniques have been proposed to improve the understanding of medical data (Kersten-Oertel *et al.*, 2014; Laha & Bowman, 2013). Their effectiveness, however, depends not only on the underlying algorithms but also on how users can manipulate the data and what depth information the display conveys during exploration. On a typical desktop, visualization occurs on a single two-dimensional monitor using a keyboard and mouse. This setup restricts rotation, translation, and scaling to a limited number of axes and decouples head motion from viewpoint motion (Heinrich, Bornemann, Lawonn & Hansen, 2020). Such constraints weaken several depth cues that many visualization methods aim to enhance (Laha & Bowman, 2013; Kersten-Oertel *et al.*, 2014). Consequently, techniques that perform well in controlled conditions may underperform in everyday use, particularly for complex and densely overlapping structures such as angiographic vessels (Kersten-Oertel *et al.*, 2014).

For unsegmented volumes, desktop viewing without head tracking suppresses important depth cues and is associated with lower accuracy and confidence during volumetric analysis, whereas immersive displays that restore these cues can improve performance (Laha & Bowman, 2013;

Kersten-Oertel *et al.*, 2014). For segmented data, where structures are labeled as organs or vessels, conventional mouse-and-keyboard interaction limits natural manipulation and makes it difficult to maintain spatial relationships among multiple parts on a flat monitor, as occlusion and context loss persist (Preim & Botha, 2013; Joshi *et al.*, 2008; Heinrich *et al.*, 2020).

0.4 Virtual Reality

Virtual reality (VR) is a promising technology that has the potential to improve upon both the depth perception and the occlusion management problems related to the visualization of medical images. In this context, VR denotes an immersive, headset-based environment that presents a computer-generated three-dimensional scene and continuously updates the viewpoint in response to head movement. With recent advancements in VR hardware and motion-tracking input devices, new paradigms for human-computer interaction (HCI) have become possible. Virtual reality significantly enhances depth perception and interaction, with modern headsets providing richer depth cues and supporting more natural manipulation than standard desktop environments. One key depth cue is binocular disparity (Preim *et al.*, 2016), which improves depth perception at close viewing distances by presenting slightly different images to each eye. Another important cue is motion parallax (Preim *et al.*, 2016), which arises from the relative movement of objects as the observer's viewpoint changes, allowing the visual system to infer three-dimensional structure and spatial relationships within a scene. Ware & Mitchell (2008) demonstrated that combining stereoscopy and motion parallax is particularly effective for enhancing depth perception, suggesting that VR could substantially improve the perception of medical data. Furthermore, ongoing research is exploring varifocal VR displays to further enhance depth perception (Toulouse, Thiele & Herkommer, 2019). These displays reintroduce the accommodation cue, which is the depth signal generated when the eye adjusts its crystalline lens to bring objects at different distances into sharp focus. As a result, they will eventually provide a more natural and physiologically accurate depth experience in virtual environments.

In terms of interaction, modern VR input devices, such as handheld controllers or Leap Motion sensors, enable more natural gestures and offer significantly greater degrees of freedom (Laha & Bowman, 2013; Laha, Bowman & Socha, 2016). Moreover, VR hardware has become substantially more affordable in recent years, making its integration into clinical and hospital environments increasingly feasible.

Contemporary VR headsets provide strong depth cues, precise head tracking, and natural hand interaction, which together can enhance depth perception and support occlusion management. In cerebral angiography, the combination of richer depth cues with direct spatial interaction can improve the interpretation of thin, branching vessels that complicate depth judgments. These capabilities also generalize to other three-dimensional datasets with substantial self-occlusion, where interactive manipulation and view-dependent cutting can reveal task-relevant structures while preserving contextual anatomy.

0.5 Research scope and methodology

The goal of this research is to investigate how the capabilities of modern VR systems can be leveraged to advance the visualization and interaction with medical imaging data. We aim to investigate how VR technology can aid depth perception and occlusion management problems present when visualizing medical images. Specifically, the work focuses on three core aspects of VR technology: improved depth perception, rendering, and interaction. It examines how these features can be harnessed to improve the clarity, intuitiveness, and overall effectiveness of medical image visualization. In this dissertation, we address three specific research questions (RQ) that have the potential to advance research in medical visualization, with the following sub-goals:

RQ1. *How do existing depth perception techniques developed for 2D displays carry over to virtual reality?*

The goal of this research question was to revisit several depth cues evaluated using two-dimensional displays and quantify how effective they are in VR. We aimed to conduct a user study on predefined angiographic volumes using a simple depth-judgment task to compare these depth cues, and analyze accuracy and behaviour. We asked participants to judge which elements of a virtual 3D scene appear nearer, observed their interactions in VR, and used these data to determine which basic visual cues most effectively support accurate spatial judgment in VR.

RQ2. *How is it possible to combine an existing occlusion management technique (clipping) with a depth enhancement technique in VR to improve depth perception of unsegmented volumes?*

The goal of this research question was to determine how specialized rendering can enhance perceived depth during clipping in unsegmented volumetric data. Clipping (Joshi *et al.*, 2008) removes user-selected parts of a 3D dataset with cutting planes or volumes to reveal hidden structures while preserving spatial relationships. This work aimed to implement an ambient occlusion algorithm, meaning a shading method that subtly darkens regions enclosed by nearby tissue to clarify depth. The difficulty arises when clipping is combined with this method, since each cut changes local neighbourhoods and makes full recomputation in each frame impractical. We aimed to design an incremental update that recalculates only values affected by the clip and reuses cached results elsewhere, enabling real-time rendering without loss of quality.

RQ3. *How does the interactivity provided by VR improve occlusion management when visualizing segmented medical data?*

The goal of this research question was to explore how the interactivity of VR headsets can support a new occlusion management technique for segmented volumetric images, where each region is labelled as a distinct anatomical structure. Inspired by anatomy books that use dissections to create cut-away views (Viola, Kanitsar & Groller, 2004),

we sought to design an interaction that lets the viewer virtually dissect a segmented medical volume to reveal internal structures. Building on clipping, the user would insert simple clipping shapes into selected segments to remove obstructing material and expose deeper anatomy, enabling highly customized views while preserving spatial relationships. We intend to evaluate the technique for usability with non-experts and assess its usefulness for surgical planning with neurosurgeons and residents.

0.6 Original contributions

This research led to the following contributions:

- Adaptation of depth cues, namely shading, pseudo chromadepth, and aerial perspective, previously studied on 2D displays to VR to examine their effectiveness in a stereoscopic, hand-interactive context.
- A VR study of cerebral angiography that compares the aforementioned depth cues in both static and dynamic variants between each other. The dynamic version of the cues was controlled with a virtual surgical pointer, and all volumes were rendered via standard ray casting with Blinn-Phong shading (Blinn, 1977). The results inform design recommendations for new VR visualization techniques.
- Development and validation of Contextual Ambient Occlusion (CAO), a volumetric ambient occlusion method that supports real-time clipping while matching Local Ambient Occlusion image quality by updating only voxels affected near the clipping region. The algorithm enables multiple clipping shapes and maintains both visual quality and interactive frame rates.
- Contribution of an open-source CAO algorithm package to the community, facilitating integration into interactive systems.
- Development and validation of AnatomyCarve, an occlusion management technique based on segment-aware clipping. The proposed technique features asymmetrical two-hand interaction

with per-segment clippable masks and stacking of clipping spheres to carve different segments. These features make it possible to create illustrations similar to anatomical books, mimicking the stepwise dissection paradigm often used in the creation of these illustrations.

- Contribution of two open-source AnatomyCarve packages to the community: one enabling VR interaction and the other supporting desktop interaction.
- Introduction of two metrics for medical image comparison of the rendering of 3D images: Mean Absolute Error (MAE) of the first encountered segment (objective metric) and a global ranking derived from participants' judgments (subjective metric).

0.7 Thesis organization

Chapter 1 reviews prior work on techniques and algorithms for medical image visualization and introduces foundational concepts in image formation, volumetric rendering, human visual perception, and virtual reality. Chapter 2 states the research objectives, and the relationship of each article to these objectives is detailed. Chapters 3 through 5 present the articles. In Chapter 3, a comparative study of depth cues with a virtual reality headset is reported. In Chapter 4, Contextual Ambient Occlusion is introduced as a volumetric rendering technique for interactive exploration. In chapter 5, AnatomyCarve is presented, an application for visualizing unsegmented volumes designed for use in virtual reality. Chapter 6 discusses the results from chapters 3 to 5, synthesizes their implications, and concludes the thesis.

CHAPTER 1

BACKGROUND AND LITERATURE REVIEW

This chapter introduces how medical images are formed to describe the data we visualize. A focus is placed on Computed Tomography (CT), Computed Tomography Angiography (CTA), and Magnetic Resonance Imaging (MRI). X-rays are introduced first, as they provide the physical basis for CT and CTA. Core ideas in volume rendering are then presented to show how volumetric data are converted into viewable images. Human visual perception is reviewed next, with limits that affect the interpretation of three-dimensional scenes and volumetric images highlighted, along with perceptual cues that help to mitigate these limits. A brief history of virtual reality is then provided, and the ways in which modern head-mounted displays can support the exploration of medical volumes are explained. Finally, methods that improve depth perception in angiographic visualization are summarized, and rendering and interaction techniques for densely populated volumetric datasets, including unsegmented and segmented volumes, are reviewed.

1.1 Creation of Medical Images

In order to explain the limitations of visualizing medical images, we should first explain how they are created. Although there exist many types of imaging modalities, in this thesis, we only focus on CT, CTA, and MRI images, and briefly describe the physics behind these imaging techniques.

In the following section, we will begin by first describing X-ray images, which are a foundation for the creation of CT images. After, we describe the generation of CT and CTA images, and finally, of MRI.

1.1.1 X-Ray

X-ray imaging, also known as projective radiography, is one of the most established techniques in medical imaging. It works by transmitting a beam of ionizing electromagnetic radiation

through the human body and recording how much energy is absorbed or attenuated by different tissues (Prince & Links, 2014, p. 135). Because X-rays are not part of the visible spectrum (see Figure 1.1), the transmitted radiation is converted by a film-screen or digital detector into visible light or electrical signals to form an image.

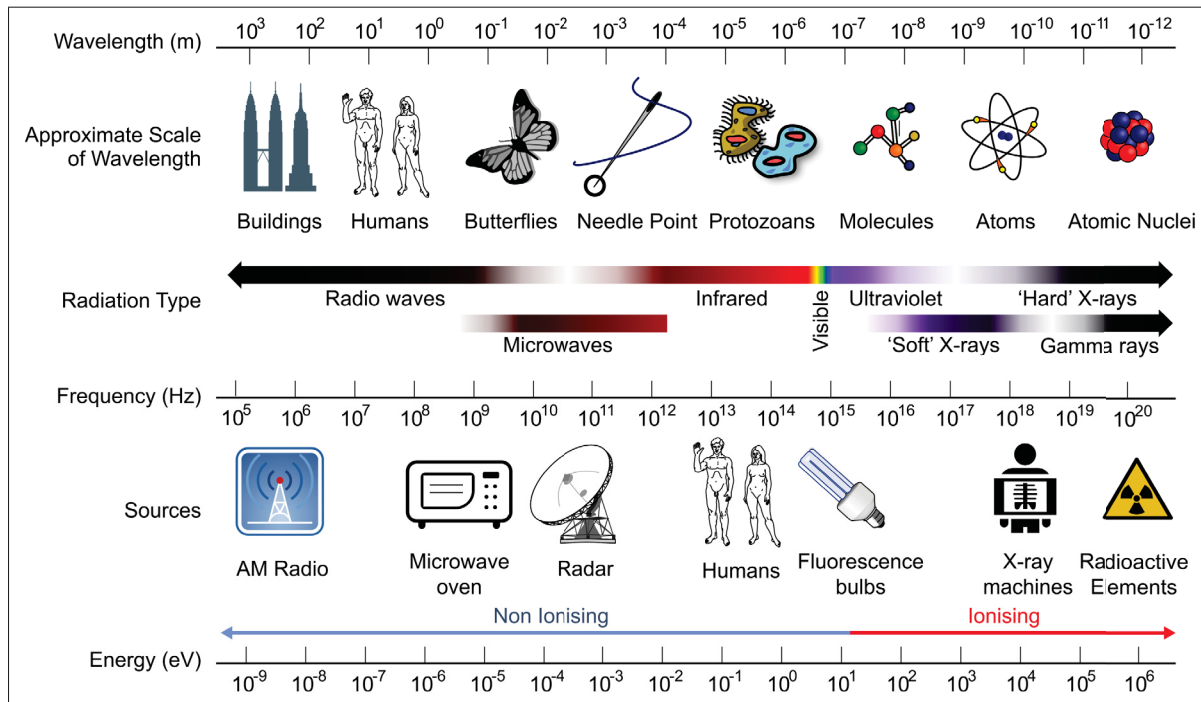


Figure 1.1 Electromagnetic spectrum with sources. X-rays used in medical imaging are located in the high-energy, ionizing range beyond the visible spectrum
Taken from Dinksbumf, Inductiveload and NASA, 2019, Wikimedia Commons, CC BY-SA 3.0

The X-rays used for diagnostic purposes are generated in an X-ray tube, where high-energy electrons are accelerated toward a metal target to produce photons (Prince & Links, 2014, p. 136). This beam is directed toward the body region of interest, and the varying absorption of X-rays by tissues of different densities creates the image contrast characteristic of radiographs. Denser materials such as bone absorb more radiation and therefore appear brighter, while softer tissues or air-filled regions allow more X-rays to pass through, appearing darker (Prince & Links, 2014, p. 119).

The transmitted radiation is captured on the opposite side of the body by a detector that records the resulting pattern to create a two-dimensional image (see Figure 1.2). In modern systems, digital detectors provide faster acquisition, improved image quality, and easier post-processing compared to traditional film. X-ray imaging remains a rapid and cost-effective method for visualizing internal anatomical structures, fractures, and other pathologies.



Figure 1.2 X-ray of a normal hand. Lighter areas demonstrate high radiodensity and correspondingly greater X-ray absorption

Taken from Mikael Häggström, 2017, Wikimedia Commons, CC0

1.1.2 Computed Tomography

Computed Tomography (CT) can be viewed as a three-dimensional extension of X-ray imaging, which reuses many of its principles (Prince & Links, 2014, p. 186). CT produces a stack of 2D X-ray slices, that combined together, create a 3D volume of data. However, unlike X-ray imaging, CT imaging creates 2D slices by making an X-ray beam rotate around the patient, which generates a series of one-dimensional projections at different angles, which are then combined together to calculate a single 2D slice. This process is repeated for each 2D slice, to further combine them into a single 3D volume.



Figure 1.3 CT scanner featuring a patient table and a gantry that houses the rotating X-ray tube and detectors
Taken from Tomas Vendis, 2024, Wikimedia Commons, CC BY-SA 4.0

In a typical CT scanner (see Figure 1.3), the patient is laid down on a small bed called the "patient table" (Prince & Links, 2014, p. 196). Around this bed, a donut-like frame is situated, called the "gantry", that contains the rotating X-ray generation tube, as well as X-ray detectors. In a conventional CT machine, the X-ray tube rotates around the patient table and generates

X-rays at different angles that are then recorded by the detectors after passing through the patient. After a sufficient number of X-rays are recorded, the patient table is moved automatically to start creating a new 2D scan of the patient.

In CT, angle-wise 1D projections form the 2D Radon transform of a slice, and applying its inverse reconstructs the original image (Prince & Links, 2014, p. 186). Each projection integrates the X-ray attenuation coefficient along the ray, and attenuation is reported in Hounsfield units, a density-related scale tied to the material's atomic number (Prince & Links, 2014, p. 187). Reconstruction is implemented as filtered back projection: each projection is smeared back along its ray and summed over all angles (yielding a blurred image), then a high-pass/ramp filter removes the blur to restore detail (Prince & Links, 2014, p. 202, 205). After all the slices have been recalculated, they are combined together to form a single 3D volume.

1.1.3 Computed Tomography Angiography

Computed Tomography Angiography (CTA) is an imaging technique that can be perceived as an extension of CT imaging that specifically focuses on angiography, meaning visualization of the blood vessels (Baliyan, Shaqdan, Hedgire & Ghoshhajra, 2019). The resulting images enhance the blood vessels, which is done by injecting a contrast agent into the blood system of the patient, which would then absorb more X-rays than the blood vessels typically would, enhancing their visibility. This type of imaging is used to detect abnormalities in blood vessels, such as aneurysms (Johnson & Fishman, 2010), as well as blockages and stenosis (Opolski *et al.*, 2016).

The procedure starts with placing a catheter into the vein, injecting a contrast agent into the blood system, typically a high iodine concentration (Abbara *et al.*, 2016). Then, low-radiation scans are performed at regular intervals to see how the agent propagates in the blood system (Kumamaru, Hoppel, Mather & Rybicki, 2010). When its concentration reaches a certain threshold, then the full CT scan is performed, creating the CTA image (see left side of Figure 1.4).



Figure 1.4 CT angiography (left) and subtraction CT angiography (right). The standard scan visualizes all captured tissues, whereas the subtraction technique isolates contrast-enhanced blood vessels

Taken from Mikael Häggström, 2017, Wikimedia Commons, CC0

Another variety of the CTA scan, called the subtraction CT Angiography, could be created, which visualizes only the blood vessels of the system (see right side of Figure 1.4) (Meijer *et al.*, 2019). In this imaging technique, a full CT scan is taken before the contrast agent injection, and another one is taken after. This procedure can improve image quality by only focusing on the visualization of angiographies (Jayakrishnan *et al.*, 2003).

1.1.4 Magnetic Resonance Imaging

Magnetic resonance imaging is an imaging modality based on the behavior of atomic nuclei (center of the atom) when exposed into strong magnetic fields (Prince & Links, 2014, p. 411). Unlike X-ray and CT, it relies on non-ionizing radio-frequency electromagnetic energy (Prince & Links, 2014, p. 413) instead of ionizing radiation, avoiding the risks associated to radiation exposure of X-rays (Prince & Links, 2014, p. 1). Also, instead of focusing on radiodensity, MRI allows for visualization of soft tissues that present significant fat and water (Prince & Links, 2014, p. 408).

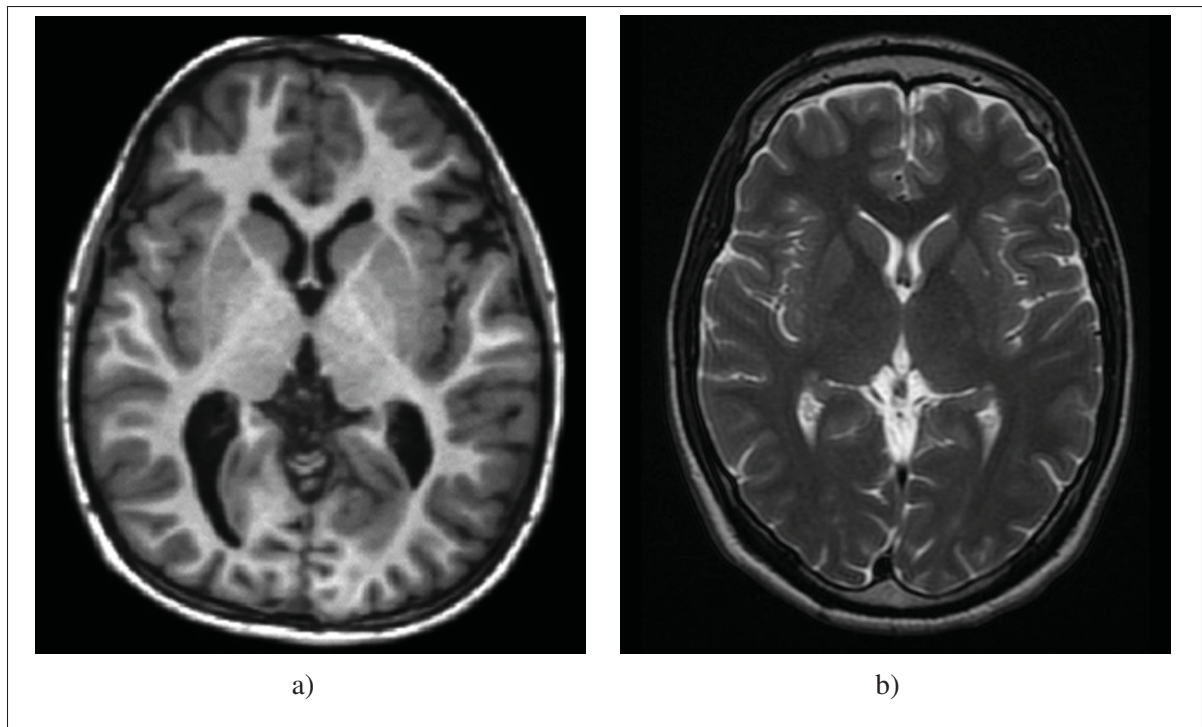


Figure 1.5 T1-weighted (a) and T2-weighted (b) axial brain MRIs from different subjects. Intensities reflect distinct tissue properties: in (a), cerebrospinal fluid (CSF) is black, white matter is light gray, and gray matter is dark gray. In (b), CSF is bright white, white matter is dark gray, and gray matter is light gray

Taken from Mim.cis, 2016, Wikimedia Commons, Public Domain (left); Sean Novak, 2015, Wikimedia Commons CC BY-SA 4.0 (right)

MRI exploits the "spin" property of hydrogen nuclei. This intrinsic property gives each proton a tiny magnetic moment, and because the body is rich in water and fat (which are both hydrogen-dense), their signals are useful (Prince & Links, 2014; Gaeta *et al.*, 2024, p. 411). In a strong magnetic field, these magnetic moments align to produce a net magnetization vector for a given volume of tissue. A brief RF pulse then tips this net magnetization vector perpendicularly by 90 degrees and synchronizes the phase of the spins, after which the system relaxes and emits detectable signals (Prince & Links, 2014; Lugauer & Wetzl, 2018, p. 411). Relaxation follows two time constants T1 (realignment with the main field) and T2 (loss of phase coherence) that differ across tissues (Prince & Links, 2014; Pai, Shetty, Hodis & Chowdhury, 2025, p. 413). By adjusting timing, MRI emphasizes either T1 or T2, producing images that highlight fat

(T1-weighted; see Figure 1.5a) or water/CSF (T2-weighted; see Figure 1.5b) (Prince & Links, 2014, p. 429). These tissue-dependent differences are captured by the receiver coil and mapped to brightness in the reconstructed image (Prince & Links, 2014, p. 407).

To create a 3D image, the patient is positioned in a chamber with a strong magnetic field. The MRI machine then selects a slice by switching a gradient parallel to the main field so only a thin 2D plane responds to the RF pulse (Prince & Links, 2014, p. 442). Then, spatial encoding uses additional gradients so the signal's frequency and phase vary smoothly across left-right and front-back directions, and the scanner collects multiple lines of data in a raw data matrix called k-space, each with a slightly different gradient pattern that together encodes the whole slice (Gallagher, Nemeth & Hacein-Bey, 2008). A computer then performs an inverse Fourier transform on k-space to reconstruct the slice in the spatial domain. This process is repeated for multiple 2D slices that are aggregated into a single 3D volume (Gallagher *et al.*, 2008).

1.2 Volume Rendering

After acquiring the medical images, it becomes possible to visualize them. As discussed previously, scans such as CT or MRI consist of a series of two-dimensional slices captured at regular intervals, which, combined together, create a single three-dimensional array (Callahan, Callahan, Scheidegger & Silva, 2008). Such a 3D array is called a volume, and it consists of voxels, which map each 3D position in the volume to some data, which depends upon the modality that was used to generate the image. Notably, not all modalities provide absolute quantitative units. In the case of CT images, this data corresponds to a measure of the radio-density of the tissues and is measured with Hounsfield units (Prince & Links, 2014, p. 127), which is a scalar numerical value. Conversely, voxel values in standard MRI are generally relative and expressed in arbitrary units based on signal intensity (Shinohara *et al.*, 2014).

Although such medical scans can be rendered by visualizing each two-dimensional slice individually, more complex rendering approaches were developed that aggregate all slices

together and visualize them as a single three-dimensional volume, as illustrated in Figure 1.6. Such rendering techniques are called volume rendering (Preim & Bartz, 2007).

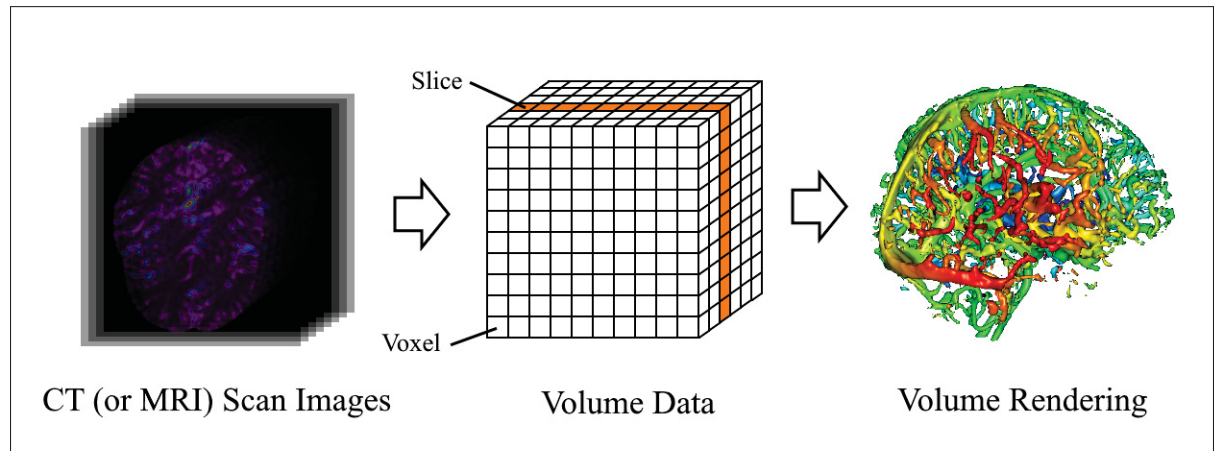


Figure 1.6 Medical imaging pipeline from 2D scan slices to 3D volume rendering. Individual slices from a CT or MRI scan are organized into a three-dimensional array of voxels to facilitate volumetric reconstruction and visualization

Volume rendering can be divided into two general categories: surface-based rendering (SR), also referred to as indirect volume rendering, and direct volume rendering (DVR) (Preim & Bartz, 2007). In SR, the volume is transformed into a mesh, and then surface rendering is used to visualize such a mesh. The advantage of such an approach is that it makes it easier to render the dataset, as it becomes made out of triangles, which modern hardware is optimized to render well (Lorensen & Cline, 1998). However, this transformation to a mesh is lossy as it forces to introduce "thresholds" that divide the space in the medical image into discrete surfaces while discarding all the space between these surfaces (Hansen & Johnson, 2011; Lorensen & Cline, 1998).

With DVR, no information is explicitly removed from the volume. The volume is sampled directly without pre-transformation to an intermediate representation (Preim & Bartz, 2007, p. 183) and these samples are mapped to opacity and color values. More recently, DVR became more widespread due to its advantage of explicit information removal and also due to the improvement of modern graphical hardware (Balsa Rodríguez *et al.*, 2013). However, SR also remains a popular rendering technique due to being less computationally expensive and

providing sharp, easily interpretable geometry, which is useful for anatomy with clear boundaries (van Ooijen *et al.*, 2003).

Many DVR algorithms exist, such as splatting (Mueller, Shareef, Huang & Crawfis, 1999), shear-warp (Lacroute & Levoy, 1994), texture mapping (Cullip & Neumann, 1993) and ray casting (Chen, Hua, Bao & Peng, 2003). However, in our work, we focus on ray casting as modern hardware allows using it in real time.

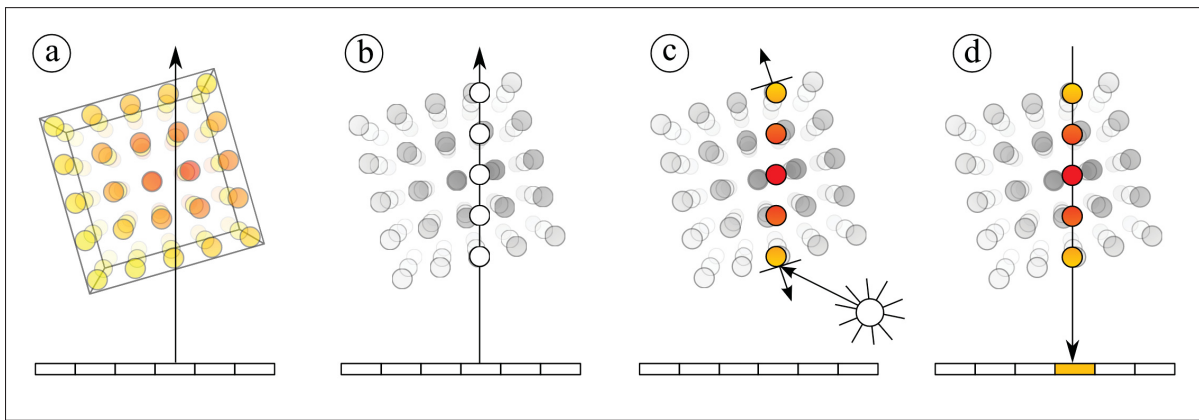


Figure 1.7 Fundamental stages of volume ray casting: (a) ray setup, (b) sampling, (c) classification, and (d) compositing. The pipeline transforms 3D volumetric data into a 2D image by accumulating optical properties along individual viewing rays
Taken from Florian Hofmann, 2011, Wikimedia Commons, CC BY-SA 3.0

Ray casting can be roughly divided into 4 general steps (see Figure 1.7), which are executed for every pixel of the final rendered image (Callahan *et al.*, 2008). In step 1 (Ray Setup), a ray is sent from the point of the viewer, which is used to calculate the color for a single pixel (Figure 1.7a). Then, in Step 2 (Sampling), the ray advances with discrete steps and samples the volume (Figure 1.7b). Then, in Step 3 (Classification), the sampled intensities from Step 2 are transformed into a color and opacity using transfer functions (Figure 1.7c). Finally, in step 4 (Compositing), the computed color and opacity are composited (i.e., blended) into a single color and opacity of the whole pixel (Figure 1.7d). We will go into more detail for each of the steps.

1.2.1 Ray Setup

In standard ray casting, a single ray is shot from every pixel of the final 2D image towards the volume. The main idea of ray setup is to determine the trajectory that the ray will have when traversing the volume. This process is typically implemented by using the colored cube algorithm (see Figure 1.8) presented by Kruger & Westermann (2003), which helps to define the position where the ray enters the volume and where it exits. In this algorithm, a colored cube that represents the normalized coordinates in the $[0, 0, 0]$ to $[1, 1, 1]$ range in terms of color is rendered to two different textures. In the first texture, the outer (front-facing) faces of the cube are rendered, while in the second texture, the inner (back-facing) faces are rendered. Together, these two textures encode the entry and exit points of the ray in normalized volume coordinates, which define the start and end positions for ray casting.

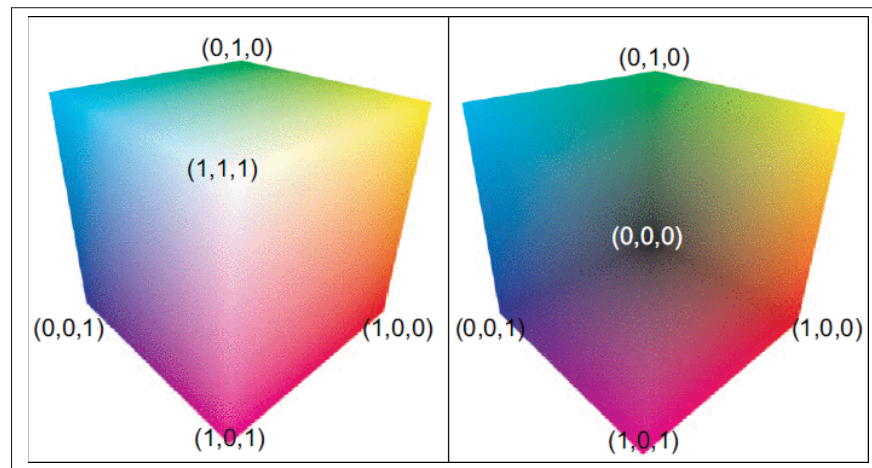


Figure 1.8 Colored unit cube used for ray setup. Mapping 3D spatial coordinates to RGB color values enables the efficient determination of ray entry and exit positions on the GPU

Reprinted with permission from Kruger & Westermann (2003). © 2003 IEEE

p. 15). It calculates a weighted average between all the surrounding voxels, where the weight is inversely proportional to the distance of the sample from the voxel. With this sampling method, linear interpolation is done for each of the three axes x , y , and z in a chained manner. The trilinear interpolation function f takes as parameters a normalized position (x, y, z) where $x, y, z \in [0, 1]$ and returns a scalar interpolated value between the 8 provided voxels, denoted as $V_{000}, \dots, V_{111}$. First, 4 linear interpolations are performed along the x axis:

$$\begin{aligned}
 c_{00} &= (1 - x) * V_{000} + x * V_{100} \\
 c_{01} &= (1 - x) * V_{001} + x * V_{101} \\
 c_{10} &= (1 - x) * V_{010} + x * V_{110} \\
 c_{11} &= (1 - x) * V_{011} + x * V_{111}
 \end{aligned} \tag{1.1}$$

Then, two linear interpolations are performed for the y axis:

$$\begin{aligned}
 c_0 &= (1 - y) * c_{00} + y * c_{10} \\
 c_1 &= (1 - y) * c_{01} + y * c_{11}
 \end{aligned} \tag{1.2}$$

And finally one linear interpolation is performed for the z axis:

$$f(x, y, z) = (1 - z) * c_0 + z * c_1 \tag{1.3}$$

The same process is repeated for every sample of the ray.

1.2.3 Classification

After acquiring the samples along the ray, it is time to classify them. Classification refers to the process of assigning a color (RGB) and an opacity (α) to each sample of the ray (Callahan *et al.*, 2008). This is done through the color and opacity transfer functions correspondingly, which may use the intensity calculated during the sampling step, the position of the sample, as well as any other additional properties to transform them into a color and an opacity. One of the most well-known algorithms in computer graphics is Blinn-Phong shading (Phong, 1975; Blinn, 1977), and it may also be used during ray casting.

Blinn-Phong shading is a photorealistic algorithm that simulates how light refracts from a surface in the real world. The Blinn-Phong model is composed of 3 independent components that are added together to create the final color, namely the ambient, diffuse, and specular components (see Figure 1.10). The ambient component represents light coming from an object uniformly from all directions, generally light that has reflected off other surfaces (indirect light) (Akenine-Möller, Haines & Hoffman, 2019). The diffuse component, on the other hand, represents light that is reflected directly from a light source, where surfaces that are oriented towards the light source are more lit than those that are at an angle. Finally, the specular component represents the reflection of the light source on the surface of an object. The model adds these 3 components together to create a single color.

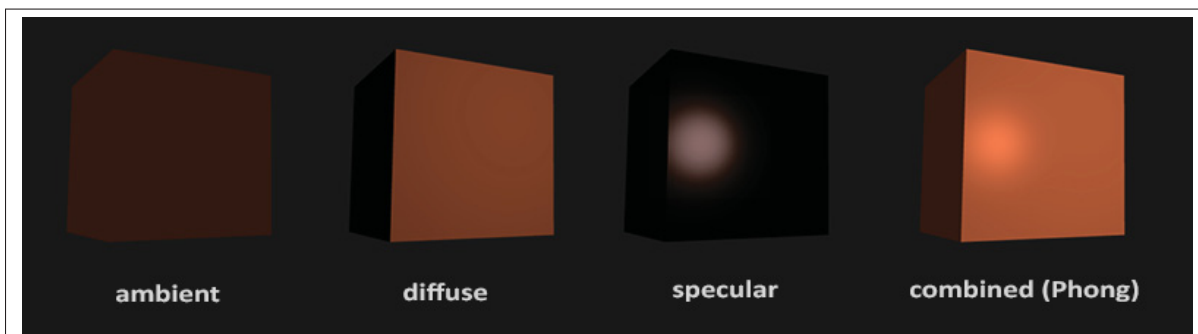


Figure 1.10 Ambient, diffuse, and specular components of the Blinn-Phong shading model. The combination of these lighting elements simulates realistic light-matter interaction to improve the perception of depth and shape in a rendered volume

Taken from Joey de Vries, 2025, LearnOpenGL, CC BY 4.0

Blinn-Phong shading requires the calculation of the "gradients" on the volume, also referred to as the "surface normals" (Levoy, 1988). These normals indicate the local direction of intensity variation within the volume at each voxel, and are commonly used to approximate light reflection. The normals are obtained by computing the intensity derivatives along the x , y , and z axes for each voxel, and then combining these three partial derivatives into a single vector, known as the gradient (Pfister, 2005). The concrete implementation typically used for this calculation is the Sobel operator (Bhattacharya & Wild, 1996).

The classification of volumetric data remains a fundamental challenge in medical volume rendering. To navigate the extensive literature in this domain, Ljung *et al.* (2016) provide a survey that groups transfer functions by their dimensionality, attributes, and rendering techniques. They explain that these functions operate by mapping specific data parameters to optical properties, namely color and opacity. A 1D transfer function takes a single scalar value, usually the material density, as its sole input. To achieve better material separation and cut down on visual clutter, 2D transfer functions introduce a second input variable, most commonly the gradient magnitude. The function then evaluates this 2D input pair to assign the final color and opacity, helping to highlight boundaries and suppress homogeneous regions. Stepping up the complexity, multi-dimensional transfer functions take multiple attributes from multivariate or multimodal datasets as inputs, allowing for a much more precise classification of complex structures.

Addressing the persistent need for better spatial context highlighted in recent surveys like Ljung *et al.* (2016), foundational work by Correa and colleagues introduced several innovative ways to expand the classification space. First, Correa, Silver & Chen (2006) developed feature-aligned techniques that use surgical metaphors like peelers and retractors to reveal internal structures interactively. To further overcome the limits of standard intensity classification, Correa & Ma (2008) developed size-based transfer functions (see Figure 1.11). This method uses a 3D scale field as an input dimension, isolating features by their local size even if they share the exact same density. Finally, in their 2009 work on the occlusion spectrum, Correa & Ma proposed using ambient occlusion to classify features based on their depth. This method is highly effective

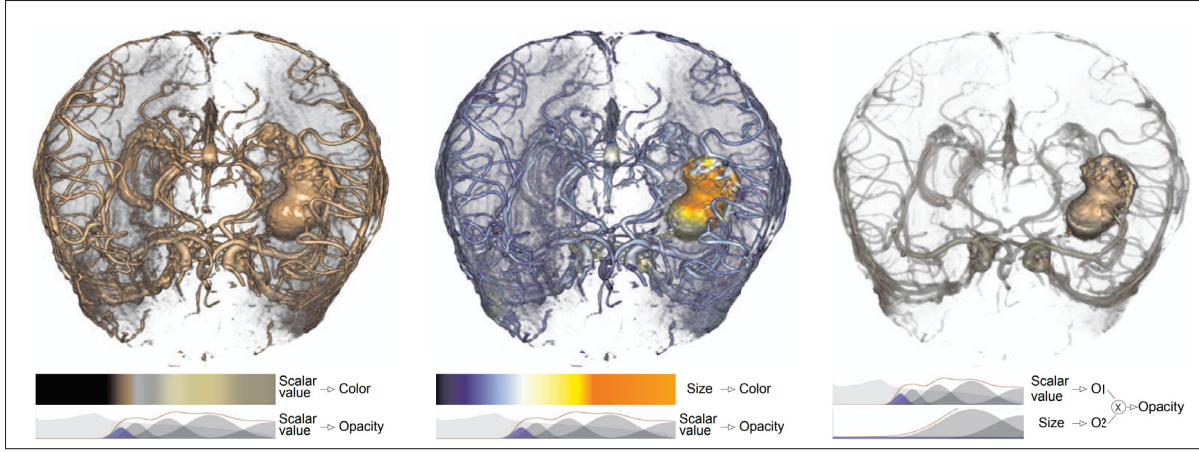


Figure 1.11 Size-based classification of an aneurysm. A traditional 1D transfer function based only on scalar value (left) visually blends the aneurysm with surrounding vessels. Mapping feature size to color (middle) clearly distinguishes the larger aneurysm. Mapping size to both color and opacity (right) isolates the main structure by emphasizing larger features and fading out smaller ones. Reprinted with permission from Correa & Ma (2008). © 2008 IEEE

for isolating coherent structures in complex medical datasets where traditional gradient-based approaches fail.

1.2.4 Compositing

Finally, the colors and the opacities of all the samples of the ray need to be aggregated together to create a single color and opacity for the pixel. This is done using compositing (Callahan *et al.*, 2008; Preim & Bartz, 2007, p.2), which, when performed following the direction of the ray, is written in the following manner:

$$\begin{aligned}
 C_{i+1} &= C_i + (1 - A_i) \alpha_i c_i \\
 A_{i+1} &= A_i + (1 - A_i) \alpha_i
 \end{aligned}
 \tag{1.4}$$

where C_i represents the accumulated RGB color at step i (assuming $C_0 = (0, 0, 0)$), A_i represents the accumulated opacity at step i (assuming $A_0 = 0$), α_i represents the opacity at sample i , and c_i represents the color at sample i . This iterative equation is applied to all the samples of the ray, and the resulting accumulated color and opacity are registered in the output pixel of the current ray.

1.3 Human Visual Perception

We now turn to human visual perception, whose inherent limitations are essential to understand in order to explain the challenges of visualizing volumetric medical images. The human eye can only perceive a 3D environment in the form of 2D projections, and because of that, depth information about the scene is lost (Preim *et al.*, 2016). To restore some of the lost depth information, the human visual system uses multiple cues to better perceive depth (Teittinen, 1993). Many of the cues can be perceived with a single eye. Those cues are called monoscopic and they are especially useful in the case of classical 2D monitors, which show the same image to both eyes (Ropinski, Steinicke & Hinrichs, 2006). However, there exist cues that require both eyes, which are called stereoscopic.

Cutting & Vishton (1995) constructed a graph illustrated in Figure 1.12 where they estimated the depth contrast provided by depth cues at different distances from the viewer. They have estimated that one of the best depth cues in terms of produced depth contrast is occlusion. It is a monoscopic depth cue where obstruction is used to define the depth order of the object. In other words, occlusion is when one object partially occludes another one. In that case, the obstructed object is seen as being further, while the fully visible one is closer (Guibal & Dresch, 2004). In addition to providing a high depth contrast, occlusion has the advantage that the viewer does not need to be familiar with the shape or size of the objects and this cue works equally well at any distance from the viewer.

Other cues work based on the viewer's familiarity with the size of the objects that they see. An example of such a cue is perspective, where closer objects appear bigger than further ones.

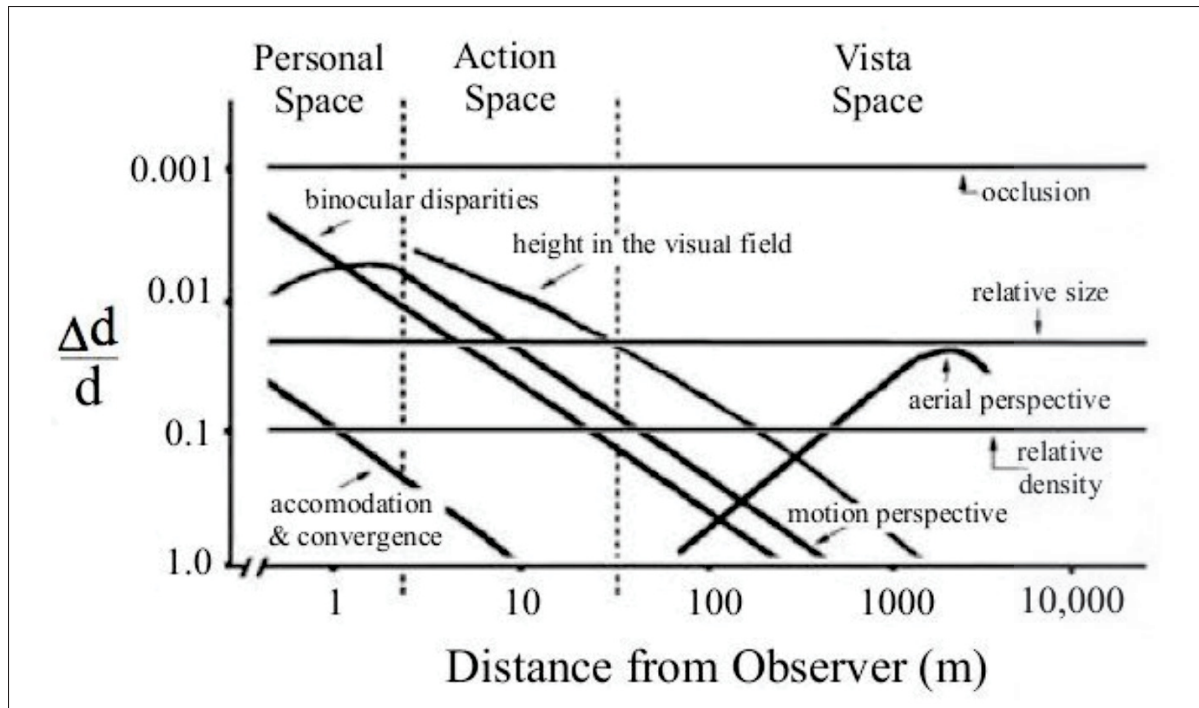


Figure 1.12 Effectiveness of various depth cues relative to the observer's distance. Specific visual cues, such as binocular disparity and occlusion, provide different levels of depth contrast depending on whether an object is located in personal, action, or vista space

Taken from Sweet & Kaiser (2013)

Perspective is especially effective when parallel lines are present in the scene, as these lines converge to some point at infinite distance (Zhai, Buxton & Milgram, 1996).

In addition to the size of the object, the way that they are shaded could also help to understand information about inner depth perception within an object. One such cue is shading, which refers to how much light is reflected from a light source depending on the orientation of the surface towards this light source (Palmer, 1999, p. 243). This cue allows to understand the orientation of different parts of the objects, which consequently helps to better understand their shape.

Some other cues directly encode the depth in the image using colors. Aerial perspective is an example of such cue where depth is encoded with a loss of contrast in the image (Kersten-Oertel *et al.*, 2014). It works similarly to the phenomenon of fog, where closer parts are highly

contrasted, and further parts fade into the color of the background. Chromadepth is another way of encoding depth where interpolation is performed between rainbow colors (Bailey & Clark, 1998). Chromadepth is not a natural depth cue that is actively used by the human visual system, but rather a technique based on an illusion that occurs because of the structure of the human eye. This illusion is called chromostereopsis (Thompson, May & Stone, 1993), and it occurs because rays with different colors are refracted to slightly different positions on the retina. This effect gives the impression that red is associated with closer depths, while blue is associated with further ones.

Some cues are based on movement of the observer and/or the scene, as for example, motion parallax (Gibson, Gibson, Smith & Flock, 1959). In that cue, when the viewer fixates their sight on a certain target in the scene and starts moving around it, objects start moving in different directions from the point of view of the observer. Additionally, the speed with which objects move indicates the relative distance from the target, with further objects moving faster than closer ones to the target. An illustration of the motion parallax cue is given in Figure 1.13.

Finally, some cues use stereoscopy. One such cue is binocular disparity (Qian, 1997), which is based upon the idea that the same scene is observed from two slightly different points of view, which results in the image being in slightly different places on the retina, and the brain is then able to deduce the relative distance to the objects in the scene based on the differences observed in the two images on the retina. Figure 1.14 demonstrates the binocular disparity cue by illustrating how two distinct points are able to produce different images on the retina of each eye. Another stereoscopic depth cue is convergence, which measures how much the eyes have to turn inward in order to perceive an object (Palmer, 1999, p. 205). The main idea of this cue is that both eyes can only fixate on one object in the 3D world at the same time, and when they do so, the closer the object is to the eyes, the more they have to be turned inward. A further cue tied to the human visual system is called accommodation (Palmer, 1999, p. 201). This cue refers to the dilation or retraction of the pupil in the eye when it tries to fixate on an object in the 3D world. The closer the object is to the eye, the more the pupil has to dilate to see it in focus, and

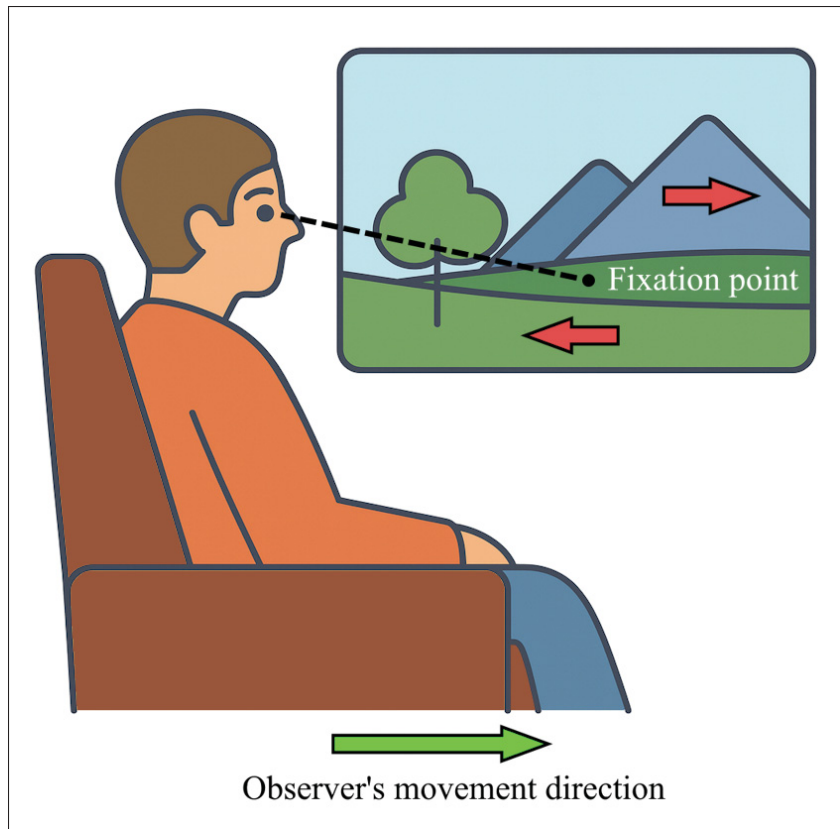


Figure 1.13 Motion parallax depth cue. When the moving observer fixates on a point in the scene, further objects than this fixation point move in the same direction as the observer, while closer objects move in the opposite direction

vice versa. Although not inherently stereoscopic, this cue is an integral part of the human visual system.

1.4 Virtual reality

Virtual reality is a three-dimensional simulation of the real world that can be interacted with, which also gives a sense of spatial presence (Abbas *et al.*, 2023). Such simulations usually feature immersive, realistic environments that can be manipulated, as well as real-time graphics (Zheng, Chan & Gibson, 1998). In this environment, the user can move around, observe it from

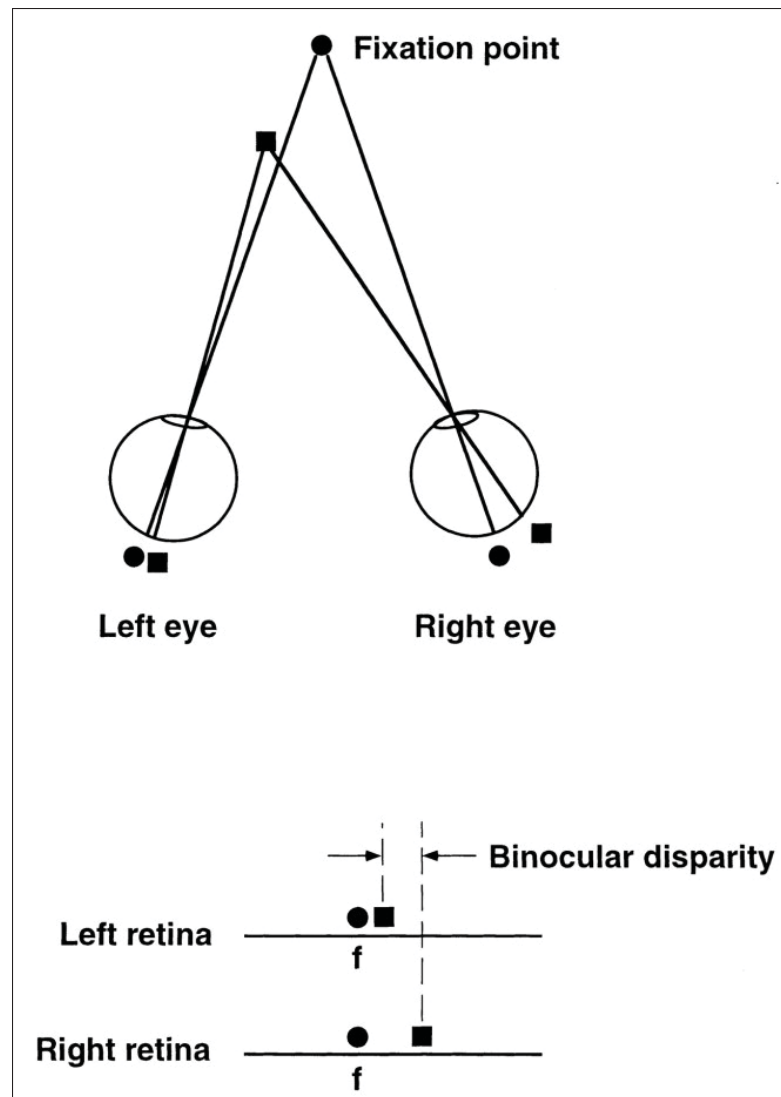


Figure 1.14 Binocular disparity as a primary depth cue. The horizontal shift between the images projected onto the left and right retinas provides the critical positional information used by the brain to calculate the relative depth of objects in 3D space
Taken from Qian (1997), Neuron, CC BY-NC-ND 4.0

different points of view, and modify it. In that sense, it is different from a classical desktop computer, as it does not have a standard GUI to interact with or any other sorts of commands.

Immersion refers to the idea that the simulation tries to remove all the distractions, so that the user of VR could only focus on the information that they want to work on, while interactivity refers to the idea that the user could manipulate and change the virtual world that is displayed to them (Zheng *et al.*, 1998).

1.4.1 History of Virtual Reality

The history of VR dates back to the 18th century (Lescop, 2017), when the first simulations of presence were made with the creation of panoramic paintings that fully filled the observer's field of view. VR remained implemented in terms of panoramas up until the 20th century, when the first head-mounted display (HMD) was introduced in 1968 (Lescop, 2017; Sutherland, 1968). Unlike the previous panoramic views, a head-mounted display was implemented in the form of a device that could be put in the participant's head to visualize some digital images. The key invention of the VR HMD was that it presented the viewer with a 3D image that changed when the participant was moving, which increased their immersion in the virtual environment. Although it was only an experiment, it laid the groundwork for future VR HMD development.

In the 1980s, VR became a topic of research (Fisher, McGreevy, Humphries & Robinett, 1987), and in 1989, the first commercially available VR headset called VPL EyePhone was released (Conn, Lanier, Minsky, Fisher & Druin, 1989). However, due to technical limitations of the technology at that time, such as poor resolution and high latency, the technology has lost its appeal. Further, due to the advancement of computational power, a revival of the VR technology has happened in the 2010s with the release of the Oculus Rift prototype in 2012 (LaValle, Yershova, Katsev & Antonov, 2014), and the subsequent release of commercially available headsets such as Samsung Gear VR in 2015, Oculus Rift in 2016, and HTC Vive Pro in 2018 (Lu, Ota & Dong, 2024). These headsets were commercially available for a price ranging from 99 to 599 USD, weighed 470-550 grams, and offered a high resolution. Additionally, starting from HTC Vive, the commercially available headsets started to come with tracked controllers (Sutherland *et al.*, 2019), which improved the interaction in the VR environment.

1.4.2 Modern Virtual Reality

Modern VR headsets (see Figure 1.15) offer improved immersion and engagement when visualizing a virtual world compared to a standard keyboard and mouse setup (Thamrongrat, Khundam, Pakdeebun & Nizam, 2023). This is achieved thanks to improved depth perception that modern VR HMDs offer, as well as improved interactivity.



Figure 1.15 Modern VR headset Meta Quest 3 with controllers
Taken from IHazACatNamedMax, 2023, Wikimedia Commons CC
BY-SA 4.0

To achieve stereo rendering, modern VR HMDs present separate images to each eye, typically using two small displays, or a single display split into left and right views positioned in front of the eyes (Hale & Stanney, 2014). To generate stereoscopic imagery for these screens, the 3D scene is rendered twice from two eye-centred viewpoints separated by the user's interpupillary distance, producing a left-right image pair with the appropriate binocular disparities.

One of the most important depth cues provided by VR HMDs that is not present in a desktop environment is motion parallax (Fulvio & Rokers, 2017). This cue is possible to achieve by moving the head while wearing the HMD, which, thanks to the real-time tracking, will readjust the movement in the virtual world. Another important cue is binocular disparity (Koulieris *et al.*, 2019), which is achieved by having two slightly different rendered images displayed in front of the eyes. In addition to binocular disparity, having two separate views allows the headset to also simulate the vergence cue by forcing the eyes to bend more for closer items. Finally, although not commercially available yet, some headsets provide the accommodation cue (Toulouse *et al.*, 2019) by applying a lens in real time while using the VR HMD. All of these cues are provided in addition to the cues that are used on a standard 2D monitor, such as occlusion, relative size, aerial perspective, and shading (Cutting, 1997).

In terms of interactivity, modern headsets provide controllers that offer 6 degrees of freedom (DoF) (Mazuryk & Gervautz, 2015), compared to only 2 present when using a standard computer mouse (MacKenzie, Soukoreff & Pal, 1997). On the mouse, these two DoFs represent the x and the y axis in which the mouse can be moved while remaining tracked. VR controllers add a third positional DoF in terms of the z -axis, which allows free movement of the controller in the 3D environment without having to remain on a surface like the mouse. In addition to the 3 positional DoFs, the VR controllers also offer 3 rotational DoFs (Mazuryk & Gervautz, 2015). This means that the user could rotate the controller around the x , y and z axes, which are called yaw, pitch, and roll, respectively. This allows the controllers to not only be freely moved in 3D space, but also freely rotated, which gives significantly more interaction possibilities than a computer mouse.

Modern VR headsets also present some important limitations that have an impact on the visual perception of the virtual world. One of them is the limited field of view (FoV), which refers to the angle of how much of the scene is visible through the headset's display (Melzer, Brozoski, Letowski, Harding & Rash, 2009). A small FoV impairs performance on spatial tasks such as target search and navigation, and diminishes presence, as the loss of peripheral cues makes users feel less immersed (Arthur, 2000; Barhorst-Cates, Rand & Creem-Regehr, 2016). Even modern

VR headsets such as the 2023 Meta Quest 3 have a combined horizontal binocular FoV of 110 degrees (Xiao *et al.*, 2025), which is significantly smaller than the normal human FoV, which can extend temporarily to up to 200 degrees (Ruia & Tripathy, 2025).

Another important limitation of modern headsets is the vergence-accommodation conflict (Hoffman, Girshick, Akeley & Banks, 2008). When viewing the natural world, eyes converge on a target, and simultaneously, lenses change focus to the same distance. In most HMDs, binocular disparity prompts the eyes to converge to a certain simulated depth, but the eye lens remains fixed at the headset's screen distance. This mismatch between convergence and focus (accommodation) can increase visual discomfort in the form of eye strain, cause headaches, and may also distort perceived distance and size (Hoffman *et al.*, 2008; Wang *et al.*, 2024).

A further issue that is important in modern VR headsets is setting the inter-pupillary distance properly so that it matches that of the person who is wearing the headset (Hibbard, van Dam & Scarfe, 2020). In the case that it is not set up properly, it can degrade image quality, cause discomfort, and create distortions in spatial perception (Livitcuka, Alksnis & Pladere, 2023).

In summary, the combination of improved depth perception as well as improved interaction makes VR a promising technology to visualize medical images. However, it also presents some notable limitations, such as limited FoV and vergence-accommodation conflict, that have to be taken into consideration when designing applications for VR.

1.5 Perceptual improvements in medical visualization

Now, we will present the previous work that has been done on improving the visualization of medical images. A wide range of techniques has been developed with the aim of enhancing the perception of 3D medical scans. Some of these techniques focus more on depth perception when visualizing cerebral angiographic CT images, while others focus on occlusion management when visualizing CT images that present significant self-occlusion.

1.5.1 Depth Perception

Depth perception is especially difficult when visualizing cerebral CTA images, due to the complicated branching and overlapping that this type of data usually presents, as well as a high variation between patients (Kersten-Oertel *et al.*, 2014). To overcome these problems, multiple depth perception techniques were proposed, and numerous user studies have been performed to assess their efficiency in conveying depth information.

One approach that is commonly used to improve understanding of the medical dataset is by using color and contrast to convey some of its characteristics. In the volume rendering pipeline, this is typically done in the classification stage with a transfer function. The transfer function maps data properties at a certain sampling location (such as the voxel intensity or the gradient) to a certain RGB color and opacity α (Abhari, Baxter, Eagleson, Peters & Ribaupierre, 2012). Transfer functions are typically composed of piecewise linear 2D functions manually defined by the viewer (Preim & Bartz, 2007, p. 142). Later during rendering, as it was previously explained in the volume rendering section, all the $RGB\alpha$ colors computed with the transfer function within a single ray are composited together to create the final color of the pixel (Callahan *et al.*, 2008). An example of an opacity and color transfer functions with the corresponding visualization created with the 3D Slicer application (Fedorov *et al.*, 2012) can be seen in Figure 1.16.

Transfer functions could be used to help understand the geometry of the visualized dataset. Among the most widely used approaches are Phong shading (Phong, 1975) and its refinement, the Blinn-Phong model (Blinn, 1977), which extend the basic principles of light reflection, which we introduced previously. Both shading models are able to convey the structure of the surface that is shaded, by allowing to manually specify the ambient, diffuse, and specular components for the structure that is shaded (Blinn, 1977). Additionally, Phong and Blinn-Phong shading models are focused on being photorealistic (Phong, 1975), so viewers could apply knowledge about how lighting works in real life to understand the geometry on the digital image. The drawback of these shading models is that they require the gradient (normal vector) to be computable every time when the transfer function is called. Although this approach works well for surface rendering,

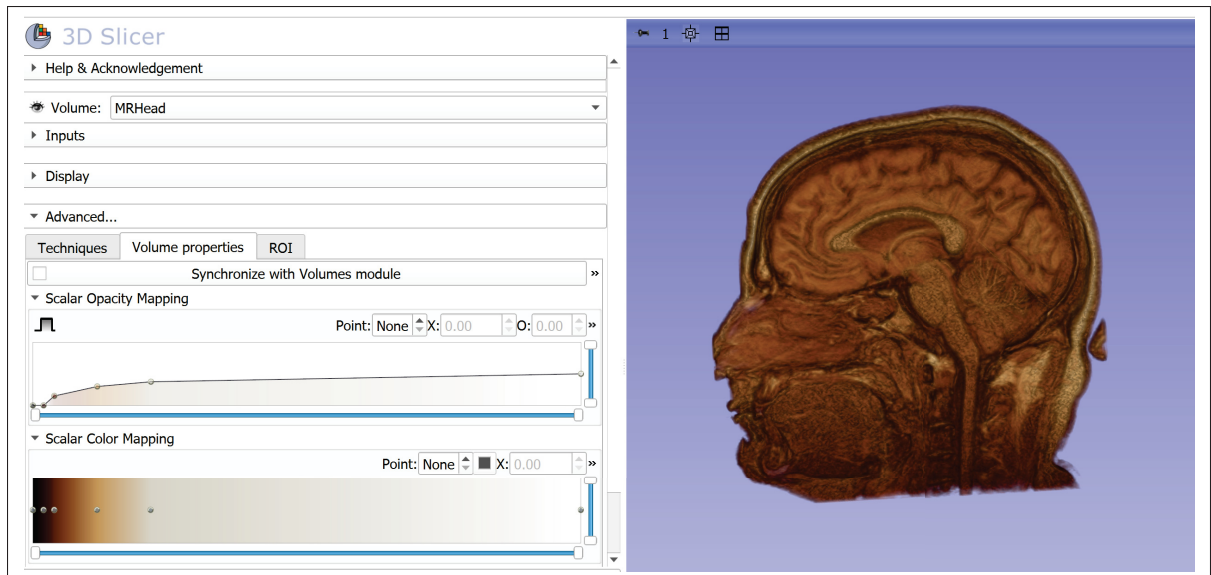


Figure 1.16 Interface of 3D Slicer (Fedorov *et al.*, 2012) demonstrating opacity and color transfer functions. These functions map volumetric data values to visual properties, enabling the interactive highlights of specific internal structures within a medical scan

in the case of volume rendering, the gradient is only defined at sharp intensity transitions and cannot be computed within a homogeneous subset of the dataset (Hernell *et al.*, 2010). Moreover, since gradients are calculated using the intensity values, in the case of noisy data, the gradients become noisier.

There exist other shading techniques that try to convey information about the volume by different means than photorealism. For example, some techniques try to help with the depth perception, by color-encoding depth (Ropinski *et al.*, 2006). The most well-known techniques of this kind are chromadepth (Bailey & Clark, 1998), pseudo-chromadepth (Ropinski *et al.*, 2006), and aerial perspective (also called fog) (Preim *et al.*, 2016). These techniques use color to represent the relative depth from the view plane to different parts of the object. In chromadepth, depth is encoded using colors of the visible light spectrum, with red color for the closest parts, blue color for the furthest ones, and orange, yellow, green, and cyan for intermediate depths. Pseudo-chromadepth limits color-encoding to only two colors, red and blue, and linearly interpolates between them for intermediate depths. Aerial perspective encodes color similar

to how fog works in real life, meaning that the closest parts are shaded so they are highly distinct from the background, while further parts lose their contrast, fading into the color of the background. Their effectiveness in conveying depth was extensively evaluated in the literature.

Kersten-Oertel *et al.* (2014) performed a user study where multiple depth cues were tested in terms of how much they help to visualize cerebral vasculature. Two user studies were performed, one by novices and another by experts. The cues that were tested were "no" cue (that displayed a solid red color), edge enhancement, kinetic depth, pseudo-chromadepth, stereopsis, as well as fog. Additionally, combinations of these cues were tested, but only by novices. The experimental task was that the user was presented with angiographic scans where two vessels were marked. The participants had to determine which of the vessels was closer to them, in terms of depth. This task was repeated among many trials, each featuring a different depth cue. These cues were compared in terms of correctness (corresponding to whether the participants actually selected the closer vessel), time to make a decision about depth, as well as user feedback. Among single cues, pseudo-chromadepth had the best results overall for both groups, both in terms of correctness and time, while the edge enhancement cue was found to be more helpful to experts. Among the combined cues, the best combination was pseudo-chromadepth combined with stereopsis, as well as the same combo with edge enhancement added on top of the previous two cues. For all cues, it was determined that both a large xy distance (distance from the vessels on the 2D screen), as well as a large z distance (in terms of depth) increase correctness. In terms of user feedback, pseudo-chromadepth was ranked the best, while in terms of experts, fog was ranked the best.

Lawonn, Luz, Preim & Hansen (2015) introduced an illustrative visualization technique that enhances depth perception when visualizing vascular structures, composed of three sub-techniques: illustrative shadows, supporting lines, and depth-dependent contours (see Figure 1.17). In illustrative shadows, a plane is put under the visualized volume, which displays the shadows of the vessel structure as if a light source was placed on top of this volume. In the supporting Lines technique, user of the system can select a point on the surface of the vasculature, and a line would be drawn from the selected point to its projected point on the plane. Additionally, a ruler-like quad will be drawn next to the line, allowing for measuring

the distances. For the depth-dependent contour, a halo is rendered around the vessels in the volume whose width is inversely proportional to the distance to the camera. A combination of all these techniques was compared to Phong shading (1975) and chromadepth in a user study. The author's illustrative method resulted in the highest accuracy and user confidence, while pseudo-chromadepth ranked second in both measures. However, regarding decision speed, pseudo-chromadepth was the fastest, followed by Phong shading, and lastly the illustrative method.

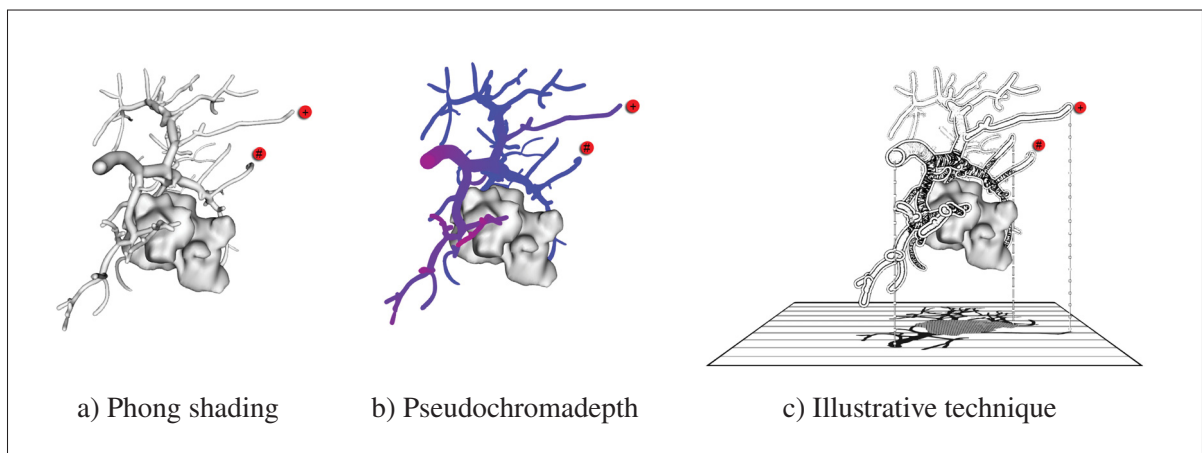


Figure 1.17 Comparison between the Phong, pseudo-chromadepth and the illustrative technique proposed by Lawonn *et al.* (2015)

Adapted from Lawonn *et al.* (2015). Reproduced with permission from Springer Nature.

In a separate study, Lawonn, Luz & Hansen (2017) proposed another illustrative method using anchor cues to show relative vessel depth on the screen. Unlike the previous method that required user input to position the supporting lines, this method automatically determines the best position of the supporting anchors that illustrate the depth of the vessels. The performance of this technique closely matched that of the previous illustrative methods. In the user study, it achieved the best accuracy and confidence, followed by pseudo-chromadepth and then Phong shading. In terms of decision time, pseudo-chromadepth remained the fastest, with Phong next and the illustrative anchors last.

Drouin, Giovanni, Kersten-Oertel & Collins (2020) presented a visualization technique where an interactive aspect was added to some of the previously described shading-based techniques. The

general idea of this approach is that a handheld surgical pointer was used to dynamically change the transfer function, which allowed for better understanding of the data. This technique was primarily developed to overcome the sterility issues within an operating room and the fact that surgical pointers are already used extensively during surgery. Furthermore, since this technique provides more DoFs than classical input devices such as the mouse and keyboard, it could be helpful for surgical planning.

Drouin *et al.* (2020) suggested a dynamic version of three well-known visualization techniques and compared them with each other, as well as with their static counterparts. The first technique is called shading, and it uses the Blinn-Phong illumination model to shade the volumetric dataset. The static version of this model has a light source fixed at the location of the eye, while the dynamic version has the light source located at the tip of the surgical pointer and thus can be moved, as illustrated in Figure 1.18. The second technique is pseudo-chromadepth, which combines shading with the classical red-to-blue depth color coding. While the static version spreads the color gradient across the full length of the volume, the dynamic version allows for dynamically changing the interval in which the gradient is located using the tracked pointer. The third visualization technique is aerial perspective, and similar to chromadepth, it allows one to either see the full gradient in the static version or reduce the interval in the dynamic version.

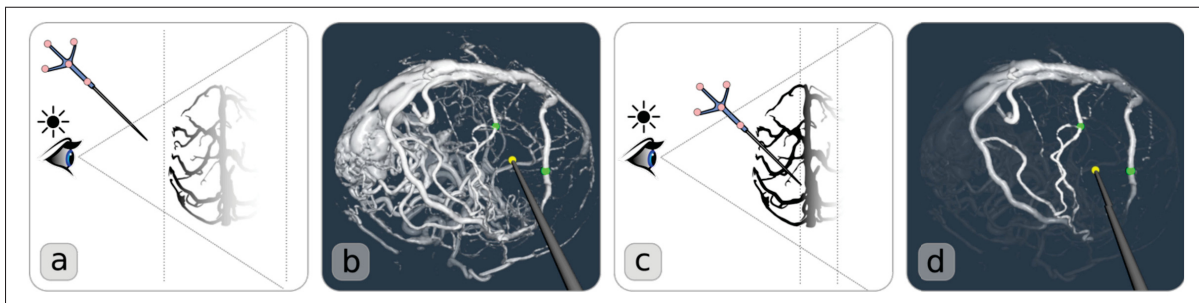


Figure 1.18 Angiographic cerebral medical scan visualized using static and dynamic shading. The physical setups of static and dynamic shading are illustrated in (a) and (c), respectively, while the resulting images for the static and dynamic variants are shown in (b) and (d).

Reprinted with permission from Drouin *et al.* (2020). © 2020 IEEE

The new interaction techniques were validated with a user study where users were presented with two targets identified in the volume, and they had to decide which one was closer to them. They had to position the tip of the pointer as close as possible to the chosen sphere without actually touching it. It was determined that on average, participants had better accuracy with dynamic cues compared to static cues, and thus, dynamic cues provide a better understanding of the local structures. However, dynamic methods did not allow for better appreciation of global structures (there was no significant difference). In terms of the trial time, dynamic cues performed generally worse than their static counterparts, with only dynamic shading not increasing the trial time. In terms of the difference between the 3 cues, shading had a significantly worse accuracy than fog and chroma, and thus resulted in a worse appreciation of the local structures. There was no statistically significant difference between the cues in terms of global appreciation (correctness). In a post-study questionnaire, subjects said that they thought that static and the dynamic version of chromadepth allowed for the best results, which was consistent with the objective results of the study. However, an important discrepancy was noted between the subjective perceived effectiveness of the dynamic cues and the objective results of the study. Participants have judged that the dynamic versions of the cues allowed them to better determine the closest target (thus, have a better correctness), although the objective measures don't suggest that correctness was improved in a statistically significant manner. Also, a discrepancy was found between the perceived and objective performance for the dynamic shading cue in comparison to static and dynamic chromadepth. Although dynamic shading resulted in worse objective results than chromadepth, participants didn't perceive their performance to be worse. Drouin *et al.* (2020) suggested that this difference might be explained by a positive bias that humans have toward dynamic shading because of its natural aspect.

Finally, some techniques have been proposed that focus more precisely on the shading aspect during the visualization to better convey depth. One such shading model was proposed by Hernell *et al.* (2010), which tries to solve the problems of the Phong and Blinn-Phong shading models when used in volume rendering. They proposed an ambient occlusion technique called Local Ambient Occlusion (LAO), which is adapted to work well with ray casting. It tries to overcome

the problem of computation of gradients inside a volume, and it does so by modifying how the ambient component (and optionally the diffuse component) are calculated. An illustration of LAO when used with a cerebral medical scan is provided in Figure 1.19, where it is compared with standard diffuse illumination.

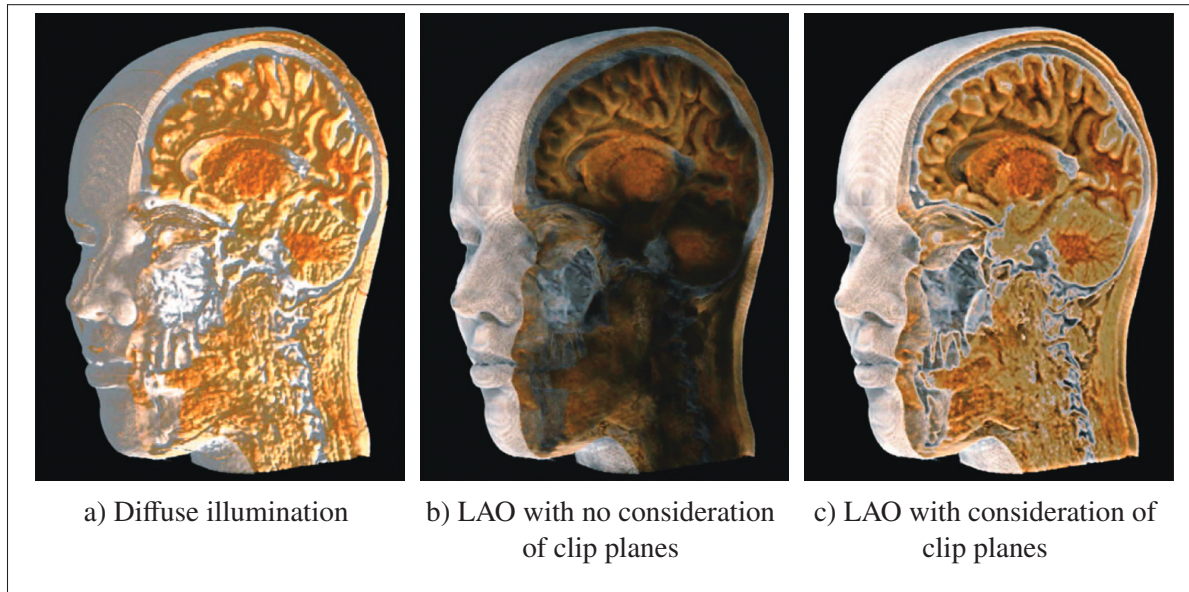


Figure 1.19 Cerebral medical scan visualized using standard diffuse shading (a), LAO where the occlusion was not adapted to the clipping plane and (c) LAO where the clipping plane was taken into account

Reprinted with permission from Hernell *et al.* (2010). © 2010 IEEE

The general idea behind the algorithm is that when performing ray casting, the neighboring voxels are taken into account to determine the occlusion for each sampling point, which affects the ambient component in the transfer function (Hernell *et al.*, 2010). For every sample taken in the volume, short rays are sent in all directions in a spherical manner, and the average value of all rays is then used to determine the level of occlusion. The rays use the standard volume rendering formula to determine the occlusion in a certain specific direction. The implementation for this algorithm uses a multiresolution dataset, which separates all data into blocks of 16^3 voxels. Each block in this texture is mapped to packed texture coordinates using a forward mapping index function. Due to the complexity of the algorithm, only one LAO ray is calculated at each frame, and all the results are then blended together.

The advantages of LAO are that it emphasizes both the structure and the relative distance between two overlapping objects, as well as the density of the material using light dispersion (Hernell *et al.*, 2010). Additionally, it is less sensitive to noise compared to Phong shading since it does not need to calculate the normal vectors, which are highly affected by noise when calculated in a 3D volume. The drawback of LAO is that, unlike diffuse shading, it does not accentuate the edges in the case of a sharp change of gradient. Thus, instead of replacing the diffuse component, LAO could also be used only for the ambient component calculation to avoid this problem.

1.5.2 Occlusion Management

Another important problem that happens when visualizing medical images is the effect of occlusion, which happens when we attempt to visualize parts of the body that contain many anatomical structures, such as organs or tissues. This is sometimes referred to as the problem of self-occlusion (Elmqvist & Tsigas, 2008), which happens in any scene where there are too many objects per volumetric unit, and is true for the human body. Occlusion management (Elmqvist & Tsigas, 2008) is an umbrella term that encompasses all the techniques that deal with this problem. A multitude of them have been developed precisely in the context of medical visualization.

We propose to divide the existing occlusion management technique into two categories: slicing/clipping-based techniques whose goal is to show a smaller subset of the dataset, and deformation-based techniques that perform linear or nonlinear transformations on the volumetric data.

1.5.2.1 Shading-Based Techniques

1.5.2.2 Slicing/Clipping-Based Interaction Techniques

Slicing or clipping techniques aim to allow the user to see the internal parts of the data that would otherwise be hidden if the entirety of the volume was rendered simultaneously.

Slicing is implemented by sampling one or more planar slices and then displaying them in 2D (Joshi *et al.*, 2008). The most basic type of slicing is axis-aligned slicing (AAS). In this technique, the splitting planes are defined to be perpendicular to the three default axes in 3D. Each plane samples the volume, and the computed 2D slices are then displayed to the viewer. The created 2D slices typically correspond to the axial, sagittal and coronal views of the patient (Laha & Bowman, 2013). Although this technique is popular and used extensively in medical software applications such as 3D Slicer and Amira (Laha & Bowman, 2013), it has an important disadvantage. Since the visualization is limited to 2D slices, it makes it difficult to understand the 3D structures of anatomical regions because they are often not aligned with any geometric shape (Preim & Bartz, 2007, p. 155). This is especially true for elongated structures such as electrodes (Joshi *et al.*, 2008) or blood vessels.

On the other hand, the goal of clipping is to hide parts of the dataset in order to visualize a smaller subset of the medical dataset (Preim & Bartz, 2007, p. 291). The data is then not only easier to understand because of the smaller number of voxels that have to be visualized simultaneously, but also because external layers of the data may occlude the internal parts of the dataset. Thus, removing those external layers may reveal some interesting internal layers that wouldn't be visible otherwise (Drouin, Collins & Kersten-Oertel, 2018; Joshi *et al.*, 2008).

Clipping typically refers to the procedure where cutting planes are inserted into the 3D dataset, subtractively cropping parts from the initial volume. The advantage of such a procedure is the ease of implementation (Preim & Bartz, 2007, p. 292). One of the simplest implementations of clipping is axis-aligned cropping, but it has the disadvantage that the mapping between the 2D slices and an arbitrarily oriented volume may require significant mental effort to perform (Joshi *et al.*, 2008). Additionally, since the visualization is limited to 2D slices, it may crop too much of the surrounding region, which is true even for non-axis-aligned clipping implementations. Thus, more complex clipping procedures were developed where non-planar shapes are removed from the volume. Such a subset of clipping is called carving (Drouin *et al.*, 2018) or cropping (Joshi *et al.*, 2008).

Slicing and clipping are different from segmentation, even though all of these techniques allow cropping parts of the data in order to better understand the remaining subset. Segmentation takes into account the data provided in each voxel, and usually has the goal of separating different types of tissue from each other (Preim & Bartz, 2007, p. 95). Conversely, slicing and clipping don't take into account the nature of the data, and the removed region is rather selected manually by the viewer.

Joshi *et al.* (2008) explored the use of irregular cropping for both operating room use and surgery planning in the domain of neurosurgery. They created a neurosurgical visualization system where the viewer could slice parts of the volume in the form of three-dimensional shapes to see the internal parts of the brain. The following shapes could be used to crop the volume: cube, sphere, cylinder and ellipsoid. When such a shape was inserted in the brain, the voxels that were inside of this shape were marked as clipped and were not rendered during the ray casting stage. Clipping parts of the brain allowed one to better view the positions of the electrodes which were inserted in the brain during the surgery. The position of the shape is defined by a tracked probe that allows 6 DoF, while the rendering was done on a regular 2D screen, producing a result as illustrated in Figure 1.20.

The advantage that irregular cropping has over axis-aligned slicing is that it preserves more context when visualizing the data by allowing to crop only a small subset of the visible data. One of the problems that experts had while using the system is that because of the complex nature of the brain dataset, when spherical cropping was performed, the clipping could not always be easily distinguished. Because of that, depth cues had to be introduced in the rendering such as distance color blending and shadows.

Chiang, Chen & Hsia (2019) presented a hand-tracked focus-plus-context clipping technique which is similar to the one presented by Joshi *et al.* (2008). In this method, one hand was responsible for planar clipping, and the position and rotation of the cutting plane were controlled by this hand. At the same time, the second hand could point towards a region of interest in the volume, and then that region will be carved, similar to how it was done by Joshi *et al.* (2008),

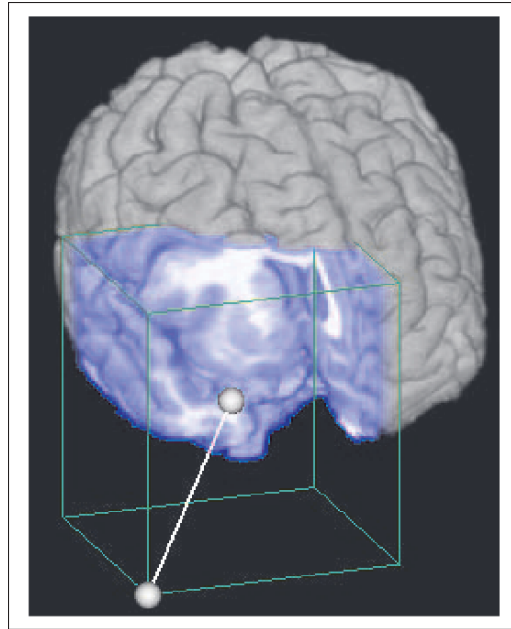


Figure 1.20 Volume carving
with a cubic clipping shape on a
cerebral medical scan
Reprinted with permission from
Joshi *et al.* (2008). © 2008 IEEE

using a cylinder or cone shape. What is particular in this method is that various gestures were used to control different parameters of the rendering, with 6 different gestures recognized in total.

1.5.2.3 Deformation-based interaction techniques

McGuffin *et al.* (2003) proposed a solution to this problem by providing the viewer with a set of deformation tools that could be used to better understand the data. He created a medical visualization system which allows to view and manipulate a volumetric dataset. The interface of this system allows the viewer to select a region in the volume, change the transfer function and deform or spatially transform the selected region. A total of 5 tools to manipulate and transform the data were presented to the viewer. Each tool can be applied to either the whole volume or

only certain layers. Every type of manipulation is animated, meaning that there is a smooth transition between different states.

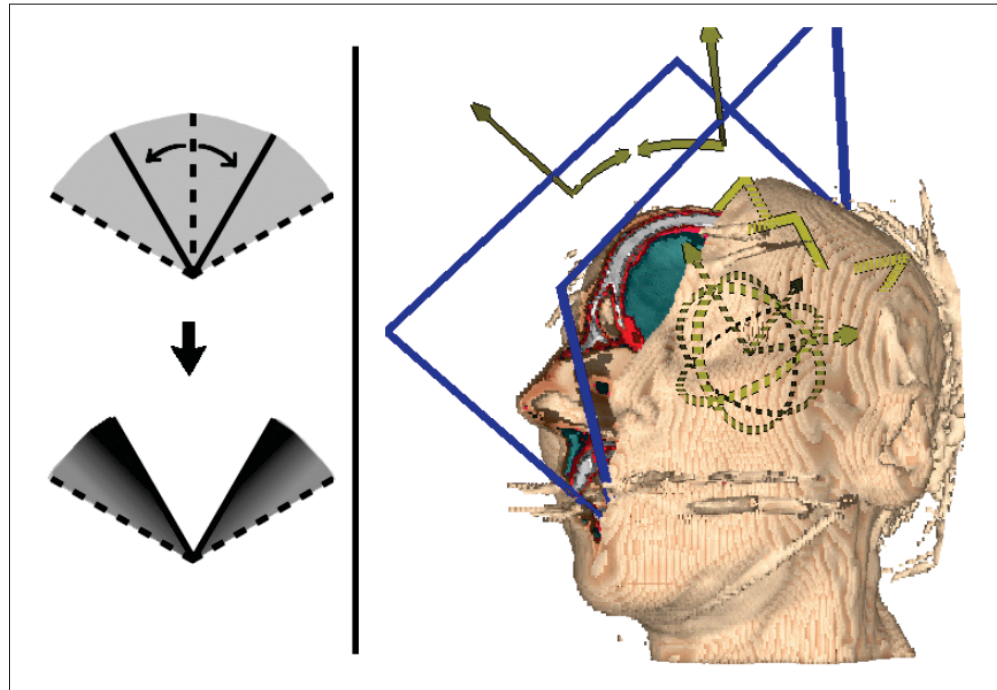


Figure 1.21 Hinge Spreader deformation tool. This interaction technique splits and rotates volumetric data to reveal internal anatomy while maintaining surrounding spatial context
Reprinted with permission from McGuffin *et al.* (2003). © 2003 IEEE

The Hinge Spreader (see Figure 1.21) allows making a deep cut into the volume and then spreading the volume to the sides, like a hinge, compressing the voxels. This allows seeing the internal voxels that were hidden by external layers. The Sphere Expander tool allows pushing voxels away in all directions from a central point, i.e., spreading the voxels from the center of the sphere. Since every voxel is rendered as a point, employing this technique results in creating holes between the voxels, which creates some sort of transparency. The Box Spreader tool works similarly to the Hinge Spreader, except that after making a deep cut, both regions of the volume are translated apart instead of being rotated. Thus, this tool does not result in any deformation of the volume. The Leafer tool works by first placing a tray-like plane in the volume, after which each half of the tray can be hinged away. However, unlike the Hinge Spreader, the voxels are

rotated without being compressed, so no deformation is happening. The last tool is called the Peeler and it also works by positioning a tray and rotating the volume on one side of the tray, but unlike the Leafer, the voxels here are compressed. A variation of this tool was implemented where the volume is peeled in a circular manner from the center, like a flower that is opening its petals.

The advantage of these deformation-based techniques is that they allow viewing of the hidden interior parts of the dataset, while keeping track of the relative position between the elements. This is an important advantage compared to transparency-based techniques, which often result in a loss of context if no focus-plus-context technique is used (Lagos & McInerney, 2019).

One of the drawbacks of McGuffin *et al.* (2003) visualization system was the lack of DoFs during interaction, caused by the use of default mouse and keyboard as input devices. As it was noted by McGuffin *et al.* (2003), multiple widgets had to be displayed on the screen in order to allow performing all the possible manipulations, such as switching between the deformation techniques, placing and rotating the splitting planes and performing the deformation animations. Another limitation is that the volume that is visualized has to have some sort of segmentation in order for some of the deformation methods and their animations (like the Leafer) to work properly.

In an article published in 2013, Laha & Bowman (2013) designed a gesture interaction technique called Volume Cracker, that took advantage of the deformation techniques presented by McGuffin *et al.* (2003). Similar to McGuffin *et al.* (2003), the goal of this new interaction technique was to allow for better understanding of three-dimensional projected data by using deformations. However, instead of using a classical computer mouse, Laha & Bowman (2013) used a bimanual asymmetrical input technique. The general idea of Volume Cracker was that a splitting plane was located at the mid-point between the two hands, and performing a grabbing and separating gesture with the hands resulted in separating the volume into two pieces. On the left hand, the user used a hand tracker and on the right hand, they held a wand. The left hand was generally

used to hold the volume in VR, while the right hand was used to perform manipulations on this volume.

The interactions that were possible to perform using this technique are the knife, the cracker, the peeler and the hinge spreader, which were described by McGuffin *et al.* (2003). All these techniques were only cutting or spatially transforming the data and were not deforming it (that is, they were not changing the distance between voxels within the same sub-volume).

When the volume was separated into multiple parts, a line that connects the two volumes was rendered to indicate how each volume was divided. However, the possible manipulations with the separated volumes were limited to avoid confusion. The movement of the divided volumes was limited to be along a plane and the rotations were limited to be around the normal of the plane.

A user study was conducted with 12 participants who had to perform two tasks: one on searching and another on pattern recognition. In the search task, letters of the English alphabet were inserted in the volume with arbitrary position, rotation and thickness, and participants had to find all occurrences of a certain letter that was repeated more than once in the volume. In the pattern recognition task, participants were presented with three complex shapes inserted in the volume, and participants had to find the odd shape, which was different from the other two.

When compared to the 2D axis-aligned slicing technique and the arbitrary slicing gesture technique, Volume Cracker led to better results for the search task, but resulted in a worse decision time for the pattern recognition task.

In 2016, Laha *et al.* (2016) proposed an alternate version of the previously described Volume Cracker algorithm where the tracked glove and controller were replaced by a bare-hand interaction technique using a Leap Motion controller. Their reasoning was that although VC showed some good results, the handheld controllers were impractical and limited the accessibility of the algorithm.

The authors presented two variants for the handheld algorithm, one of which was symmetrical and the second one was asymmetrical. In both approaches, a Leap Motion device was used to track the position of the free hand(s). In terms of rendering, no virtual or augmented reality visualization was used in either case, the participants were viewing the volume on a 2D monitor.

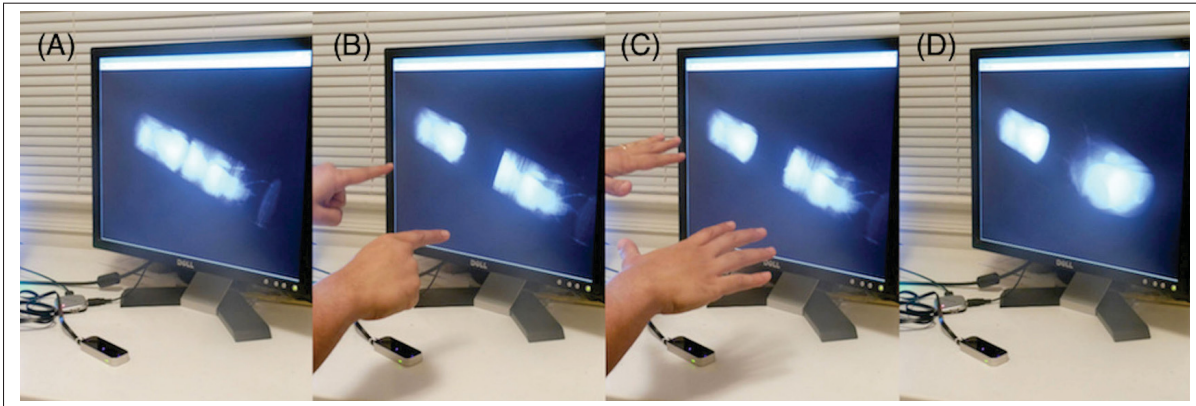


Figure 1.22 The symmetrical bare-hands "Volume Cracker". (a) demonstrates the full original volume, (b) demonstrates the preview of the split, (c) demonstrates the actual cracking gesture and (d) demonstrates the resulting split volume
Taken from Laha *et al.* (2016), *Frontiers in Robotics and AI*, CC BY 4.0

For the symmetrical approach (presented in Figure 1.22), the viewer could separate the volume by putting their open palms in front of them and then moving their hands away from each other to separate the volume. They could additionally use a single pointed finger gesture on both hands to preview the result of a volume split. After the volume was split, the viewer could rotate their hands to preview parts of the volumes from different angles. For the asymmetrical approach, the Space Navigator Pro controller was used for the left hand instead of a bare hand. The idea behind such an asymmetrical control mechanism was that the Leap Motion tracker does not provide reliable 6 DoF hand tracking, while the controller does. The right hand (that was still tracked by the Leap Motion tracker) hovered around the volume, attracting the voxels close to it, which indicated how the volume would be split.

The bare-hands volume cracker (BHVC) was compared to axis-aligned slicing (AAS) and arbitrary slicing (AS) methods with a user study with 14 participants. The participants had to perform 10 different tasks, which could roughly be divided into three categories: naive search

tasks, counting-based search tasks, shape description tasks and spatial judgment tasks. BHVC resulted in the best accuracy, but required the highest amount of time to perform the tasks due to the number of presented features. Although BHVC was not compared to VC using a user study, the authors noted that the original VC method is less constrained and allows for more precise manipulation, but BHVC is less cumbersome.

For the volume splitting techniques described by Laha & Bowman (2013) and Laha *et al.* (2016), no segmentation was necessary to perform transformations on the volumes. Although these techniques didn't feature cerebral anatomic data, they could potentially be used on unsegmented MRI cerebral scans. For the deformation techniques presented by McGuffin *et al.* (2003), some of them such as Hinge Spreader or Leafer could be used even with unsegmented cerebral MRI data, while others such as Peeler or Sphere Expander would require segmentation. In fact, if the cerebral scan is segmented, a different set of interaction techniques could be explored.

He, Guayaquil-Sosa & McGraw (2017) presented a medical data visualization technique that helps in viewing pre-segmented brain parts that are organized in a hierarchical structure. The exploration of the data was performed in VR using handheld controllers. The location of the 3D controllers was generating points that were then used to compute a Hermite spline curve. These splines were used to define the trajectory of the movement of the segmented parts of the data. The trackpad could then be used to reparametrize the curve, which changed which segments are presented in an exploded manner, and which ones are not. The computed splines also kept track of the orientation of the volume. While moving, the trajectory was optimized so that segments rotate as little as possible. The possible deformations of the dataset were all linear, and they were inspired by those described by McGuffin *et al.* (2003): linear explosion, leafing, fanning and popping out a single slab.

As this technique requires fine-grained pre-segmentation and organization into a hierarchical structure, it may not be well applicable for preoperative planning, as performing those tasks well on a dataset that contains a pathology (like a tumor) may require manual segmentation, which requires significant time and effort (Preim & Bartz, 2007, p. 96). However, as it was determined

by He *et al.* (2017), this technique is useful for learning the brain anatomy. It increased both the learning speed and the certainty of naming the different brain parts.

CHAPTER 2

OBJECTIVES AND METHODOLOGICAL APPROACH

Effective visualization of medical images relies heavily on both depth perception and interactivity. Depth perception is particularly important when examining elongated anatomical structures, such as vasculature visible in computed tomography angiography (CTA) scans. Conversely, interactivity plays a key role when exploring volumetric datasets that exhibit a high degree of self-occlusion. To address these challenges, this study focuses on the use of virtual reality (VR) headsets, as we hypothesize that the enhanced depth cues and interactive capabilities they provide can facilitate more intuitive interpretation of medical images and improve their utility in tasks such as diagnosis, preoperative planning, and surgical guidance.

This thesis investigates the potential of modern virtual reality (VR) technology in the field of medical imaging, with a focus on identifying its strengths for visualizing anatomical data. Specifically, we address two primary challenges in medical image interpretation: depth ambiguities in angiographic image visualization, and self-occlusion in volumetric MRI and CT datasets, both segmented and unsegmented.

The main objective in this thesis is subdivided into three specific goals as listed below:

1. To investigate, in the context of cerebral angiographic data visualization, how depth cues typically employed on standard 2D displays translate to a VR environment.
2. To evaluate how interactive clipping, when combined with custom rendering techniques, can enhance the visualization of unsegmented volumetric data that exhibits significant self-occlusion in VR.
3. To develop a VR-based occlusion-management method for segmented volumetric data, leveraging interactive clipping to address challenges arising from self-occlusion.

Each of these objectives resulted in a dedicated scientific publication.

The first study was initially presented at the Augmented Environments for Computer-Assisted Interventions (AE-CAI) workshop, held in conjunction with the Medical Image Computing and

Computer-Assisted Interventions (MICCAI) conference, and was later published in the journal *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization* (Titov, Kersten-Oertel & Drouin, 2022). Its contents are presented in Chapter 3, which focuses on depth perception in cerebral angiographic images. This chapter reports on a user study designed to assess whether the depth cues typically used on standard 2D displays remain effective when presented through a VR headset. Both static and dynamic cues were evaluated, differing in their level of interactivity, and the results were analyzed statistically.

The second study was first presented at the Graphics Interface conference, with an extended version subsequently published in the journal *Computers & Graphics* (Titov, Kersten-Oertel & Drouin, 2024). It is detailed in Chapter 4, which introduces a new volumetric rendering technique that enables a high-quality yet traditionally slow ambient occlusion method to perform efficiently in real time within a VR environment. Building on the concept of local ambient occlusion, normally too computationally expensive for real-time applications, we proposed a variant that significantly reduces computational cost, making it suitable for integration with interactive clipping of medical volumes.

The third study has been submitted to the *IEEE Transactions on Visualization and Computer Graphics* (TVCG). It is presented in Chapter 5, where we introduce *AnatomyCarve*, a novel interactive occlusion-management technique for VR that enhances the visualization of segmented volumetric data. The algorithm supports segment-aware clipping of anatomical structures within a medical volume, allowing the creation of visualizations similar to those found in anatomical atlases. The usability of *AnatomyCarve* was evaluated in a user study with non-experts, while its potential for preoperative planning was assessed through feedback from practicing surgeons and residents.

CHAPTER 3

THE EFFECT OF INTERACTIVE CUES ON THE PERCEPTION OF ANGIOGRAPHIC VOLUMES IN VIRTUAL REALITY

Andrey Titov^{1,2}, Marta Kersten-Oertel², Simon Drouin¹

¹ Department of Software and IT Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

² Gina Cody School of Engineering and Computer Science, Concordia University,
1455 De Maisonneuve Blvd. West, Montréal, Québec, Canada H3G 1M8

Article published in the journal "Computer Methods in Biomechanics and Biomedical
Engineering: Imaging & Visualization" in November 2021

3.1 Introduction

Interaction is an important aspect in medical data exploration and one that has not been sufficiently studied despite the complex nature of medical scans and the limitations of 2D displays (McGuffin *et al.*, 2003; Drouin *et al.*, 2018). It has also been shown that recognizing objects from different angles is easier if the angle change results from self-movement (Simons, Wang & Roddenberry, 2002) or active user input (Harman, Humphrey & Goodale, 1999) rather than from passive object movement (e.g. a rotating scan). Thus, virtual reality (VR) environments where the user can explore complex data by actively interacting with it offer a promising solution for medical data exploration.

Three-dimensional cerebral angiographic scans, which are used for diagnosis, neurosurgical planning and intraoperative guidance (Abhari *et al.*, 2012), are particularly complex due to the intricate branching and frequent overlapping between vessels. This makes these types of data scans particularly difficult to visualize and understand spatially (Kersten-Oertel *et al.*, 2014). Researchers have studied the use of various perceptual cues, illustrative methods and interaction techniques (Ropinski *et al.*, 2006; Preim *et al.*, 2016) to improve depth perception and spatial understanding of 3D vascular data. However, most of this work has targeted classic 2D monitors for displays coupled with keyboard and mouse for interaction. This setup suffers from a limited

capacity to represent visual information and a lack of degrees of freedom for interaction and thus may not be optimal for exploring angiographic datasets (Heinrich *et al.*, 2020).

With the recent advancements in VR hardware and controllers, new human-computer interaction paradigms have become possible. Modern VR headsets allow for *binocular disparity* (slight differences between the right and left eye images that allow an observer to see 3D) and *motion parallax* (a change in position of an object caused by the movement of an observer), which provide strong cues for depth perception (Vienne, Masfrand, Bourdin & Vercher, 2020). A recent study by Heinrich *et al.* (2021) showed that VR significantly improves perception of depth order with angiographic data. Additionally, efforts are being made to develop varifocal VR displays to improve depth perception even further (Toulouse *et al.*, 2019). Moreover, the input devices that can be used in VR such as handheld controllers or hand tracking devices allow more natural gestures and significantly more degrees of freedom (Laha & Bowman, 2013; Laha *et al.*, 2016), enabling more intuitive and natural exploration of 3D datasets.

In this paper, we revisit several of the cues previously studied for the visualization of angiographic scans using VR in the context of pre-operative planning, simulation and in general for anatomical exploration. It is important in VR that the surgeon uses the best performing depth visualization techniques to improve their perception (Heinrich *et al.*, 2021). In particular, we aim at determining whether the effectiveness of the selected depth cues is affected by the increased expressivity provided by VR devices. Building on the work of Drouin *et al.* (2020), we aimed to determine the impact of dynamic and interactive depth cues within a VR context. Thus, this work addresses the need for a better understanding of the interplay between interactive perceptual rendering and the impact it has on spatial understanding of volume-rendered medical data within VR contexts.

3.2 Related Work

Various perceptual visualization methods have been developed with the goal of improving the perception of angiographic data. The effectiveness of such techniques was extensively studied previously by performing user studies. The task of these user studies is often to determine the

depth relationship of specific vessels, and performance is typically measured using correctness and decision time. These studies have been typically done using a 2D monitor, and to the best of our knowledge, only one such study was done so far in a VR environment with a head mounted display (HMD).

Ropinski *et al.* (2006) performed a user study where they compared the effectiveness in depth perception of previously studied techniques such as edge enhancement, Phong shading, depth of field, stereoscopy, chromadepth, and pseudo-chromadepth. It was determined that pseudo-chromadepth performs better than full chroma, and is the best shading technique in terms of decision time. However, in terms of correctness, the overlaid edges technique was found to be better than pseudo-chromadepth. In a similar study, Kersten-Oertel *et al.* (2014) compared the effectiveness of kinetic depth, stereoscopy, edge, pseudo-chromadepth and fog for novices and experts. It was determined that pseudo-chromadepth generally resulted in the best decision time and correctness, similar to the results obtained by Ropinski *et al.* (2006). It was also found that the edge cue was more helpful to experts than novices. Kreiser, Hermosilla & Ropinski (2018) introduced the concept of Void Space Surfaces (VSS) that encode the depth of vessels in the scene using the surrounding background. They then compared the chromadepth and pseudo-chroma versions of VSS to directly applied versions of these cues (to the surface of the vessels) using standard Phong shading as the base case. Although directly applied cues and VSS resulted in similar correctness, they both performed better than Phong. In terms of decision time, direct cues performed better than Phong, while VSS performed worse. The results of this study regarding the effectiveness of pseudo-chromadepth slightly differ from those obtained by Ropinski *et al.* (2006) and Kersten-Oertel *et al.* (2014) in that pseudo-chromadepth was not found to be more effective than chromadepth.

Dynamic versions of these cues were studied by Drouin *et al.* (2020). They introduced a dynamic aspect to Phong shading, pseudo-chromadepth and fog where the user could use a surgical pointer to control certain rendering parameters. The new dynamic versions of cues resulted in a better understanding of the local structures but not global ones, when compared to their static counterparts. Additionally, it was found that dynamic pseudo-chromadepth and dynamic fog

had higher decision times compared to their static versions. In a post-study questionnaire, users said that they preferred pseudo-chroma and that dynamic cues allowed them to perform better in terms of time, although this was not confirmed by the time measures.

We are aware of only one study that has been done using a VR environment for angiographic data exploration. Heinrich *et al.* (2021) conducted a user study where the effectiveness of Phong, pseudo-chromadepth and fog was compared in a monoscopic desktop environment versus a stereoscopic VR environment. They found that the environment did not have a significant impact on decision time and that fog resulted in the quickest trials, while pseudo-chromadepth resulted in the longest ones. In terms of correctness however, VR resulted in the least amount of sorting errors, and participants were also more certain about their decisions compared to the 2D desktop environment. Another interesting finding was that the type of cue that was used had a lesser impact in VR than the monoscopic desktop environment. This suggests that VR offers powerful perception cues that allow for correct depth judgements regardless of the shading technique.

3.3 Materials and Methods

The effectiveness of three depth cues in VR was studied: shading, pseudo-chromadepth and aerial perspective, as well as the dynamic version of each of these techniques as described by Drouin *et al.* (2020). In the dynamic version, the user modifies the rendering parameters of the cue by moving a virtual 3D surgical pointer using the VR controller. Our goal was not only to identify which visualization methods provide the best spatial understanding in a VR environment but also to see if generally, adding an interactive aspect to the depth cues significantly improves depth and spatial perception of cerebral vascular volumes in VR. As in the study by Drouin *et al.* (2020), all cues were implemented using standard ray casting and shaded with the Blinn-Phong reflection model (Blinn, 1977). A video demonstrating the implemented visualizations is provided in the supplemental material.

3.3.1 Depth Cues

In this work, we studied shading, pseudo-chromadepth and aerial perspective cues and their dynamic counterparts. The reason behind this choice is that these cues have been extensively studied in 2D environments. Shading has often been used as the base case (eg. Abhari *et al.* (2012); Lawonn *et al.* (2017)), while pseudo-chromadepth and aerial perspective generally perform well in terms of correctness and decision time (Kersten-Oertel *et al.*, 2014). Furthermore, the dynamic component for these cues was previously proposed by Drouin *et al.* (2020) and studied in a 2D environment. We thus chose these cues to better understand the impact of dynamic perceptual cues in a VR environment.

Each cue was tested twice, in two different variants: static and dynamic. Static cues require no user input and are calculated automatically using the 3D position of the VR headset. In the case of dynamic cues, the position of the tip of the handheld pointer controlled by the user modifies some parameters in the rendering.

Shading

Shading is a photorealistic cue that simulates how an object reflects light when illuminated by a light source. One of the main advantages of this cue is that it is intuitive since it lets the viewer use knowledge about how objects are illuminated in real life. In the static version of shading, the light source is located at the midpoint between the two eyes, similarly to the light on a miner's helmet. This way, the volume is always fully illuminated, no matter the position of the head in the virtual environment (see Figure 3.1a). In the dynamic version of shading, a point light source is attached to the tip of the pointer. Thus, as the user moves the pointer around the volume the anatomy around the pointer tip is illuminated and other parts of the volume fall into shadows. In other words, the light source works similarly to a match or flashlight that illuminates an object in a dark environment (see Figure 3.1b). The light source has a linear decay in intensity, which provides an additional cue for localization of the pointer inside the volume. Shading was implemented using the Blinn-Phong reflection model (Blinn, 1977).

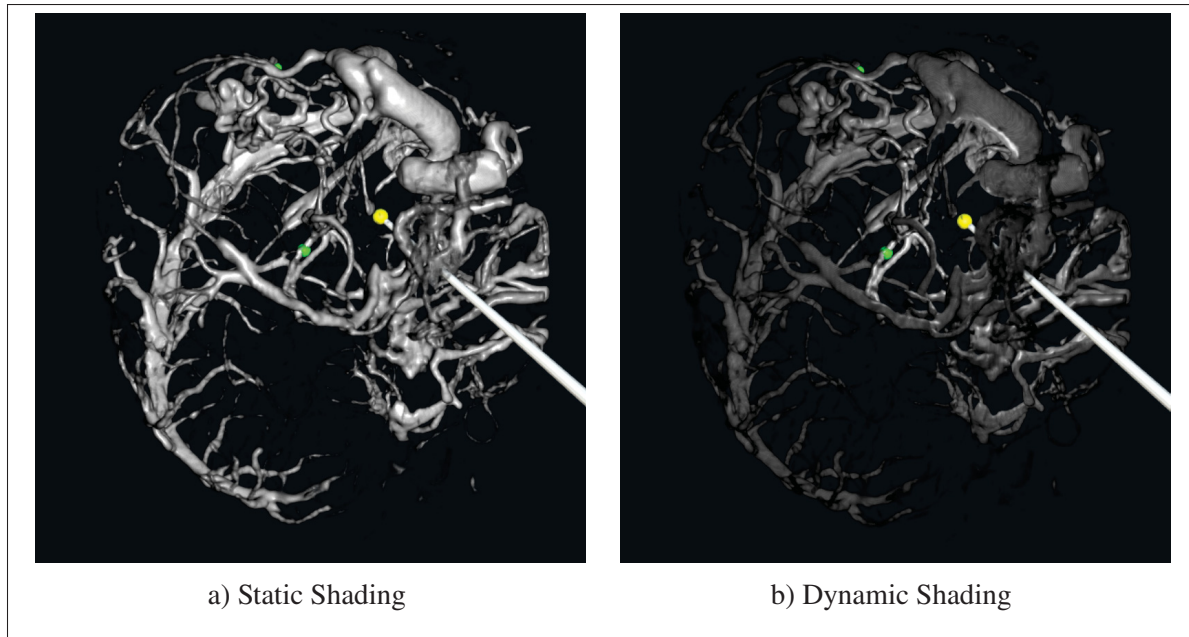


Figure 3.1 Static and dynamic shading. With static shading (a), the volume is fully illuminated from the point of view of the user, while with dynamic shading (b), the tip of the handheld pointer controls the position of the light source, whose intensity decays with further distances

Chromadepth

Chromadepth is a non-photorealistic depth cue that uses color to encode depth from the position of the viewer. The most popular chromadepth techniques are full chromadepth presented by Bailey & Clark (1998) and two-color (red to blue) chromadepth presented by Ropinski *et al.* (2006), also called pseudo-chromadepth. Kersten-Oertel *et al.* (2014) and Ropinski *et al.* (2006) determined that pseudo-chromadepth generally leads to better depth perception, thus we used this version in our study. For brevity, we will further refer to pseudo-chromadepth as chromadepth or simply chroma. In the static version, the interpolation is defined by the closest and furthest visible parts of the volume and adjusted to follow the view direction, so that the closest parts to the viewer appear in red and the furthest ones appear in blue. The closest and furthest visible points of the angiographic volume were estimated using a bounding sphere whose diameter was calculated as the average between the physical measures of width, height and depth of the volume.

In the dynamic version, the tip of the handheld pointer defines the near plane which corresponds to the start of the chroma interval and the red color. The far plane is located at 5 *cm* from the near plane and corresponds to the end of the chroma interval and the blue color. Any part of the object closer than the close plane is shaded in red, while parts further than the far plane are shaded in blue. The distance between the near and far planes was defined manually using a trial and error approach, with 5 *cm* resulting in the best distance. Figure 3.2 illustrates static and dynamic chromadepth.

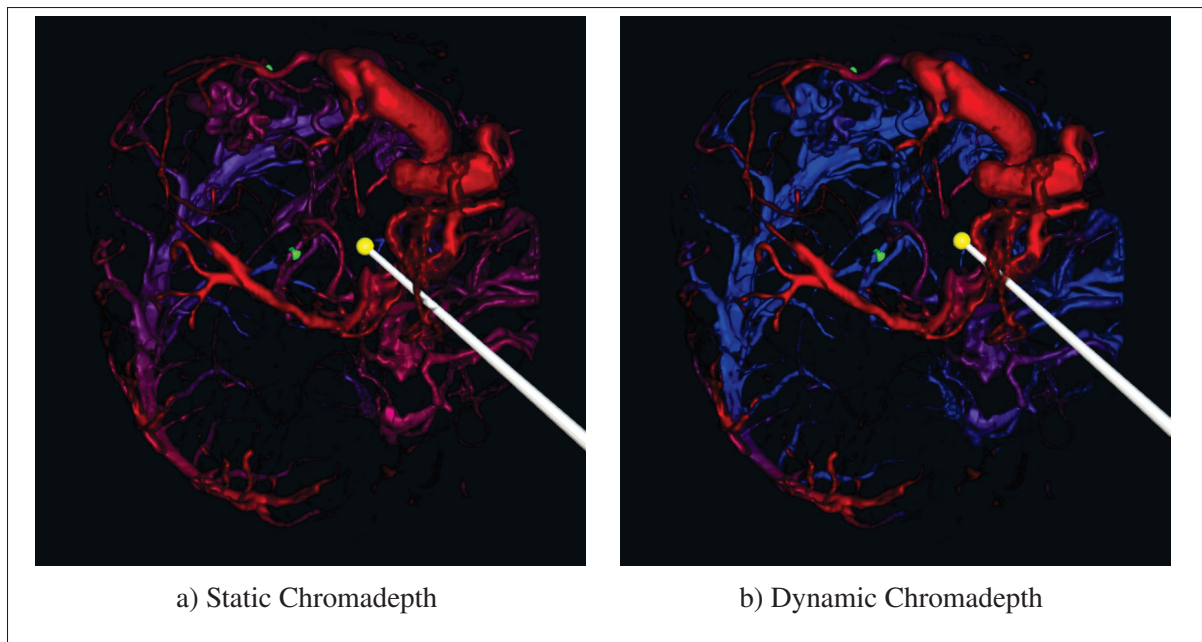


Figure 3.2 Static and dynamic chromadepth. In static chromadepth (a), the color-coded depth interval spans through the entire volume. With dynamic chromadepth (b), the tip of the handheld pointer positions the close plane, which marks the start of the color gradient. This gradient spans for a short predefined distance

Aerial Perspective

Aerial perspective encodes depth similar to the natural phenomenon of fog in real life, i.e. using a loss of contrast (Preim *et al.*, 2016). In other words, parts of the volume which are close are highly contrasted from the background and are simply shaded with Blinn-Phong (Blinn, 1977), while further points fade into the color of the background. The encoding of this cue is very

similar to chromadepth described previously. In the static version of aerial perspective, the interpolation is done across the full visible depth range of the volume. In the dynamic version, the user controls the depth at which the loss of contrast starts with the tip of the pointer. For brevity, we will sometimes refer to aerial perspective as fog. Figure 3.3 illustrates static and dynamic aerial perspective.

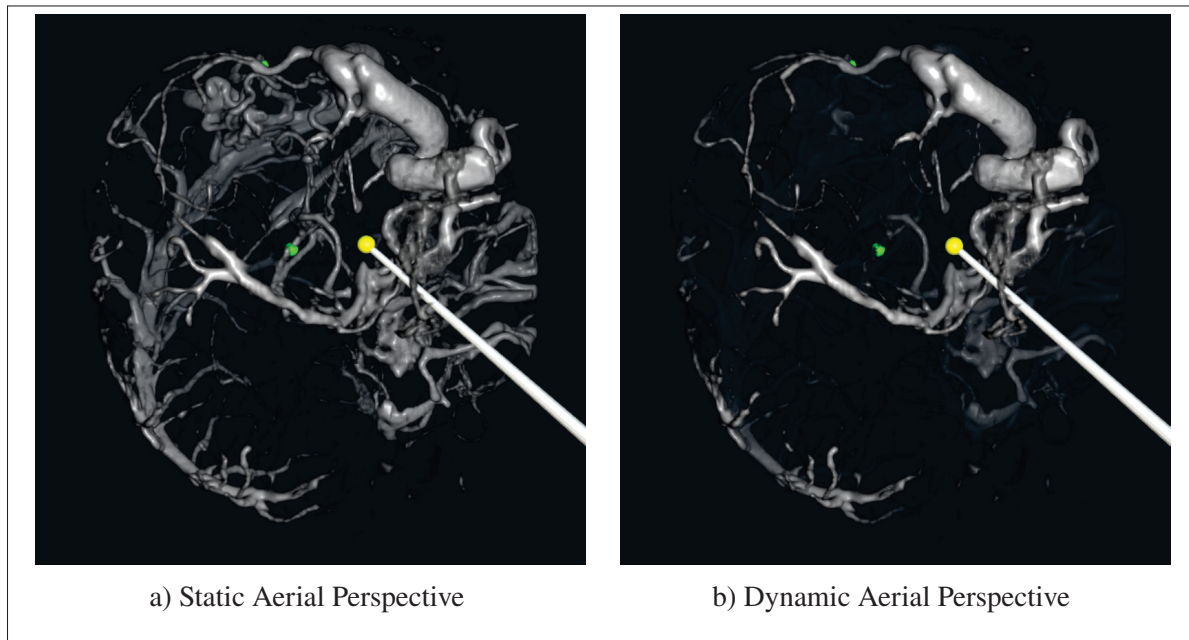


Figure 3.3 Static and dynamic aerial perspective. In static aerial perspective (a), the loss of contrast spans through the entire volume. With dynamic aerial perspective (b), the tip of the handheld pointer positions the close plane, which marks the start of the contrast decay. The loss of contrast then spans for a short distance, and further parts become almost invisible

3.3.2 Experiment

A preliminary user study was conducted with the goal of comparing the effectiveness of visualization of angiographic medical scans using the three previously described cues in their static and dynamic version in VR. The goal of the study was to determine which techniques resulted in the best spatial and depth understanding of the angiographic vasculature, when judged using objective metrics.

Data

The data consisted of 10 computed tomography angiographies (CTA) with 8 predefined views for each. Thus the study consisted of 80 trials. Half of the views had a distance of 20 *mm* between the targets, while the other half had a distance of 60 *mm*. The positions of the targets were chosen randomly, with the limitation that they had to be attached to some vessel in the volume. To achieve more variety for a single CTA scan, all views presented a different rotation. Additionally, the 80 trials were presented in a random order to each participant with a random cue associated with each to avoid any biases.

The angiographic scans have a resolution of $504 \times 509 \times 207$ *voxels* on average and are rendered based on the actual size, which is $218 \times 220 \times 152$ *mm* on average for all volumes. To achieve real-time direct volume rendering, the two-pass rendering algorithm described by Kruger & Westermann (2003) was used. Each participant saw each of the 3 cues 13 or 14 times, both in their static and dynamic version.

Setup

The study was designed as a sitting VR experience and was performed with the Valve Index headset (Valve Corporation, Bellevue, WA, USA). The participants interacted with the virtual environment using the Valve Index controllers which were secured to their hands. In the VR environment, the medical volumes are rendered in 3D in a stereoscopic manner, so that binocular disparity may be used by the participant as an additional cue to understand the volume. Interocular distance was adjusted for each subject.

Experimental Task

Prior to performing the experiment the users were given a presentation about the study and how each of the cues works and gave their informed consent. After this, each participant did a tutorial within the VR environment to get used to the system.

The experimental task was designed to replicate as closely as possible the work of Drouin *et al.* (2020) to enable a direct comparison of the effect of dynamic and static cues in a VR environment versus when using a 2D display. Minor adjustments were made given the context of the VR display.

The task, similar to previous works in this research area (Ropinski *et al.*, 2006; Kersten-Oertel *et al.*, 2014; Kreiser *et al.*, 2018), required the participant to determine which of two target vessels on the CTA volume is closer to the participant. The volume was fixed at the midpoint between the user's hands within the VR environment and the participant could move their head around to better understand the volume or to better localize the targets. The task of the user was to position the tip of a virtual pointer as close as possible to the closest of the two targets, without touching it. Each trial began with the user positioning the pointer on an orange spherical marker situated slightly below the volume (see Figure 3.4). This ensures that for all trials, the controller is positioned at approximately the same position at the start. Once the orange marker is touched, a new trial begins. Each trial ends in one of two ways: (1) the participant presses a button on the control that is not representing the surgical pointer or (2) if, by accident, one of the two targets is touched with the tip of the pointer. Performing the trials in this manner allowed us to measure spatial and depth understanding at both a global and local level. To determine which of the targets is closer to them, the participant must have a good global understanding of the angiographic volume. Whereas, to position the pointer as close as possible to the target without touching the target, the user must have a good understanding of the local structures of the volumes. As well as requiring precise localization, the limitation of not letting the user touch the target with the pointer also avoids having an additional occlusion cue which may affect the results.

Implementation

The study was implemented on the Unity Engine (Unity Technologies, San Francisco, CA, USA) with a custom GPU ray casting implementation using HLSL shaders. The volume rendering component was implemented based on the direct volume rendering algorithm described by

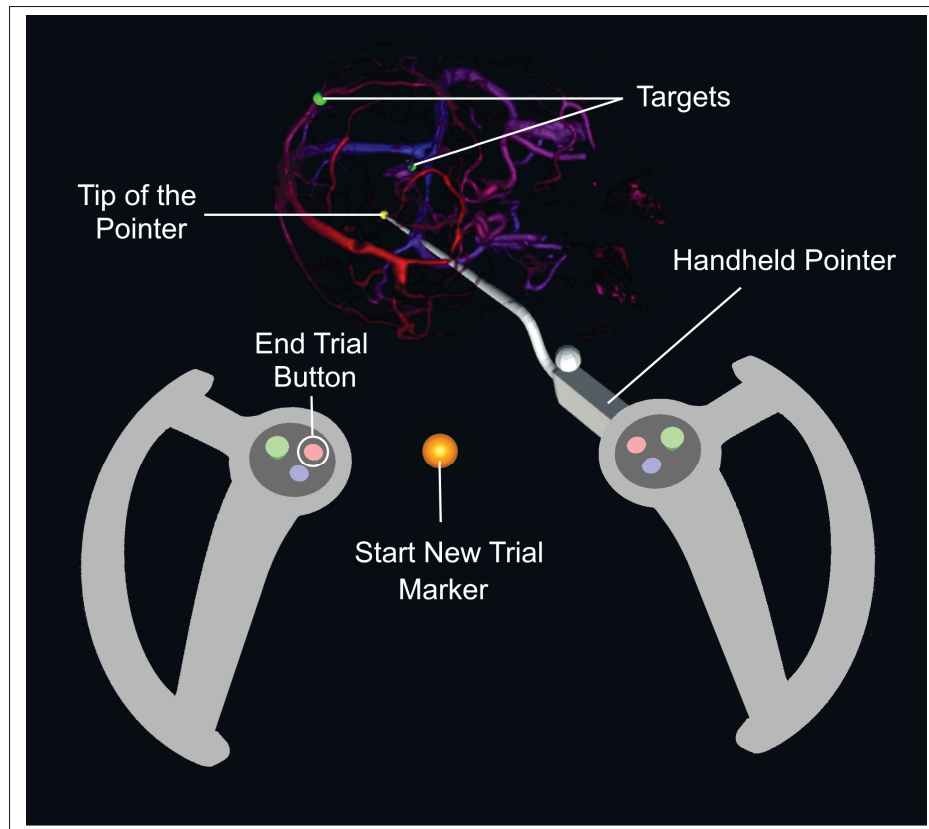


Figure 3.4 Physical setup superposed with the image observed in the VR headset. The right controller modified the position and rotation of the handheld pointer, while the left controller is used uniquely to press a button to end the trial. Targets are indicated in green and are always attached to a vessel of the angiographic scan

Drouin & Collins (2018). For the VR component, the Unity XR SDK¹ was used to allow hardware-agnostic VR. The implementation is currently closed-source.

The Valve Index VR Kit was used for VR during the user study. The rendering was done on a Windows 10 machine with an AMD Ryzen 7 3700X processor, 32 Gb of RAM and an NVidia RTX 2080 Ti video card with 11 Gb of video memory. The Valve Index headset provides a 1440×1600 LCD panel for each eye with 80 Hz, 90 Hz, 120 Hz and 144 Hz as possible refresh

¹ <https://docs.unity3d.com/Manual/XR.html>

rates². In our case, the rendering was done at 90 Hz in the virtual environment to ensure a smooth image.

3.4 Results

A total of 12 subjects (7 male, 5 female, age 19-45y) participated in the study. All subjects had normal or corrected to normal vision and none of them were color-blind. In the case of corrected eyesight, subjects were wearing glasses while using the headset.

Although none of the participants were clinicians, they were doing research related to medical image visualization or computer image perception enhancement. It was determined with a post-test questionnaire that 92% of the participants had at least some previous experience with either virtual or augmented reality, with 42% rating their level of experience with either as "Very High". Furthermore, 83% of the participants had at least some experience with volume rendering, 92% had at least some experience with medical images and 67% had at least some experience with neurovascular anatomy. However, only 25% of the participants said that they at least somewhat used their knowledge about neurosurgical anatomy to execute the tasks of the study

3.4.1 Objective Measures

The following metrics were recorded for each trial of the experiment:

1. **Correctness:** whether the participant properly determined which target was closer.
2. **Accuracy:** the Euclidean distance between the tip of the handheld pointer and the chosen target.
3. **Trial time:** the time that the participant took to complete the trial.
4. **Head path:** the total distance that the head has traveled during the trial calculated as the sum of Euclidean distances between the positions of the head at each frame. The position

² <https://www.valvesoftware.com/en/index/headset>

of the head is defined as the 3D position of the headset in the real environment, which is measured using the Base Stations³ in the Valve Index VR Kit.

We performed a 2-way repeated measures ANOVA on each of the four dependent variables, with the two discrete independent variables being the cue that was used (shading, chroma, fog) and whether the cue was static or dynamic. We will refer to these discrete independent variables as “cue” and “dynamics” correspondingly. The results are plotted in Figure 3.5.

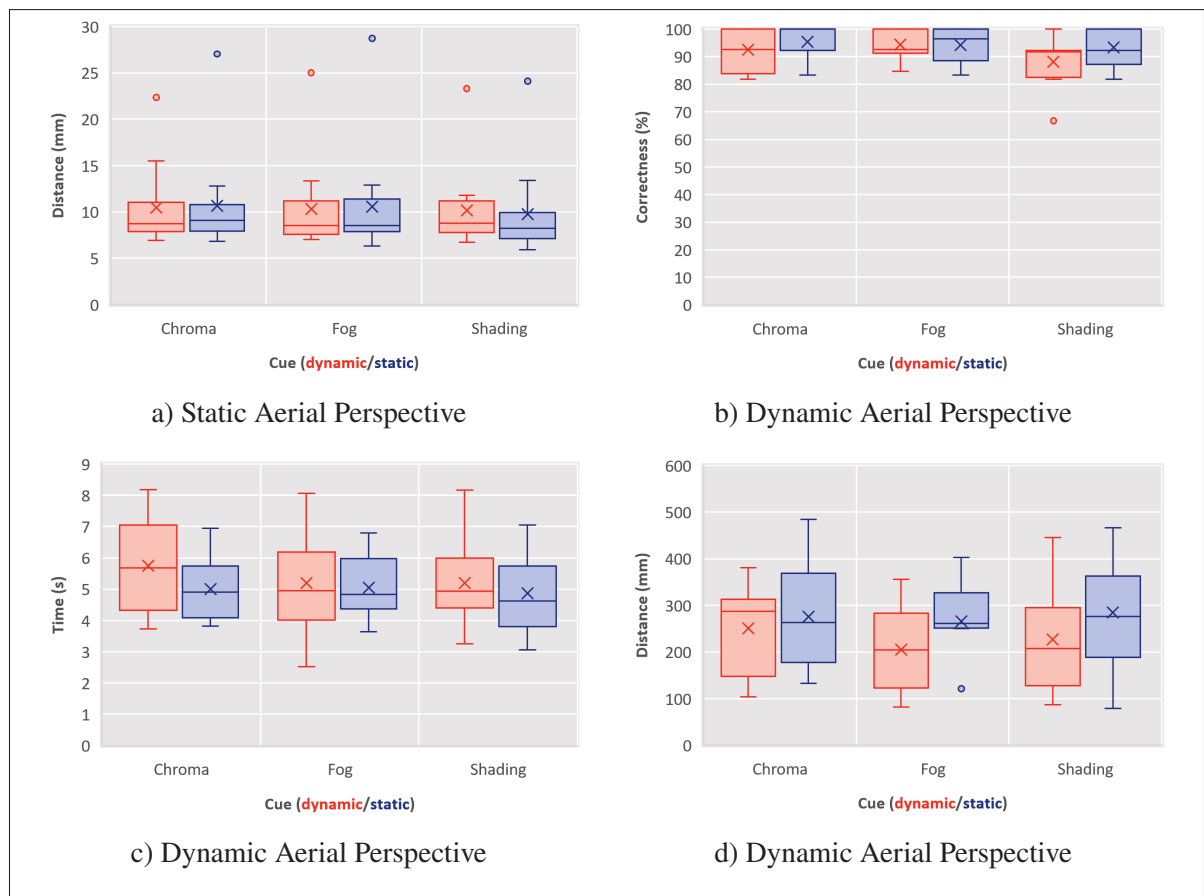


Figure 3.5 Collected data for each metric. Box plot represents the distribution of data in 4 quartiles. Dots represent outliers. Crosses represent the means

³ <https://www.valvesoftware.com/en/index/base-stations>

Effect on Accuracy

An ANOVA indicated that cue has a significant main effect on accuracy ($F(2, 22) = 3.484, p = 0.048$). Mauchly's test showed that the sphericity assumption was not violated ($\chi^2(2) = 1.48, p = 0.477$), so Greenhouse-Geisser correction was not applied. The Post-hoc Bonferroni-corrected pairwise t-tests have revealed that there exists a significant difference between the shading and chroma cues, and that shading results in 0.60 *mm* closer distance to the target (95% *CI* (0.06, 1.15), $p = 0.030$). No statistically significant difference was found between fog and other cues. Dynamics ($F(1, 11) = 0.006, p = 0.94$) and the interaction between cue and dynamics ($F(2, 22) = 0.867, p = 0.43$) did not have a significant impact on accuracy.

Effect on Correctness

A significant main effect of dynamics was found on correctness ($F(1, 11) = 7.163, p = 0.022$). With static cues, participants properly guessed the closest target 2.6% more often (95% *CI* (0.5, 4.7)). There was no significant main effect of cue ($F(2, 22) = 1.686, p = 0.208$) or the interaction between cue and dynamics ($F(2, 22) = 0.997, p = 0.385$) on correctness.

Effect on Trial Time

In terms of trial time, no statistically significant main effect was found for dynamics ($F(1, 11) = 3.41, p = 0.092$) or cue ($F(2, 22) = 0.98, p = 0.392$). Additionally, no significant two-way interaction between dynamics and cue was found ($F(2, 22) = 1.563, p = 0.232$).

Effect on Head Path

A statistically significant main effect on head path was found for dynamics ($F(1, 11) = 6.058, p = 0.032$). It was determined that dynamic cues resulted in a 47.5 *mm* shorter head path than static ones (95% *CI* (5.0, 89.9)). In terms of cue ($F(2, 22) = 0.67, p = 0.52$) or the interaction between dynamics and cue ($F(2, 22) = 0.51, p = 0.61$), no statistically significant effect was found.

3.4.2 Subjective Measures

Every participant completed a post-test questionnaire where they were asked to rate, using a Likert scale from 1 to 5, how easy it was to (1) determine which target is closer to them and (2) reach the chosen target accurately. The first is a subjective evaluation of **correctness** for each cue-dynamics combination. The second is a subjective evaluation of **accuracy** for each cue-dynamics combination. A two-way repeated measures ANOVA was calculated using the acquired information from the questionnaire where the first discrete variable is the cue and the second one is dynamics. The results are plotted in Figure 3.6.

Effect on Accuracy

In terms of user perception of accuracy, a statistically significant main effect was found on dynamics ($F(1, 11) = 8.13, p = 0.016$). The post-hoc t-test showed that subjects rated dynamic cues 0.75 points higher than static ones (95% *CI* (0.17, 1.33), $p = 0.016$). Additionally, a statistically significant effect was found on cue ($F(2, 22) = 9.14, p = 0.001$). The pairwise t-tests, Bonferroni-corrected, determined that subjects have rated chroma significantly higher than fog by 0.67 points (95% *CI* (0.27, 1.07, $p = 0.002$)).

Effect on Correctness

In terms of user perception of correctness, a statistically significant main effect on cue was found ($F(2, 22) = 5.34, p = 0.013$). The post-hoc pairwise t-tests (Bonferroni-corrected) indicated that there exists a significant difference between perception of chroma and fog, and that participants rated chroma 0.58 points higher than fog on average (95% *CI* (0.07, 1.10), $p = 0.026$).

3.5 Discussion

In our study, we found that cue had no impact on relative depth perception or on decision time, while in many of the previous studies (performed using a 2D display) cues had a significant effect on depth perception (Ropinski *et al.*, 2006; Kersten-Oertel *et al.*, 2014; Kreiser *et al.*,

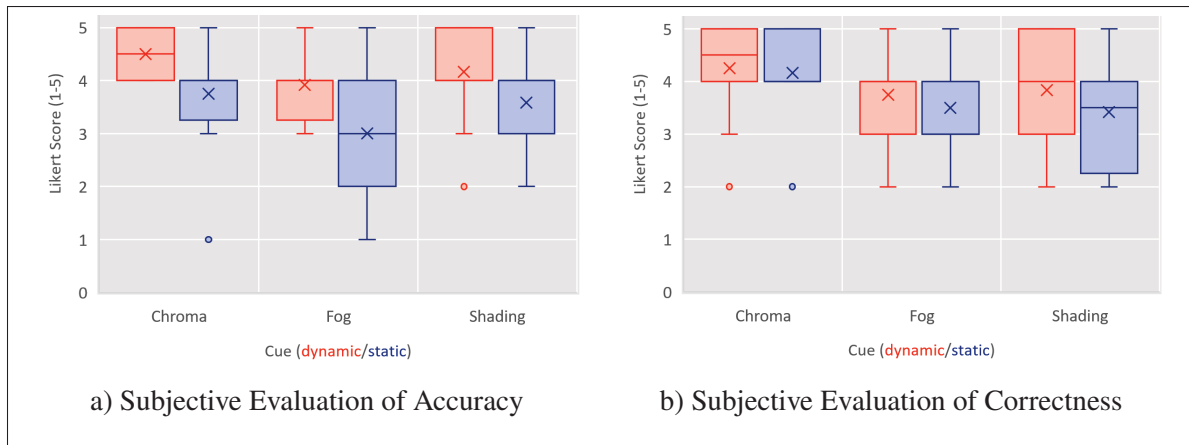


Figure 3.6 Collected subjective data for each metric. For each cue-dynamic combination, the subject was asked to rate how easy it was to reach the target (a) and how easy it was to determine the closest target (b)

2018). In these studies, chromadepth or its two-color variation were generally found to be more efficient than shading, both in terms of correctness and trial time. However, it seems that in VR, cue has little effect on those two variables. This result is very similar to what was observed by Heinrich *et al.* (2021), where cue had a significant impact on depth perception in the desktop 2D environment, while having little impact in the VR environment. Similarly to what was suggested in that study, we posit that this discrepancy may be due to the fact that in VR offers additional perception cues such as stereoscopy, which provide a good spatial and depth perception even with simple rendering. Thus, the specific rendering parameters in VR have less impact than on 2D displays.

The results of our study suggest that in terms of appreciation of local structures of the CT angiographic data (i.e. accuracy), shading performs better than chromadepth. Participants were able to get significantly closer to the target using shading than when using chromadepth. This differs from the results obtained by Drouin *et al.* (2020), where chroma and fog resulted in better appreciation of local structures than shading. This difference may suggest that in VR, shading provides a stronger depth understanding particularly if the light source is coupled with head movement. Even in static shading, the light is attached to the position of the viewer, so

it is possible that by moving their head, viewers reveal some additional information about the geometry. However, more research is needed to confirm this hypothesis.

In terms of the effects of dynamics, we found that dynamic cues did not have a significant impact on understanding of local structures. However, in terms of understanding of global structures by identifying the closer target (measured by the “correctness” variable), we determined that static cues performed better than their dynamic counterparts.

When compared to the study by Drouin *et al.* (2020), we found that for local and global understanding of the structures, dynamic cues performed worse in VR than when using a 2D display. On a 2D display, dynamic cues resulted in better understanding of local structures, while not having an effect on understanding of global structures (Drouin *et al.*, 2020). In our study, dynamic cues performed equally or worse than static ones. This may suggest that in VR, binocular disparity and motion parallax play an important role in depth perception, which could sometimes negate the advantages of the dynamic cues.

Another difference that we observed between our and Drouin *et al.* (2020) study is that we didn’t have a two-way interaction for any of the variables. In our results, there was no cue-dynamics combination which performs better than others in any metric.

When it comes to head movement, participants moved their heads significantly less with dynamic cues than with static cues. This is an interesting result which could potentially be a strong argument in favor of dynamic cues. The more head movement in static cues suggests that the reason why participants were able to achieve better global appreciation of the vessel structure is because they relied on the motion parallax cue more. Dynamic cues could be particularly interesting if they are used in a context where the point of view is fixed, for example with a surgical microscope in the operating room (Drouin, Kersten-Oertel & Louis Collins, 2015). We posit that the dynamic aspect could compensate for the absence of motion parallax which could have a significant impact on bringing these technologies in an intuitive and ergonomic way into an operating room context. We plan to explore this in future work.

When it comes to the comparison of effectiveness between static and dynamic cues, a discrepancy between objective and subjective results was found. This discrepancy is very similar to the one noted by Drouin *et al.* (2020), in that participants were overly confident about the effectiveness of dynamic cues. Even though subjects got similar accuracy with both static and dynamic cues, they thought that they were more effective with dynamic cues. A similar pattern could be observed for correctness; although subjects were able to correctly determine the closest target better with static cues, they thought that they performed equally well with both static and dynamic cues.

Overall, 75% of subjects said that they prefer dynamic cues over static ones. Even though this did not translate into better appreciation of local or global structures with static cues. We believe that this is due to the fact that with static cues participants had to move their head significantly more (perhaps to use the motion parallax cue), which may be more energy consuming for the user.

One of the limitations of the study was a relatively small number of participants (12) which was caused by the lockdown restrictions of the COVID-19 pandemic. However, this study size is similar to previous vascular volume visualization studies performed in a research lab environment where the number of participants varies from 10 (Abhari *et al.*, 2012) to 30 (Heinrich *et al.*, 2021). To ensure a high quality of data with subjects that managed to participate, the users were asked to remove their mask while using the VR headset to ensure their comfort. The headset and controllers were cleaned between each participant.

While our initial analysis focused on establishing the statistical significance of these visualization cues, future studies building on this work should incorporate effect size metrics (e.g., Cohen's d 2013). Relying solely on p -values can obscure the practical magnitude of differences, especially with smaller sample sizes. Quantifying these effect sizes will provide a clearer picture of the real-world utility of dynamic versus static cues.

3.6 Conclusion

The goal of this study was to determine how medical volumes rendered with specific depth cues, initially studied on 2D displays rendering, could translate into a virtual environment. To do so, we performed a user study where we implemented three widely studied perceptual cues (shading, chromadepth and aerial perspective) and their dynamic counterparts proposed by Drouin *et al.* (2020).

Unlike with 2D displays, chromadepth and aerial perspective had little effect on previously widely studied metrics such as trial time and correctness. It seems to us that other cues in VR such as stereoscopy and motion parallax overshadow the transfer function-based ones if judged by these two metrics. However, in terms of local appreciation of the volume, we determined that shading results in better accuracy than chromadepth. This is a surprising result which might suggest that there exists an interaction between shading and stereoscopy or motion parallax. We plan to explore this in future work.

In terms of dynamics, dynamic cues perform equally or worse than static ones if we judge them using the same metrics that exist in 2D. However, if we look at observer movement during the study, the total path that the head travelled for each participant during the study was significantly less for dynamic cues. This may be considered as an advantage of dynamic cues. It is possible that for static cues, participants have to rely significantly more on the motion parallax cue, which explains the additional movement. Another advantage that dynamic cues have is that participants generally like them more, and they also think that they perform better with them than what can be observed using objective measures. Determining if there exists a link between higher preference for dynamic cues and smaller head movement could be an interesting research topic in the future.

While this study established the main effects of visualization cues on performance, we did not explore correlations between dependent variables, such as speed versus accuracy or speed versus head path. Given our sample size of 12, an in-depth correlational analysis lacked the statistical power required for robust conclusions. Future work with a larger cohort will evaluate these

relationships using scatter plots, trendlines, and R^2 values to better explain the trade-offs users make during data exploration.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by the NSERC, Discovery program, RGPIN-2020-05084.

CHAPTER 4

CONTEXTUAL AMBIENT OCCLUSION: A VOLUMETRIC RENDERING TECHNIQUE THAT SUPPORTS REAL-TIME CLIPPING

Andrey Titov^{1,2}, Marta Kersten-Oertel², Simon Drouin¹

¹ Department of Software and IT Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

² Gina Cody School of Engineering and Computer Science, Concordia University,
1455 De Maisonneuve Blvd. West, Montréal, Québec, Canada H3G 1M8

Article published in the journal "Computers & Graphics" in April 2024

4.1 Introduction

Ambient occlusion (AO) is a global illumination technique in computer graphics that is used to estimate how much each particle in the scene is illuminated with ambient lighting. The ambient factor of a particle can be viewed as an “accessibility” of the particle (Miller, 1994). The general idea is that particles that are more obstructed and thus less accessible receive less ambient light and become darker consequently. In volume rendering, the particle is surrounded with semi-transparent or opaque voxels that cause this darkening by absorbing the ambient light and not producing any emission (Hernell, Ljung & Ynnerman, 2007).

One of the variations of volumetric AO, Local Ambient Occlusion (LAO), is a shading technique that could be used in volume rendering to provide better understanding of volumetric structures by darkening regions less likely to be exposed to ambient light (Hernell *et al.*, 2010). However, the computation of a darkening factor is expensive because it requires sampling the neighborhood of a voxel and is usually computed prior to visualization. When using interactive volumetric clipping techniques, the ambient occlusion term needs to be recomputed every time the clipping region is modified because any change to the voxel opacity will affect how much neighboring voxels are obstructed. This computation is difficult to achieve in real-time, as can be seen on Figure 4.2. In this paper, we propose a new technique called Contextual Ambient Occlusion

(CAO) that enables fast recomputation of regions where ambient occlusion is likely to have changed every frame, while maintaining temporal stability.

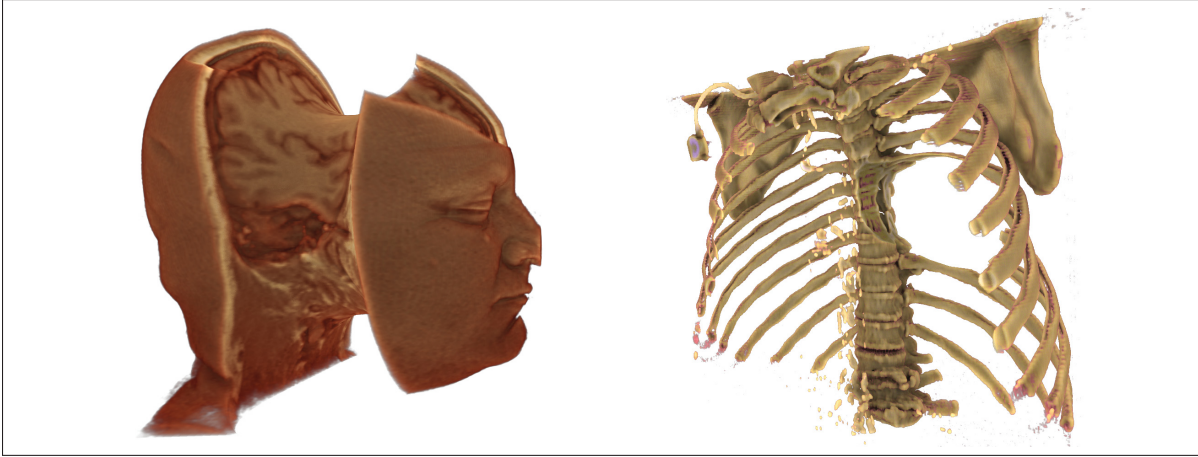


Figure 4.1 CAO applied on medical volumes. One concave clipping shape (torus) and one convex shape (cylinder) are applied to an MRI head on the left. The same convex clipping mesh (cylinder) is applied to a CT chest image on the right. Both medical images were taken from the default dataset provided by 3D Slicer software¹

The main idea of the algorithm is that in LAO, the ambient factor of a voxel is affected only by a small number of surrounding voxels in a certain spherical radius. Therefore, when parts of the volume are clipped, only a small number of voxels have their value affected by the clipping. In our algorithm, we propose a pipeline that allows us to identify those voxels and perform the LAO recalculation only for them. Figure 4.1 illustrates an example of an image that we were able to obtain with this algorithm.

The algorithm was specifically designed for volume ray casting where a perspective transformation (which includes all linear transformations such as translation, rotation, and scaling) could be applied to the clipping region every frame. Therefore, it is particularly well suited for interactive visualization of volumetric data where the user controls the position and rotation of the clipping regions, such as with the interactive clipping technique presented by Joshi *et al.* (2008). In addition to the fast calculation time, the algorithm achieves high AO quality because it uses the discrete volume rendering integral for opacity calculations, similar to Hernell *et al.* (2010).

¹ <https://www.slicer.org/>

However, unlike the implementation described by Hernell *et al.* (2010) where only one ray is cast per frame, the algorithm calculates the final AO factor for all rays every frame, which avoids having a blurry effect that comes from the combination of multiple frames. This makes it possible to use our algorithm even with a significant change of the position, rotation or scale of the clipping region every frame.

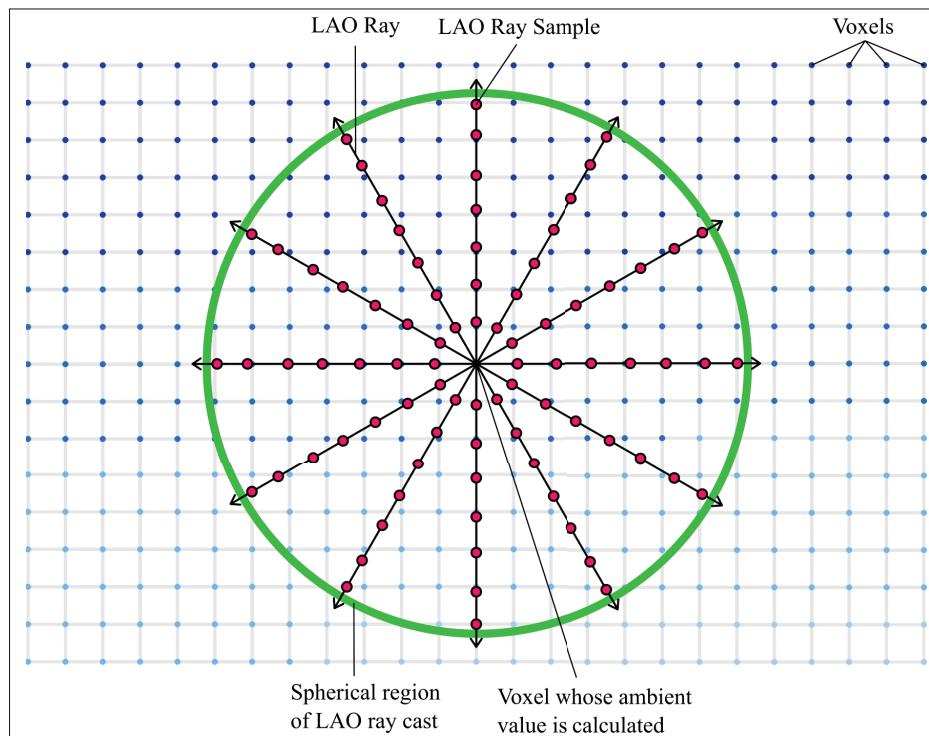


Figure 4.2 Ray cast performed for each voxel to calculate the LAO factor. Each red point represents a tri-linearly interpolated sample taken from the volume. The ambient factors of all rays are then aggregated together to create a single ambient value for the voxel

In addition to the previous advantages, the new method is compatible with virtual reality (VR) and augmented reality (AR) displays because all the ambient occlusion factor calculations are done in data space. This avoids the problem of screen-space stereoscopic inconsistency (Shi *et al.*, 2022) and makes the algorithm scale well with an increasing resolution and number of viewports. We evaluated the performance of the algorithm in comparison with a naive implementation where the ambient factor is recalculated for all voxels every frame and found

that our algorithm offers a 2-5x speedup. Additionally, if we compare our algorithm to the one proposed by Hernell *et al.* (2010), the latter takes multiple frames (e.g. 54 to achieve a similar quality to images presented in Figure 4.9) to recover from a change of clipping frames, which is unacceptable for interactive VR environments, and our algorithm fixes this problem.

This paper is an extended version of the "Contextual Ambient Occlusion" article (Titov, Kersten-Oertel & Drouin, 2023) presented during the 2023 Graphics Interface conference. The main contributions compared to the conference paper are the publication of the source code, the description of an extension of the algorithm that supports multiple simultaneous clipping meshes, as well as a thorough evaluation of the effect of the number of rays used during the spherical ray casting on the quality of the image, which highlights the advantages of the CAO algorithm compared to LAO.

4.2 Related Work

One of the most popular techniques to calculate ambient occlusion that is commonly used in real-time computer graphics is Screen-Space Ambient Occlusion (SSAO), which has the advantages of only requiring a depth map as an input and having good performance due to the fact that all calculations are done in screen-space (Bavoil & Sainz, 2008). However, in volume rendering where a depth map cannot be obtained due to the absence of isosurfaces in the dataset that needs to be visualized, alternate algorithms to estimate the ambient factor have to be used.

One of the first papers to present a real-time implementation of AO-alike volume shading technique was Vicinity Shading (Stewart, 2003). This algorithm first calculates the normal for each voxel, and then sends 3D Bresenham rays in a half-spherical neighborhood at the same side as the normal vector. Each ray traverses the volume until it reaches a voxel whose opacity is higher, and the results of all rays are aggregated together using averaging. This averaged value then represents that vicinity value of the voxel, which can also be interpreted as the ambient factor of the voxel.

Díaz, Vázquez, Navazo & Duguet (2010) presented an algorithm that speeds up the precalculation part compared to vicinity shading and does not require calculating the normal of every voxel. The algorithm uses the concept of 3D summed area tables (SAT) which are used to quickly aggregate and calculate the average opacity of multiple voxels. In the precalculation step, a 3D SAT volume of opacities is constructed from the initial volume and used during rendering to quickly evaluate the average vicinity value for a single voxel with very few texel fetches. The biggest disadvantage of this algorithm is that 3D SATs give only a rough approximation of the ambient value because all voxels used to calculate the vicinity value have equal weight, no matter their distance from the voxel. To improve upon this problem, another algorithm based on the idea of using 3D SATs was presented by Schlegel, Makhinya & Pajarola (2011). This improved algorithm changes how the 3D SAT is used at runtime. Instead of simply estimating the average vicinity value around the voxel, the algorithm calculates “shells” around the volume where each one has a different distance from the voxel. Then, the values of all shells are aggregated together using the absorption formula presented by Max (1995), similar to the standard volume rendering integral. However, even in that case, due to the aggregation of many voxels together, only a rough estimation of the ambient factor is obtained.

Unlike 3D SAT table algorithms which focus on fast precalculation and rendering time, some other algorithms focus on having a high quality of the ambient value by focusing on the physical accuracy of the ambient value calculation. Such is the case of the Local Ambient Occlusion (LAO) algorithm presented by Hernell *et al.* (2010). This algorithm presents an expensive precalculation step where for each voxel its ambient value is calculated. This calculation consists of casting multiple rays in a spherical pattern around each voxel. Each of the rays samples the opacities in the volume and aggregates them using the absorption formula described by Max (1995). The values of all rays are then averaged together to create a single ambient value for the voxel. The algorithm produces high quality results as further voxels affect the ambient value exponentially less than closer ones, which is more physically accurate.

Some ambient occlusion algorithms focus on being able to properly estimate the ambient factor if the opacity transfer function of the volume changes in real-time. Such is the case of the

Table 4.1 Comparison of existing volumetric ambient occlusion methods. Quality represents the relative quality of the obtained ambient value. The “Real-time modification to” column indicates what type of modification could be done in real-time within the algorithm without a loss in performance. The red, yellow and green circles represent low, medium and high quality of the ambient factor correspondingly

Author(s)	Name	Quality	Real-time modification to:		
			Radius	TF	Clipping
Stewart (2003)	Vicinity shading	● ● ○	✗	✗	✗
Ropinski, Meyer-Spradow, Diepenbrock, Mensmann & Hinrichs (2008)	Clustered histograms	● ● ○	✗	✓	✗
Díaz <i>et al.</i> (2010)	SAT of opacities	● ○ ○	✓	✗	✗
Díaz <i>et al.</i> (2010)	Local Ambient Occlusion	● ● ●	✓	✗	✗
Schlegel <i>et al.</i> (2011)	Shelled SAT of opacities	● ● ○	✓	✗	✗
Engel & Ropinski (2021)	Deep-learned AO	● ● ○	✗	✓	✓
Ours	Contextual Ambient Occlusion	● ● ●	✗	✗	✓

clustered histograms algorithm presented by Ropinski *et al.* (2008), whose main idea is that voxels with approximately the same neighborhood will have roughly the same ambient factor independently of the transfer function used. In this algorithm, for each voxel, a local histogram of neighboring voxels is created, and then similar histograms are clustered together. This histogram stores that degree of influence that each neighboring voxel has on the main voxel using inverse distance weighting. Consequently, a new 3D texture is filled which contains the cluster index for every voxel. With this information, the ambient factor is calculated in real-time by applying the transfer function on the histogram. In 2021, a different approach was proposed to estimate the ambient factor with dynamic transfer function change by Engel & Ropinski (2021). In this algorithm, deep learning was used to predict the impact of a change of the transfer function on the ambient value. Many combinations of MRI volumes and transfer functions were created and rendered with Monte Carlo ray tracing, and the resulting voxel ambient values were learned.

The created model could then be used to predict the ambient value for any MRI image and any opacity transfer function.

In Table 4.1, we presented a summary of volumetric ambient occlusion algorithms where we compared them in terms of their ambient factor quality, and we also evaluated what types of real-time changes can be done to the algorithm. In order to rate the quality, we separated all the algorithms into 3 logical groups, ranging from low to high that measure how physically accurate the ambient factor is. The high-quality group encompasses all algorithms that follow the absorption model presented by Max (1995) and perform spherical ray casting. The medium-quality group consists of algorithms that provide some variation of inverse weighting for voxels based on their distance from the voxel for which the ambient value is calculated. The low-quality group consists of algorithms that calculate an unweighted average of opacities of neighboring voxels for a single voxel.

4.3 Methods

The CAO algorithm is able to recalculate in real-time the changes to occlusion that occur in the volume when parts of this volume are clipped, while achieving exactly the same quality as LAO. Because the algorithm simply prevents unaffected voxels from being recalculated, the calculation is always exact and remains stable over successive frames. All of the ambient occlusion computations are done in data space to avoid the stereoscopic screen-space inconsistency problem described by Shi *et al.* (2022), which makes it suitable for VR and AR environments. The source code is published in the form of a Unity3D package², which can be easily integrated in existing Unity3D projects.

² <https://github.com/andrey-titov/ContextualAO>

4.3.1 Pipeline

The central idea of the CAO algorithm is that we render an inflated version of the clipping shape in two depth buffers, and then we use these buffers to determine which voxels need recomputation of the ambient factor.

The algorithm requires as input a volume V stored in a 3D texture, the opacity transfer function TF_o as well as the clipping mesh M . The mesh M should satisfy the requirements described by Weiskopf *et al.* (2003), meaning it has to be either convex, or concave with the limitation that it could be fully rendered with exactly two opacity peeling steps (Everitt, 2001). For example, as indicated in Figure 4.1, a concave shape such as a torus can be used in this algorithm because it satisfies this constraint.

We propose a novel approach to obtain the correct ambient factor for all voxels every frame by only recalculating the ambient factor for voxels where it was affected by clipping. We use shadow test volume clipping (see subsection 4.3.2) to perform the clipping and dilation maps (see subsection 4.3.3) to determine which voxels were affected by clipping.

A visual representation of the pipeline is given in Figure 4.3, where the steps executed using a compute shader are marked as C.S., while the steps executed using the rendering pipeline are indicated with R.P.

Steps done before ray casting:

- I. (C.S.) The ambient factor is computed for all voxels using LAO spherical raycasting assuming that no clipping is applied on the volume. The resulting AO values are stored in a volume called V_{PAO} and used later.
- II. (R.P.) The front and back faces of the clipping mesh are rendered using two depth-only rendering passes and the resulting depth maps are stored in two textures.
- III. (C.S.) Dilation maps are computed from the depth maps created in step I and stored in two textures. The dilation maps are computed by calculating the Minkowski sum

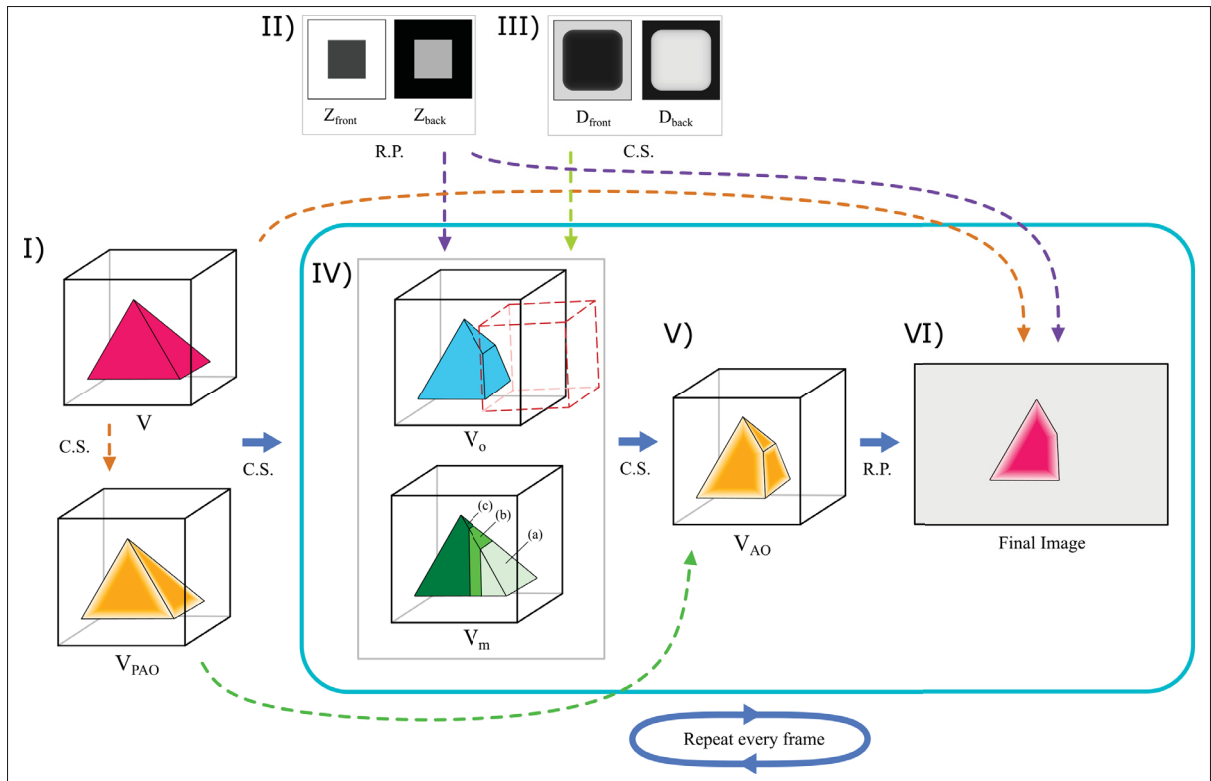


Figure 4.3 Visual representations of the 6 steps necessary to obtain the volume shaded using CAO. Steps I-III are done once before the rendering, while steps IV-VI are done every frame. C.S. represents compute shaders while R.P. stands for the rendering pipeline

(Schneider, 2013) between the depth map and a sphere that represents the LAO rays cast region.

Steps done during rendering:

- IV. (C.S.) Shadow test volume clipping (see subsection 4.3.2) is done for each voxel of the initial volume V , and the result is stored in a volume V_o containing the opacity of each voxel after clipping is applied. Additionally, a mask volume V_m is populated that indicates for each voxel if it was clipped, and in the case that it was not, whether its ambient value was affected by clipping.

- V. (C.S.) A volume containing the final LAO coefficients V_{AO} for the image is outputted. The volume V_m is read to determine whether the LAO value should be recalculated with a spherical ray cast, or if the initial ambient value stored in V_{PAO} could be used as-is.
- VI. (R.P.) The initial volume V is rendered using shadow test volume clipping (see subsection 4.3.2), and at every ray step the V_{AO} volume is sampled to determine the ambient factor for the current ray sample.

4.3.2 Shadow test volume clipping

In our implementation, we use a clipping algorithm similar to shadow test clipping presented by Weiskopf *et al.* (2003), which we adapted to be used with ray casting volume rendering. It is used during the computation of LAO and to render the clipped volume. Shadow test clipping is a modification of convex volume cutting based on depth clipping described by Weiskopf *et al.* (2003). The advantage of this algorithm is that it makes it very simple to test for any point inside the volume if it is clipped, by only requiring to sample two depth maps.

In standard depth clipping (Weiskopf *et al.*, 2003), a convex mesh is inserted in the volume, and the boundary of the mesh is used to separate the clipped parts from the visible ones. In our case, we consider the region outside the mesh as being visible, while the inside part is considered to be invisible, and therefore clipped. This corresponds to depth clipping described by Weiskopf, Engel & Ertl (2002), which can be seen as a subtraction of the clipping mesh from the original volume. However, other types of logical operations can be supported too. The algorithm could be adapted to work with volume probing (Weiskopf *et al.*, 2002) where the region inside the mesh is visible and everything else is invisible. This would correspond to the intersection logical operator between the mesh and the volume. The modification would require performing a contraction instead of a dilation of the clipping mesh in step III of subsection 4.3.4, as well as inverting the conditions described in Lines 24 and 26 of algorithm 4.2 in step IV.

In the original implementation of depth-based volume cutting (Weiskopf *et al.*, 2003), the front and back faces of the clipping mesh are rendered from the point of view of the observer in

two different passes, and the created depth buffers Z_{front} and Z_{back} are stored. Then, during rendering, each fragment³ f is considered as visible only if it passed the following clipping test (assuming that smaller depth values correspond to closer objects to the camera):

$$visible(f) = (z_f \leq z_{front}) \vee (z_f \geq z_{back}), \quad (4.1)$$

where $visible(f)$ is a Boolean function that indicates if the fragment f is visible (i.e. not clipped), z_f corresponds to the depth value of f , and z_{front} and z_{back} represent the depths within the depth buffers Z_{front} and Z_{back} for the current pixel location of f .

Shadow test volume clipping is a modification of the depth-based clipping where the location from which Z_{front} and Z_{back} are computed differs from the location of the observer (Weiskopf *et al.*, 2003). In this scenario, these two depth buffers are created from an arbitrary position in the scene with projection parameters that may differ from the projection camera, and we will further refer to it as the clipping camera. Because of this change, when the clipping test is performed on each fragment f during rendering, this fragment should be transformed to the coordinate space of the clipping camera. This technique is referred to as shadow test clipping because the idea of this technique is very similar to shadow mapping (Williams, 1978), where a light source creates a depth map of the scene, and each sample in the scene has to be transformed into the coordinate space of the light source projection camera, and its depth is compared to the one in the shadow map to determine whether the sample is lit or in a shadow. An illustration of shadow test clipping is given in Figure 4.4.

In our implementation, we use shadow test volume clipping both during the computation of the volume containing LAO information, but also during the rendering of the volume. As indicated in Figure 4.4, we use an orthogonal projection for the clipping camera to obtain the highest precision in the depth buffers. During the rendering, the position and rotation of the virtual

³ Note that the word “fragment” is used to describe samples of the volume because the original clipping algorithms were implemented with texture mapping (Weiskopf *et al.*, 2003). In ray casting, “sample” should be used instead to denote information retrieval from a 3D texture.

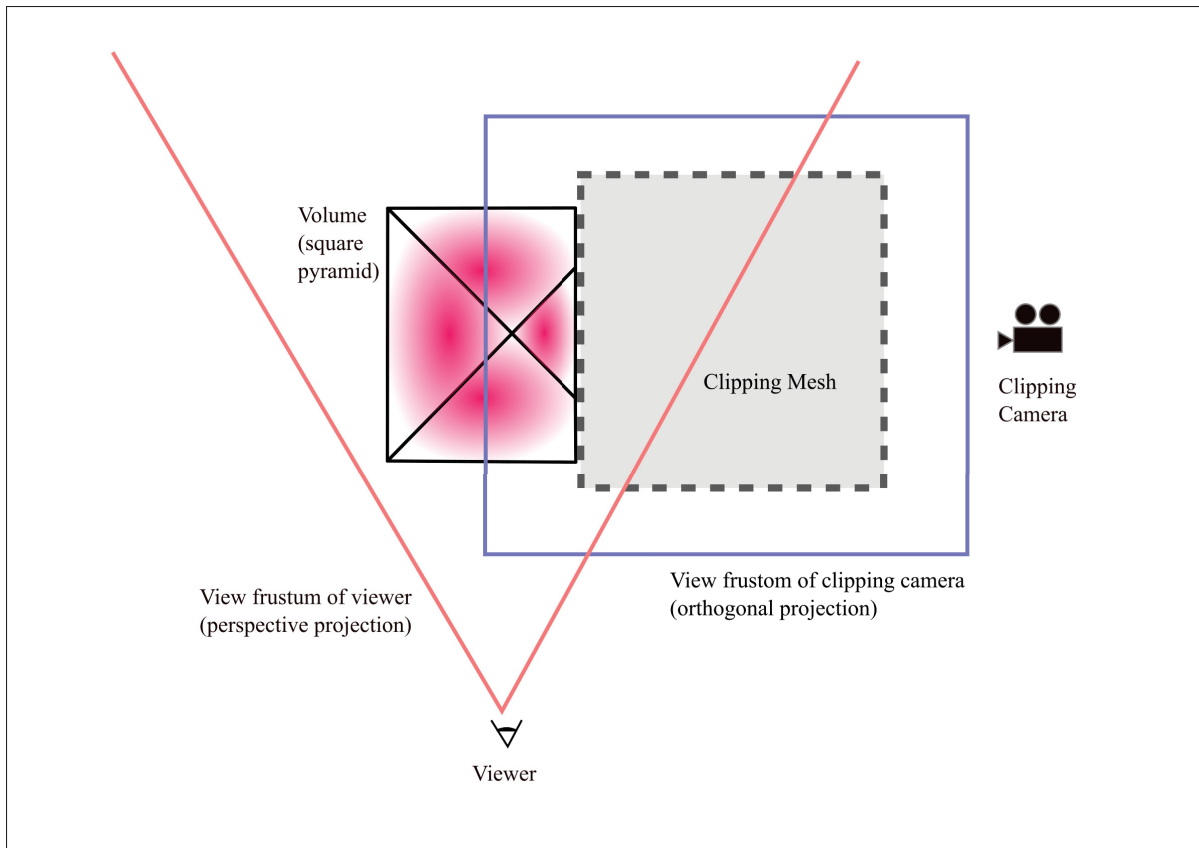


Figure 4.4 Shadow test volume clipping. The clipping camera is attached to the clipping mesh and renders its front and back faces once to depth buffers. These depth buffers are then used every frame to determine which fragments are clipped by performing a clipping test. Each fragment needs to be transformed to the coordinate space of the clipping camera, sample the depth buffers and then perform the clipping test to determine whether the fragment is visible

camera is synchronized with the clipping mesh, so that the depth maps never change from the point of view of the camera. This makes it possible to precalculate these depth maps before the rendering and never change them after, unless the geometry of the clipping mesh would dynamically change.

4.3.3 Local Update of LAO

The key idea of CAO is that clipping modifications only affect the ambient factor of a small number of voxels located close to the clipping boundary, so LAO could be recalculated only for

the voxels that were affected by clipping. This is illustrated in Figure 4.5, where all voxels fall into 3 categories: (a) voxels that were clipped, (b) non-clipped voxels whose ambient factor was affected by the clipping and (c) non-clipped voxels that weren't affected by the clipping. By only recalculating the ambient factor of voxels that fall into category (b), it is possible to significantly speed up the calculation time of LAO without compromising the quality of the final rendered image. In order to do so, the CAO algorithm precalculates the LAO factor for all voxels of a volume before any clipping is applied, and intelligently recalculates LAO only for those affected voxels.

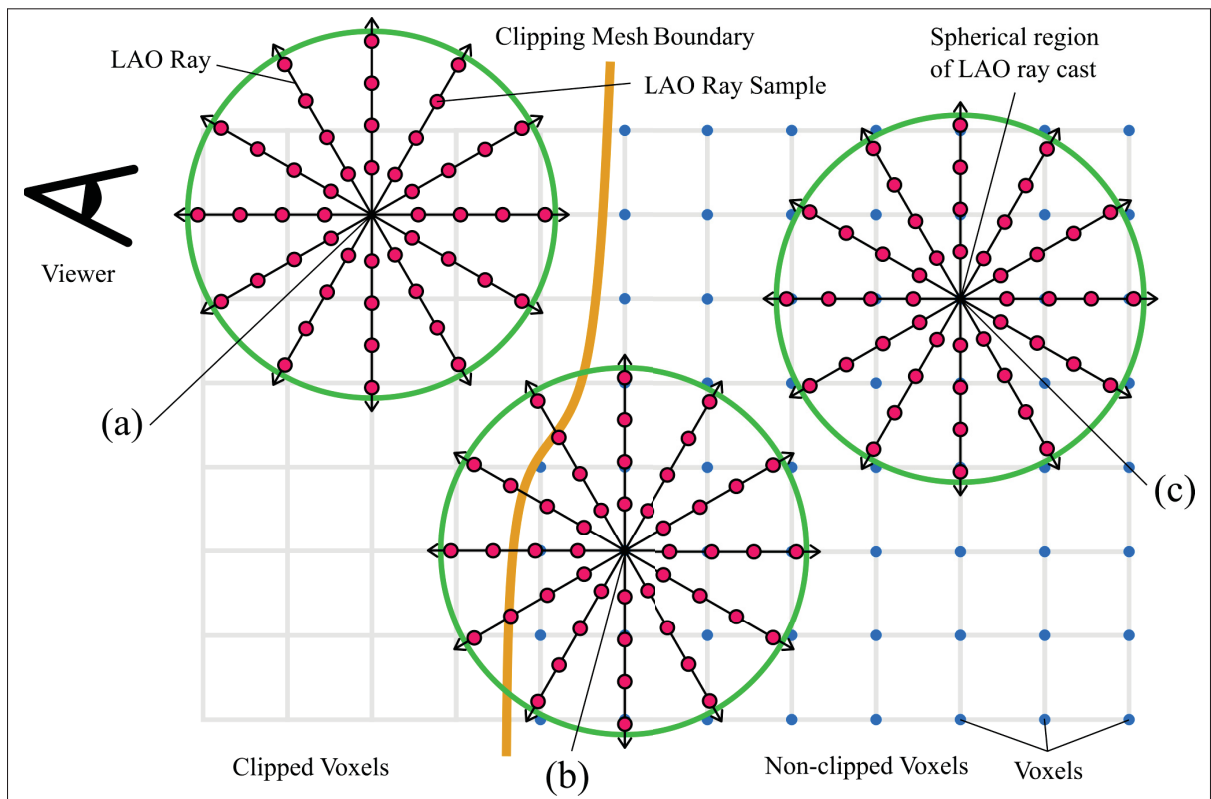


Figure 4.5 3 possible categories for the voxel. The green circles represent the spherical neighborhood used to perform the LAO calculation. The orange line represents the boundary of the clipping mesh

To distinguish which voxels have an ambient factor affected by clipping, a Minkowski sum (Schneider, 2013) is calculated between the clipping mesh and a sphere representing the neighborhood considered for LAO calculation, as illustrated in Figure 4.6. The radius of the

sphere must correspond to the length of the rays sent during the ray casting for each voxel. The mesh created from this Minkowski sum could be used with depth-based mesh clipping to delimit the region of voxels for which LAO should be recalculated. We will further refer to it as the dilation shape.

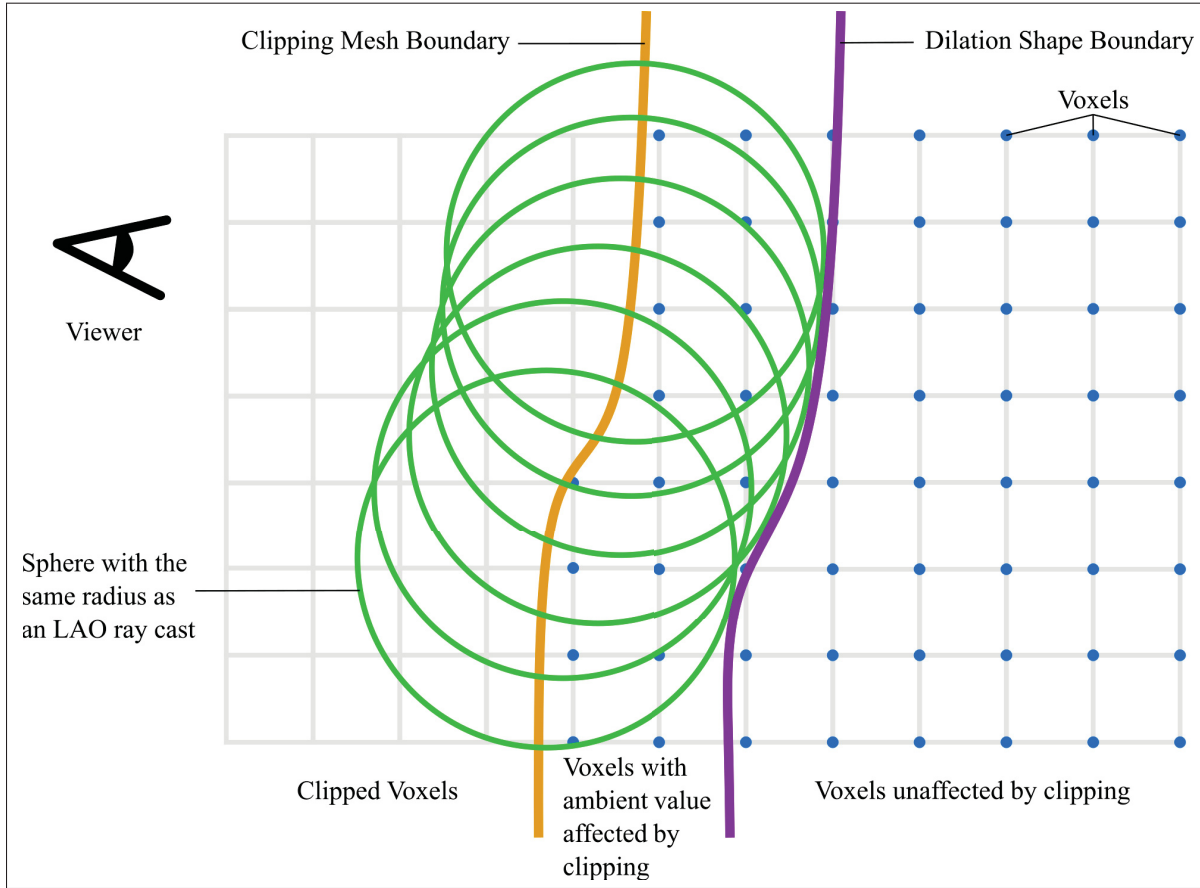


Figure 4.6 3 possible categories for the voxel with boundaries. The green circles represent the spherical neighborhood used to perform the LAO calculation. The orange and the purple lines represent the boundaries of the clipping mesh and the dilation shape correspondingly

In the actual implementation, this Minkowski sum (Schneider, 2013) does not need to be explicitly performed and the dilation shape does not need to be created in the form of a mesh because doing so is computationally expensive, and a significantly simpler approach could be taken instead. Since the initial clipping mesh already creates depth maps used for depth-based volume clipping, the depth maps of the dilation shape could be computed using them. Therefore,

instead of computing a Minkowski sum in 3D, it can be calculated in 2D by adding a half-spherical depth map to the depth maps created by the initial mesh. This sum will create the depth map of the dilation shape which is then used to classify voxels. Since the rendering of the volume will be done with shadow test volume clipping, this operation would only need to be performed once before the rendering begins.

4.3.4 Precomputation of the Dilation Shape Depth Maps

Before starting rendering, three precalculation steps should be completed.

- I. **The ambient factor is computed for all voxels in the volume in a spherical neighborhood using LAO and the result is stored in a volume called V_{PAO} .** In order to compute the ambient factor for a single ray, the absorption model is used Hernell *et al.* (2010). Thus, for a ray k sent from the voxel x the ambient factor is expressed as:

$$A_k(x) = \frac{1}{M} \sum_{m=1}^M \prod_{i=1}^{m-1} (1 - a_i), \quad (4.2)$$

where M represents the number of samples taken by the ray, m and i are used to iterate through the samples (assuming that smaller numbers corresponds to closer samples), and a_i represents the opacity at the i -th sample. To combine the ambient values of all rays, the following formula is used (which performs an averaging):

$$A(x) = \frac{1}{K} \sum_{k=1}^K w_k A_k(x), \quad (4.3)$$

where K represents the number of rays, k is used to iterate through the rays, $A_k(x)$ represents the ambient value of the ray k , and w_k is an optional parameter that could be used to perform directional weighting. The obtained ambient factor $A(x)$ ranges from 0 (the voxel is fully occluded) to 1 (the voxel is completely unobstructed).

- II. **A “clipping” camera is introduced in the scene which renders the clipping mesh with two depth-only passes to two framebuffer objects (FBOs) using orthogonal projection.** The clipping camera is used to render this mesh from some point in the

scene, similar to how a light source generates a depth map in shadow mapping (see Figure 4.4). The front and the back faces of the mesh are rendered into each FBO correspondingly, and the value of empty pixels of Z_{front} is inverted. This way, for any voxel in the volume, it can easily be determined if it is clipped by transforming it to the coordinate space of the clipping camera. The two FBOs Z_{front} and Z_{back} are kept without change for all future rendering.

The 3D pose of the clipping mesh relative to the clipping camera should respect some constraints. If the clipping mesh is convex, then it should be positioned in the center of the clipping camera and have a margin around it to leave space for the dilation map generated in step III. If the mesh is concave, then an additional limitation should be respected, which is that the clipping mesh could be rendered with 2 depth peeling passes from the point of view of the clipping camera. If this limitation is not respected, the clipping mesh would cut regions similar to the convex hull of the concave mesh.

- III. **The Minkowski sum of the clipping mesh and the LAO ray cast region is rendered to two additional FBOs.** This Minkowski sum (Schneider, 2013) does not need to actually be represented with a mesh in order to be projected, it can be calculated in screen space from the two FBOs which contain the original projected mesh. These new textures containing the depth maps of the dilation shape are called D_{front} and D_{back} .

To calculate D_{front} from Z_{front} , a compute shader needs to be executed that will combine the depth map Z_{front} with a spherical kernel that represents the depth map of a sphere, as illustrated in Figure 4.7. The compute shader instantiates a thread for each pixel of the output depth map D_{front} , which has the same 2D dimensions as Z_{front} . The algorithm requires as input the radius of the sphere called r_{xy} which is measured in pixels, as well as the z-offset r_z which represents a depth in the range $[0, 1]$. These values are calculated by projecting the dilation radius into the coordinates of the carving camera and measuring the pixel length, as well as the depth. This dilation radius depends on the length of the rays used in step I. algorithm 4.1 represents the steps needed to be performed to

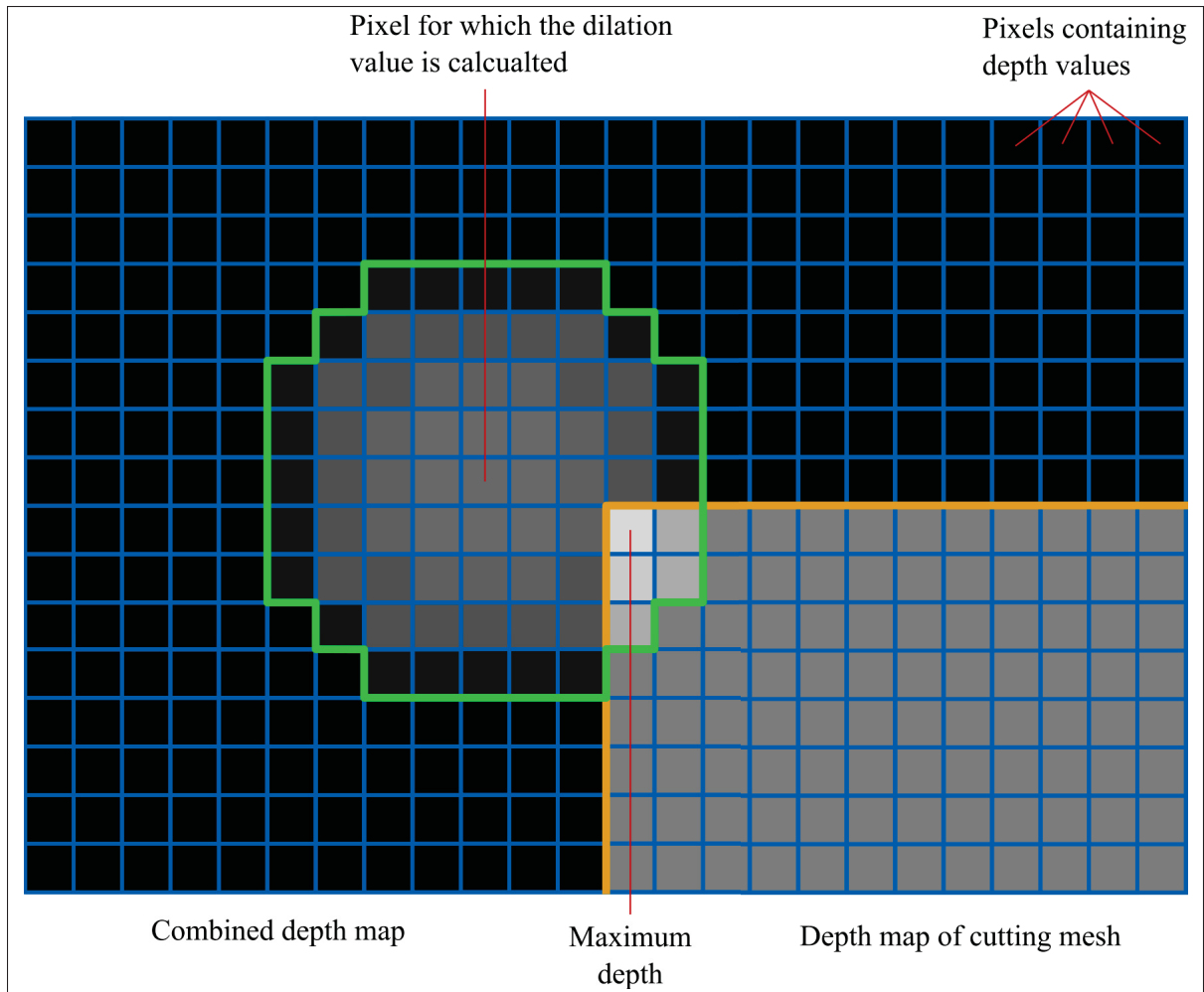


Figure 4.7 Computation performed for every pixel during the Minkowski sum calculation. The algorithm “inflates” the original depth map using a spherical depth map and stores the biggest value that was possible to obtain. Note that here, the creation of D_{back} from Z_{back} is illustrated

calculate a single depth value of D_{front} at pixel position (x, y) using Z_{front} . Then, the same process is repeated for the buffer D_{back} using Z_{back} with the exceptions that in line 7, z_{sphere} is added to z_{mesh} , and z_{min} is replaced with z_{max} , so the *max* function instead of the *min* one is applied at line 8.

Algorithm 4.1 Calculation of a dilated depth value executed for every pixel (x, y) of D_{front}

Input: Pixel at position (x, y) of the FBO D_{front}

Output: Updated FBO D_{front} at position (x, y)

```

1  $z_{min} \leftarrow +\infty;$ 
2 for  $i$  between  $x - r_{xy}$  and  $x + r_{xy}$  do
3   for  $j$  between  $y - r_{xy}$  and  $y + r_{xy}$  do
4     if  $(i, j)$  is within radius  $r_{xy}$  of  $(x, y)$  then
5        $z_{mesh} \leftarrow Z_{front}[i, j];$  /* Fetch  $Z_{front}$  at pixel  $(i, j)$  */
6        $z_{sphere} = r_z \cdot \sqrt{1 - \left(\frac{norm((x-i, y-j))}{r_{xy}}\right)^2};$ 
7        $z_{dilated} \leftarrow z_{mesh} - z_{sphere};$ 
8        $z_{min} = \min(z_{dilated}, z_{min});$ 
9     end if
10  end for
11 end for
12  $D_{front}[x, y] \leftarrow z_{min};$ 

```

4.3.5 Steps performed each frame

Each frame, the following computation is performed:

IV. **A compute shader clips the initial volume V and outputs a 3D texture V_o where all the clipped voxels are set to be transparent, while the visible voxels hold the opacity of the voxel. Additionally, the mask volume V_m is populated.** Here, the computer shader code is executed once for each voxel, and outputs either 0 (for zero opacity or full transparency), the original opacity of the voxel as if no clipping was applied, or the anti-aliased opacity value. Figure 4.8 illustrates the importance of using anti-aliasing when filling the V_o volume.

To calculate the anti-aliased opacity value, multisample anti-aliasing (MSAA) is used. First, for each voxel, one sample is taken in its center, and 8 samples are taken at the corners of the voxel. If it is determined that some of the samples around a voxels are clipped, and some are not, then N additional samples are taken at predefined positions in

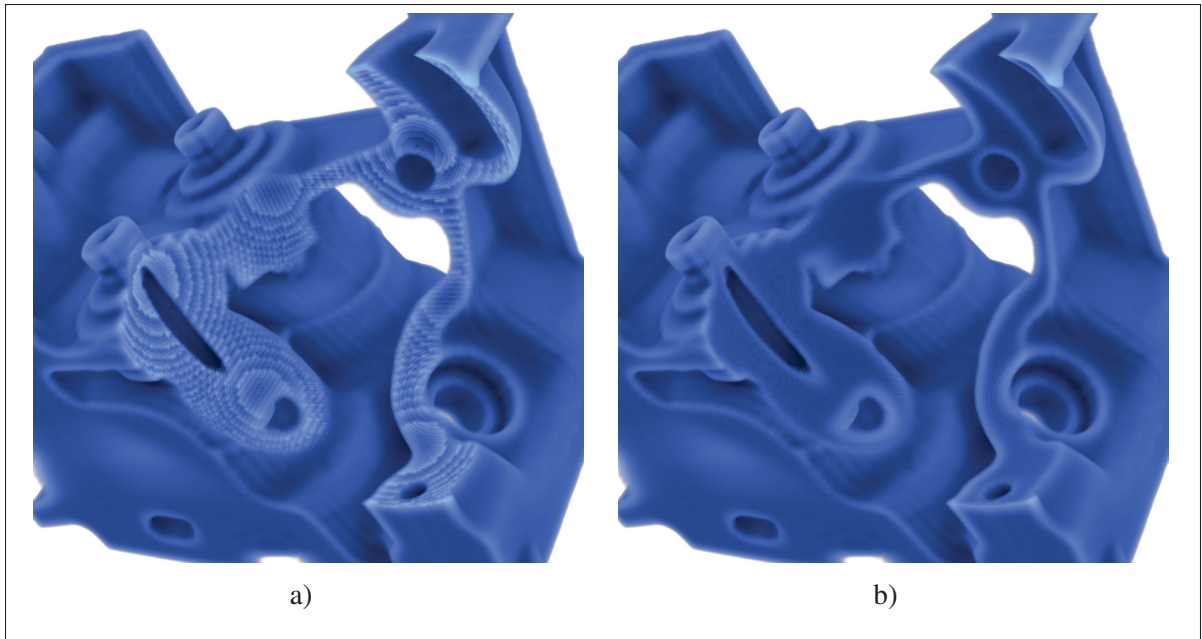


Figure 4.8 Comparison between CAO images created (a) without anti-aliasing and (b) with anti-aliasing with $N = 32$. Both images are rendered with a 54-ray spherical raycast

the $[-1, -1, -1]$ to the $[1, 1, 1]$ neighborhood of the voxel, for a total of $N + 9$ samples. algorithm 4.2 demonstrates this anti-aliased shadow test clipping procedure, which also includes populating the mask volume V_m in addition to filling the opacity volume V_o . Additional notes regarding some steps of the algorithm are provided in Appendix I.

- V. **A compute shader reads the mask volume, calculates the LAO factor in a different manner depending on the mask value V_m , and outputs the LAO factor to the V_{AO} volume.** If the voxel is of category a , then no calculation is performed and the ambient factor that is saved is 1. If the category is c , then no LAO calculation is performed either, and instead the precalculated V_{PAO} volume is sampled and its value is copied in the output V_{AO} volume as-is. Finally, if the category is b , only then the LAO calculation is performed for the voxel.

For each voxel in category b , rays are sent in K predefined directions, similarly to how it was made in step I for the volume V_{PAO} . The ambient factors of all rays are then averaged, and this value is written in the output 3D texture V_{AO} .

- VI. **The volume V is rendered with a ray casting algorithm, and for each sample of each ray, the V_{AO} volume is sampled to determine the ambient factor.** Here, the volume is rendered using standard ray casting, and at every sample of each ray, shadow test volume clipping using the Z_{front} and Z_{back} depth maps is used to determine if the sample is visible or not. If it is visible, then the V_{AO} volume is sampled.

4.3.6 Multiple simultaneous clipping meshes

The original CAO algorithm could be extended to support multiple clipping meshes that carve in the volume simultaneously, while being completely independent from each other. This can be done both to have multiple clipping meshes that move independently (as for example in the case where both hands control a mesh each) or to simulate a complicated concave clipping shape made out of simpler meshes, as long as each simpler mesh satisfies the constraints described in step II of subsection 4.3.4.

To support multiple clipping meshes, any mesh added to the scene should be associated with its own clipping camera that renders it to a different set of FBOs. Then, before any rendering is done, steps II and III of the CAO pipeline described in subsection 4.3.4 should be performed for each mesh with its associated clipping camera. This way, each mesh will have its own Z_{front} , Z_{back} , D_{front} and D_{back} buffers. Then, algorithm 4.2 of step IV should be modified so that Lines 19 to 23 are performed in a loop for all meshes that are carving in the volume. Additionally, the conditions of incrementing the counters $c_{visible}$ in Line 25 and $c_{dilated}$ in Line 27 should be modified to aggregate the results of depth tests from all meshes. In Lines 24 and 25, $c_{visible}$ must only be incremented if the sample is outside all meshes. This means that if the sample is inside at least one clipping shape, then it is considered as being invisible, but otherwise, it is visible. Similarly, in Lines 26 and 27, $c_{dilated}$ must be incremented only if the sample is within at least one dilation shape. That would mean that the ambient value of the associated voxel of this sample was affected by clipping of at least one mesh, and the voxel of the sample should be marked as requiring recalculation.

Algorithm 4.2 Anti-aliased shadow test clipping calculation executed for every voxel (x, y, z) of the original volume V

Input: Voxel at position (x, y, z) within the original volume V
Output: Updated opacity volume V_o and mask volume V_m at position (x, y)

```

1   $c_{visible} \leftarrow 0$ ;
2   $c_{dilated} \leftarrow 0$ ;
3  ShadowTestClipping(9 samples around voxel  $(x, y, z)$ );
4  if  $c_{visible} = 9$  then
5       $V_o[x, y, z] \leftarrow TF_o(x, y, z)$ ;
6  else
7      if  $c_{visible} = 0$  then
8           $V_o[x, y, z] \leftarrow 0$ ;
9      else
10         ShadowTestClipping( $N$  MSAA samples of  $(x, y, z)$ );
11          $V_o[x, y, z] \leftarrow \frac{c_{visible}}{N+9} \cdot TF_o(x, y, z)$ ;
12     end if
13 end if
14 if  $c_{dilated} = 0$  then
15      $V_m[x, y, z] \leftarrow 1$ ;
16 else if  $c_{visible} > 0$  then
17      $V_m[x, y, z] \leftarrow 0.5$ ;
18 else
19      $V_m[x, y, z] \leftarrow 0$ ;
20 end if

21 Function ShadowTestClipping(samples):
22     foreach sample  $(x_l, y_l, z_l)$  of samples do
23          $(x_c, y_c, z_c) \leftarrow \text{ApplyClippingCameraMVP}(x_l, y_l, z_l)$ ;
24          $z_{front} \leftarrow Z_{front}[x_c, y_c]$ ;
25          $z_{back} \leftarrow Z_{back}[x_c, y_c]$ ;
26          $d_{front} \leftarrow D_{front}[x_c, y_c]$ ;
27          $d_{back} \leftarrow D_{back}[x_c, y_c]$ ;
28         if  $z_c < z_{front}$  or  $z_c > z_{back}$  then
29              $c_{visible} \leftarrow c_{visible} + 1$ ;
30         end if
31         if  $d_{front} < z_c < d_{back}$  then
32              $c_{dilated} \leftarrow c_{dilated} + 1$ ;
33         end if
34     end foreach

```

4.4 Evaluation

To evaluate the CAO algorithm, 3 scenes were created, each featuring a different dataset with different parameters, as indicated in Table 4.2. The volumes were taken from the ImageVis3D 1 dataset⁴. In each scene, a volume was rendered, from which a region was clipped away. We used a single-color transfer function for each volume to better highlight the effect of LAO rendering on the dataset. The 3 rendered volumes are demonstrated in Figure 4.9 with a close-up, with 54-ray spherical ray cast used for LAO. The screenshots of the scenes using 6 and 54 ray configurations are available in the Appendix I Figure I-1, and close-ups of these screenshots are presented in the Appendix I Figure I-2. Regarding the number samples N used for anti-aliasing, we used 32 for a total of 41 samples including the 9 original ones taken for all voxels. To execute all our tests, we used a Windows 10 machine with an AMD Ryzen 7 5800X CPU, 32 GB of RAM and an AMD RX 6700XT GPU with 12 GB of VRAM.

Table 4.2 Scenes tested during the evaluation

Scene	Resolution	Samples Per Ray	Clipping Mesh	% Voxels Recalculated
Engine	$256 \times 256 \times 110$	20	sphere	14.94
Beetle	$416 \times 416 \times 247$	40	cube	20.24
Skeleton	$512 \times 512 \times 512$	50	rectang. prism	16.13

First, we evaluated the percentage of voxels that would need to be recalculated in our CAO algorithm, which we have also written in Table 4.2. This was measured by comparing the number of voxels within the V_m volume that were marked as requiring recalculation compared to the total number of voxels in the volume. As can be seen from that table, this percentage is relatively low, hovering around 15-20%. Note that the samples of the LAO rays have a distance of 1 mm between each other, which also corresponds to the Euclidian distance between any two neighboring voxels.

Second, we measured the rendering performance of the CAO algorithm and compared it to a naive implementation where LAO was recalculated for all voxels every frame, as can be seen in

⁴ <https://www.sci.utah.edu/software/imagevis3d.html>

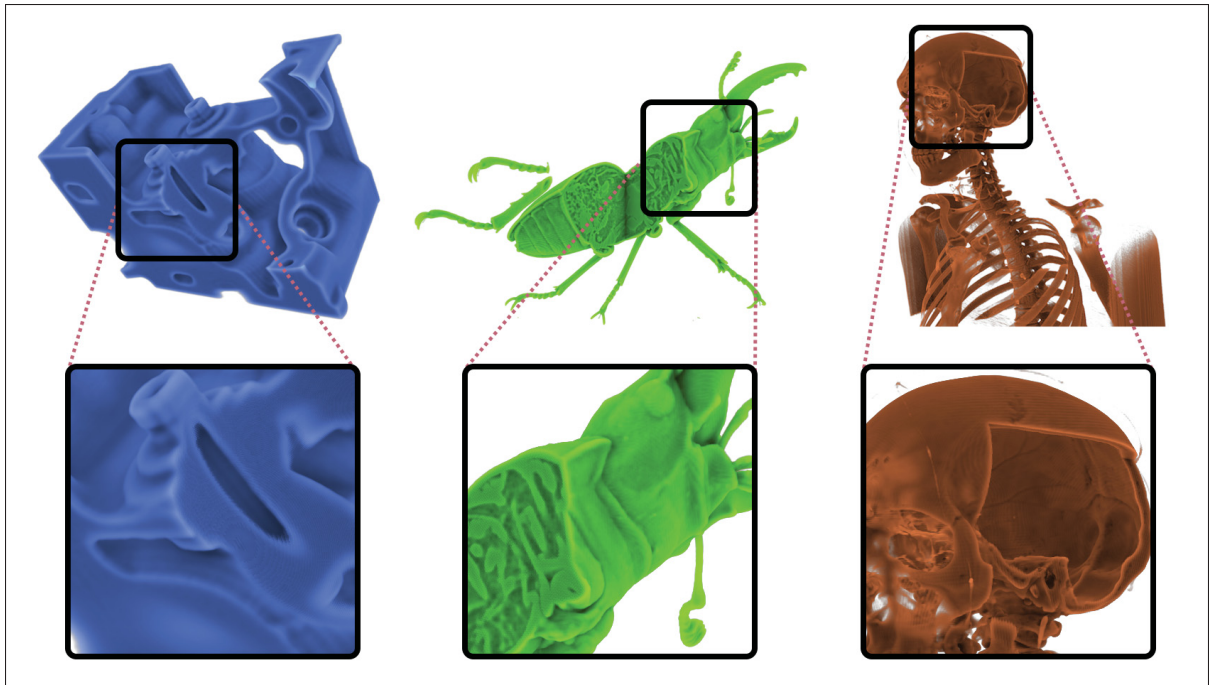


Figure 4.9 Scenes used for the evaluation of the CAO algorithm, as well as their close-ups. All presented images were created with 54-ray spherical ray cast. The left, middle and right images demonstrate the Engine, Beetle and Skeleton scenes correspondingly

Table 4.3. In the naive implementation, the V_m volume as well as the depth maps of the dilation shape D_{front} and D_{back} were not calculated. However, the output image produced by the naive LAO algorithm is exactly the same as the CAO one. Additionally, we compared the performance of the algorithm with Solid Color rendering where the same color was applied to all samples instead of LAO, but this implementation also featured shadow test volume clipping.

Third, we determined the computation cost breakdown when calculating a single frame of each of the 3 scenes using the LAO and CAO algorithms, with 6 and 54 rays, as can be seen in Figure 4.10. The graphs demonstrate that in terms of the amount of calculations, the longest step to perform along those described in subsection 4.3.5 is step V, where the spherical ray cast is performed. Further, with increasing dimensions of the volume that is rendered, the proportion of the step 5 computations becomes even larger, overshadowing steps IV and VI in terms of calculation time.

Table 4.3 Mean frames per second (FPS) for each scene. The speedup represents the ratio of the framerate of our algorithm (CAO) compared to a naive implementation (LAO)

Scene	Number of Rays	Solid Color	LAO - FPS	CAO - FPS	Speedup of CAO
Engine	6	272.846	111.549	184.902	1.66
	14		67.910	155.245	2.29
	26		42.831	126.378	2.95
	54		22.543	85.363	3.79
Beetle	6	128.288	15.497	42.581	2.75
	14		7.509	25.581	3.41
	26		4.192	16.063	3.83
	54		2.012	8.526	4.24
Skeleton	6	95.142	4.316	17.514	4.06
	14		2.020	9.594	4.75
	26		1.107	5.729	5.18
	54		0.529	2.825	5.34

Additionally, we have evaluated the overhead that having multiple clipping meshes has on the rendering performance. We created a modified version of the 3 scenes described previously where we added a spherical mesh that also clips each volume. The modified Engine scene can be seen in Figure 4.11, while screenshots of all scenes can be seen in the Appendix I Figure I-3. In all 3 scenes, we have positioned the new mesh so that it overlaps with the already existing clipping mesh, to test our aggregation function of samples from different depth and dilation maps. The comparison between the frame rates of having one or two clipping meshes can be seen in Table 4.4. As can be seen from these results, using an additional clipping mesh results in a slowdown hovering around 8 – 16%. Moreover, we can see that the loss of performance is more significant (ranging from 14.3% to 15.7%) for a smaller datasets, as in the Engine scene for example, compared to bigger datasets such as the Skeleton (where the slowdown ranges from 8.3% to 10.5%). We hypothesize that the reason for this difference is that larger datasets have a much longer execution of the spherical ray casting step (step V of the algorithm), as can be seen from the computation cost breakdown above. Therefore, even though additional clipping would also affect the amount of calculations needed to be performed in step V, the main step affected by the algorithm is step IV due to the doubling of the amount of sampling that has to be

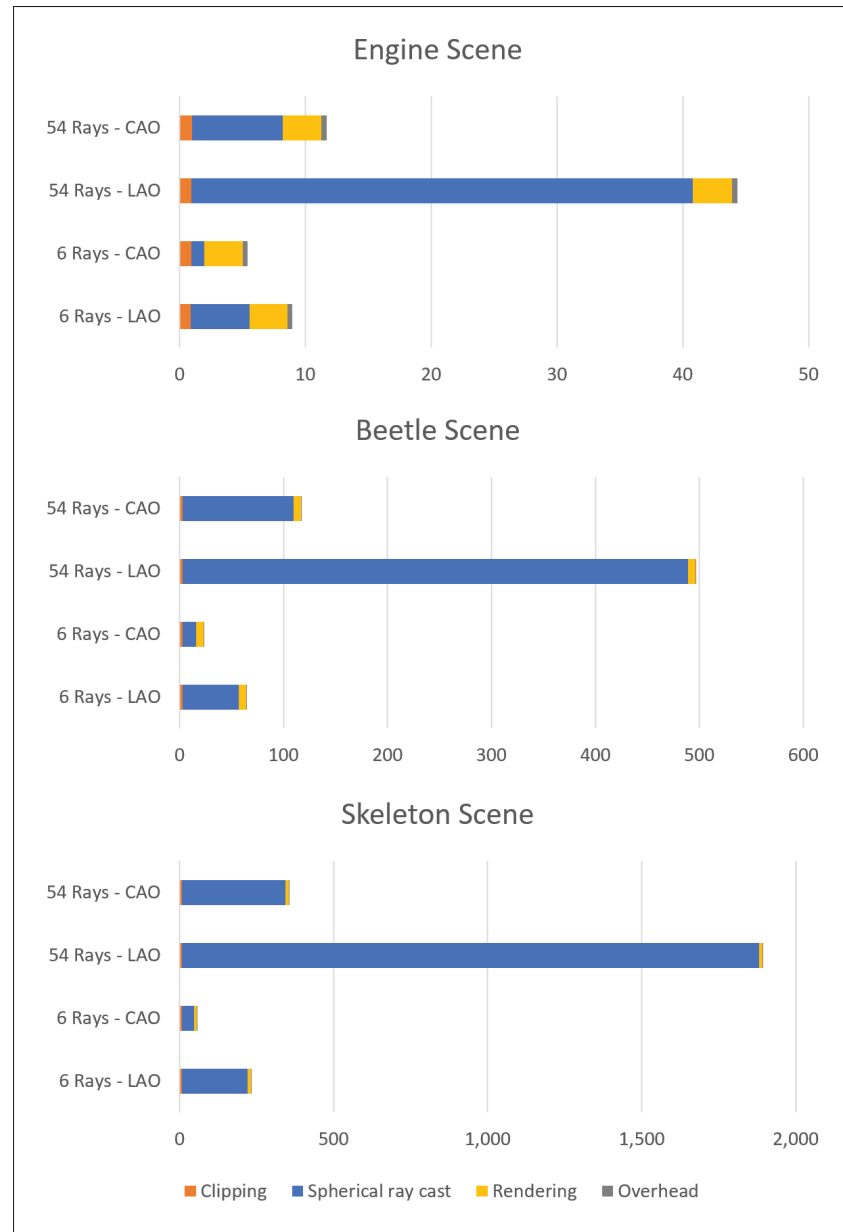


Figure 4.10 Runtime computation cost breakdown for a single frame using the LAO and CAO algorithms, in milliseconds (ms). "Clipping" corresponds to the step IV of the algorithm explained in subsection 4.3.5, "Spherical ray cast" corresponds to step V, "Rendering" corresponds to step VI, and "Overhead" represents time spent by the Unity3D engine on calculations not related to AO

performed for each voxel sample. Thus, smaller datasets where step IV takes a larger proportion of the execution time would be slowed down more. Another important aspect that can be noted from Table 4.4 is that the number of rays has little impact on the slow down time, which is to be expected because it mainly affects step V, which isn't directly affected by changes to the number of clipping meshes used in the scene.

Table 4.4 Performance comparison between using 1 and 2 clipping meshes in each scene, as well as the corresponding slowdown percentage

Scene	Number of Rays	CAO 1 mesh - FPS	CAO 2 meshes - FPS	% Slowdown
Engine	6	184.902	158.508	14.3
	14	155.245	132.542	14.6
	26	126.378	107.000	15.3
	54	85.363	71.936	15.7
Beetle	6	42.581	37.690	11.5
	14	25.581	23.097	9.7
	26	16.063	14.150	11.9
	54	8.526	7.487	12.2
Skeleton	6	17.514	15.717	10.3
	14	9.594	8.800	8.3
	26	5.729	5.173	9.7
	54	2.825	2.527	10.5

Furthermore, we evaluated the effect that the number of rays has on the quality of the created image. This was done to demonstrate the importance of having a high number of rays during the spherical ray casting, which is specifically the case where CAO performs particularly better compared to LAO. We measured this by calculating the Root Mean Square (RMS) between the indicated number of rays and the ground truth image which was calculated using 512 rays which were distributed using the Fibonacci Lattice (Keinert, Innmann, Sanger & Stamminger, 2015) on the surface of a sphere. The ray patterns were chosen according to the 26-neighborhood of a voxel, with a jittering factor added to increase the randomness. For the numbers of rays, they were chosen so that they approximately double each time. In the case of 6 rays, the directions with a Manhattan distance of 1 were chosen, while for 14 rays, directions of a Manhattan distance of 1 and 3 were chosen. For the 26 rays case, all 26 neighboring directions were considered, while for 54 rays, the centers of all cube faces of a Rubik's cube were chosen.

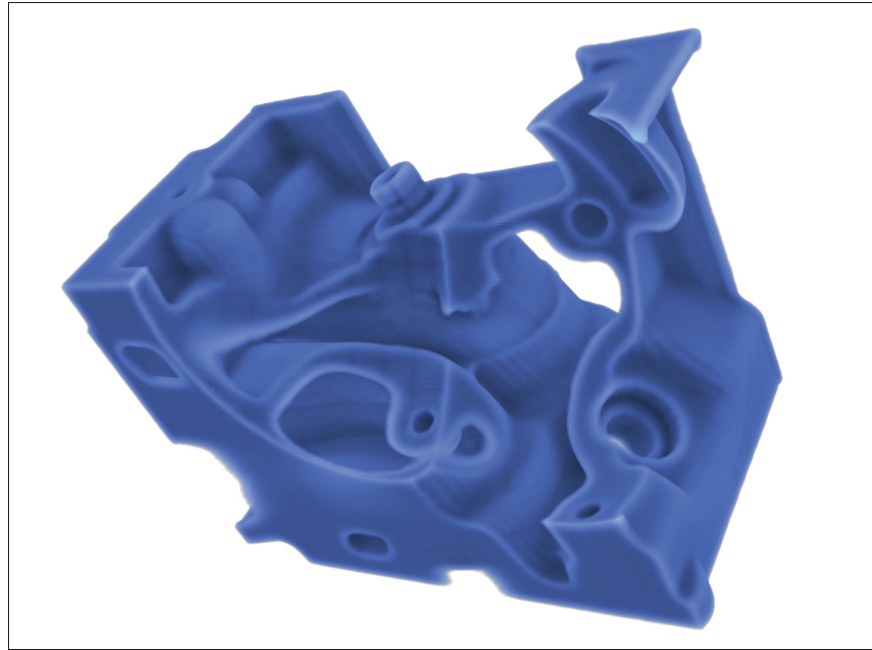


Figure 4.11 Modified Engine scene with two clipping meshes

Table 4.5 Root Mean Square per-voxel (3D) difference between the ambient occlusion factor obtained with a different number of rays and the AO factor obtained for the “ground truth” volume obtained with a large number of rays (512), for each scene. Range: [0, 1]

Scene	6 Rays	14 Rays	26 Rays	54 Rays
Engine	0.03907	0.01203	$9.759 * 10^{-3}$	$6.937 * 10^{-3}$
Beetle	0.01746	$8.765 * 10^{-3}$	$5.144 * 10^{-3}$	$3.438 * 10^{-3}$
Skeleton	0.02520	0.01205	$7.910 * 10^{-3}$	$5.208 * 10^{-3}$

The RMS was calculated both for the voxel values indicated in Table 4.5, as well as the final color of the pixels in the rendered image as indicated in Table 4.6. Additionally, we made a per-pixel demonstration of the error between each number of rays and the ground truth image, which we have shown in the Appendix I Figure I-4. We also added an example that demonstrates the difference in error for the Engine scene in Figure 4.12. The colors in these images were inverted to better demonstrate the error. We performed this comparison to demonstrate the importance of having a high number of rays, which is also where our algorithm has the highest speedup.

Table 4.6 Root Mean Square per-pixel (2D) difference between the image pixel values obtained with a given number of rays and the pixel values obtained for the “ground truth” image obtained with a large number of rays (512), for each scene.

Range: [0, 255]. Each pixel component was treated as an individual value

Scene	6 Rays	14 Rays	26 Rays	54 Rays
Engine	11.771	2.755	1.936	1.032
Beetle	4.485	1.950	1.072	0.848
Skeleton	5.292	1.603	1.134	0.956

Finally, knowing that the algorithm produces exactly the same image as LAO, to make sure that our implementation is correct, we compared the screenshots produced by the LAO and CAO algorithms for the 3 scenes. We have created screenshots using 6, 14, 26 and 54 ray configurations and we made a per-pixel comparison between the LAO and CAO images for each configuration. We have found that every pair of pixels had the exact same *RGB* components, confirming the correctness of our implementation. This result was expected because our CAO algorithm dynamically determines every frame what voxels could have a LAO value that is affected by clipping, and performs a recalculation on these voxels. Thus, with a correct implementation, it is impossible to obtain a different AO value for LAO and CAO.

4.5 Discussion

The most important advantage of our algorithm is that it offers exactly the same quality of image as the LAO algorithm presented by Hernell *et al.* (2010), while reducing AO factor recomputation time when clipping parameters are modified. This algorithm produces a physically accurate ambient factor, as it follows the absorption model described by Max (1995). We propose a more sophisticated rendering pipeline for the LAO algorithm that is able to take care of carving within the volume while maintaining proper recalculation of the ambient factor. Additionally, our algorithm supports clipping with all convex shapes and concave shapes that can be rendered with two opacity peeling steps (Everitt, 2001).

When compared to LAO, it can be seen from Table 4.3 that our algorithm offers a significant speedup, with 2-5x the frame rate compared to a full recalculation of all voxels. It can also be seen that the speedup is higher for volumes with a higher 3D resolution and with a higher number of rays. This is expected because in those cases the total amount of computation necessary for the spherical ray cast becomes larger, so cutting down the amount of computation at this step yields the best speedup, as can be seen in Figure 4.10. For example, for our largest dataset (Skeleton), for the 26 and 54 ray cases, the speedup was 5.18 and 5.34 correspondingly. Thus, our algorithm is especially useful when a high-quality ambient factor needs to be calculated every frame with a volume with high dimensions. To highlight the importance of using a high number of rays, we have also measured the effect of that number on the quality of the obtained image. As can be seen from Table 4.6, the 6 and even the 14-ray patterns have a relatively high error, and to get to a mean error that is below 1 within the $[0, 255]$ color range, one needs to use at least a 54-ray configuration. The difference of the 2D error between the 6-ray configuration and the 54-ray configuration from the ground truth image made with 512 LAO rays can be seen in Figure 4.12.

Additionally, since the calculation is done in data space, it avoids the stereoscopic screen-space inconsistency described by Shi *et al.* (2022), which can result in viewer discomfort. This inconsistency exists in screen-space algorithms such as SSAO due to the fact the left and right eye view slightly different parts of the dataset, which might consequently lead to a different calculated ambient value for the same object in each eye.

The algorithm that we presented is particularly suitable for interactive environments where the position and rotation of the clipping region might change every frame. Unlike the algorithm presented by Hernell *et al.* (2010) in which only one LAO ray is cast per frame, in our implementation a full ray cast is performed every frame. Because of this, the user always perceives the final LAO image without having blurring or visual artifacts from previous frames, which makes it more suitable for scenarios where the clipping region is controlled by direct user input. Such is the case of interactive VR and AR environments where the clipping region is controlled with hand gestures or controllers with 6 degrees of freedom, so small changes to

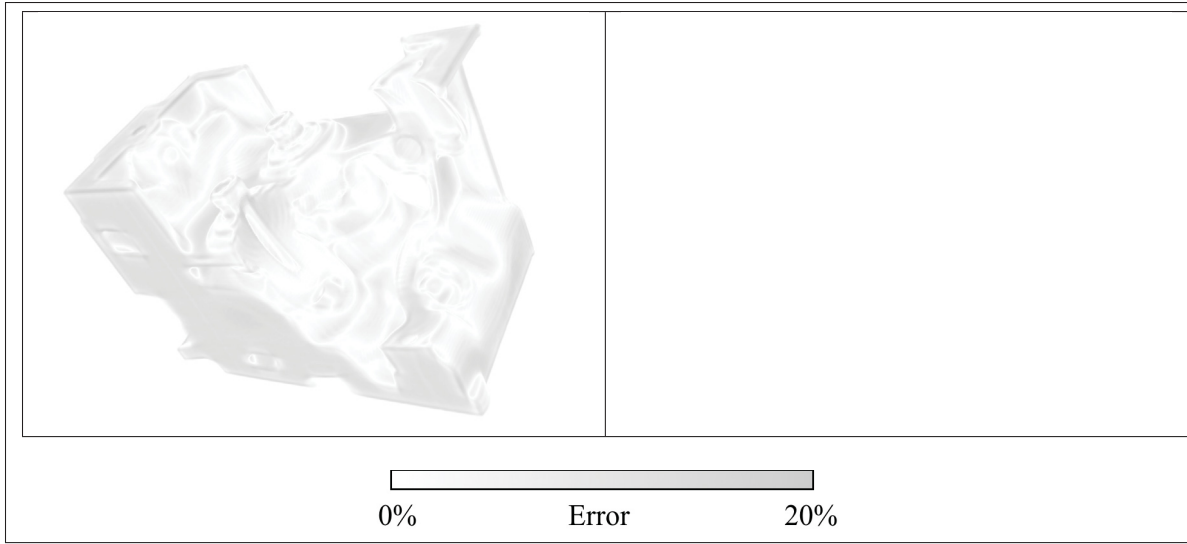


Figure 4.12 The 2D error (pixel value difference) between the image rendered with the 6-ray configuration and the ground truth image with 512 rays (left), as well as the error between the 54-ray configuration and the same ground truth image (right). Each pixel color aggregates the average error between the RGB components. The pixel colors are inverted, and thus darker colors represent higher error. The error ranges between 0% and 20%, with 20% being the highest recorded error for a single pixel in the images

the clipping region could occur every frame due to hand shaking. In addition to this, the AO calculation performance scales well with increased number of viewports and the resolution in each of the viewports. This is due to the fact that CAO is calculated in data space, but also because the rendering of the volume is done with shadow test volume clipping, which avoids needing to generate the depth map for every viewport like it is done by Weiskopf *et al.* (2003). This further makes the algorithm suitable for VR and AR displays which may have more than one viewport and a relatively high resolution for each eye.

The main drawback of the algorithm is that it requires a significant amount of video memory, especially when rendering volumes with a high spatial resolution. As can be seen in Figure 4.3, there are 5 volumes with the same dimensions as the dataset that have to be kept in video memory simultaneously. Another drawback is that any change to the opacity transfer function would require a recomputation of the volume V_{PAO} that stores the precalculated LAO values. In that case, the framerate of the algorithm would decrease to the one shown for the LAO algorithm in

Table 4.3. However, as can be seen from this table, even in that case the framerate will remain highly interactive, and only the configurations where the dataset has high dimensions and a high number of rays would be severely affected.

4.6 Conclusion

In this paper we have presented a novel volumetric ambient occlusion algorithm that was optimized to work with real-time clipping of the volume. The code of the algorithm was published on GitHub in the form of a Unity3D package, with the link given in the Methods section. In this algorithm, we determine during each frame which voxels' ambient value was affected by clipping and we perform an LAO recomputation only on these voxels. Since those affected voxels are located in a small radius around the clipping shape, their number is significantly lower than the total number of voxels in the volume. Our algorithm is able to find these voxels by calculating the dilated depth map of the clipping volume, and then by using shadow test volume clipping to determine if the voxels require recomputation. The algorithm offers a 2-5x speedup over recalculating the ambient factor of all voxels every frame. Additionally, a simple extension can be made to the algorithm that would make it support multiple simultaneous clipping meshes. Furthermore, the algorithm is suitable for interactive environments since the final image is generated every frame, and it is suitable for VR and AR environments since it does not have any stereoscopic inconsistency.

In the future, more tests should be made to measure the effect of different parameters on the quality of the rendering. A user study could be performed to measure the effect of the number of rays, to determine the right compromise between the quality of the rendered image and the performance. Additionally, the effect of the resolution of the FBOs of the clipping camera needs to be evaluated both in terms of the quality of the created image, as well as the performance. Moreover, the use of more complex concave meshes could be explored that are not made out of multiple simpler meshes. With these complex concave meshes, P depth peeling passes may be required, in which case P depth and dilation buffers should be used, similar to how concave mesh clipping was described by Weiskopf *et al.* (2003). Further, a more rigorous evaluation

should be done to determine how our algorithm combined with interactive clipping helps the perception of volumetric data compared to more basic volume rendering techniques such as Phong shading (Phong, 1975) or using a simple 1-dimensional transfer function.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

This work was supported by the NSERC, Discovery program, RGPIN-2020-05084, as well as by the Fonds de recherche du Québec – Nature et technologies (B2X Doctoral Scholarship - 334501, <https://doi.org/10.69777/334501>).

CHAPTER 5

ANATOMYCARVE: A VR OCCLUSION MANAGEMENT TECHNIQUE FOR MEDICAL IMAGES BASED ON SEGMENT-AWARE CLIPPING

Andrey Titov^{1,2}, Tina N. H. Nantenaina¹, Marta Kersten-Oertel², Simon Drouin¹

¹ Department of Software and IT Engineering, École de Technologie Supérieure,
1100 Notre-Dame Ouest, Montréal, Québec, Canada H3C 1K3

² Gina Cody School of Engineering and Computer Science, Concordia University,
1455 De Maisonneuve Blvd. West, Montréal, Québec, Canada H3G 1M8

Article submitted to the journal "IEEE Transactions on Visualization and Computer Graphics"
in August 2025

5.1 Introduction

Visualizing three-dimensional medical images, such as those produced by MRI or CT scanners, poses significant challenges in effectively depicting internal anatomy primarily due to self-occlusion. In the context of anatomical images, self-occlusion occurs when certain anatomical structures obscure others within the body. For example, outer layers like skin or fat can obscure deeper organs, making it difficult to discern internal anatomy and understand spatial relationships.

Various techniques have been developed to address this issue, including axis-aligned slicing (Joshi *et al.*, 2008), arbitrary slicing (Laha & Bowman, 2013), cut-away views (Viola *et al.*, 2004), focus and context techniques (Wang, Zhao, Mueller & Kaufman, 2005), interactive clipping (Joshi *et al.*, 2008; Laha & Bowman, 2013), and deformations (McGuffin *et al.*, 2003). These techniques aim to provide interactive visualizations that reveal different anatomical structures while preserving their spatial relationships within the body. However, they often fail to deliver intuitive, context-preserving views of complex anatomy, limiting their effectiveness for education and surgical planning.

To address these limitations, we introduce *AnatomyCarve*, a virtual-reality-based (VR) tool inspired by traditional anatomical illustrations, such as *Gray's Anatomy* (Standring, 2015).

AnatomyCarve uses segment-specific clipping to reveal internal structures while preserving spatial context. Its interactive VR environment allows users to manipulate individual anatomical structures in real time, maintaining their spatial relationships and producing intuitive, immersive visualizations (see Figure 5.1).

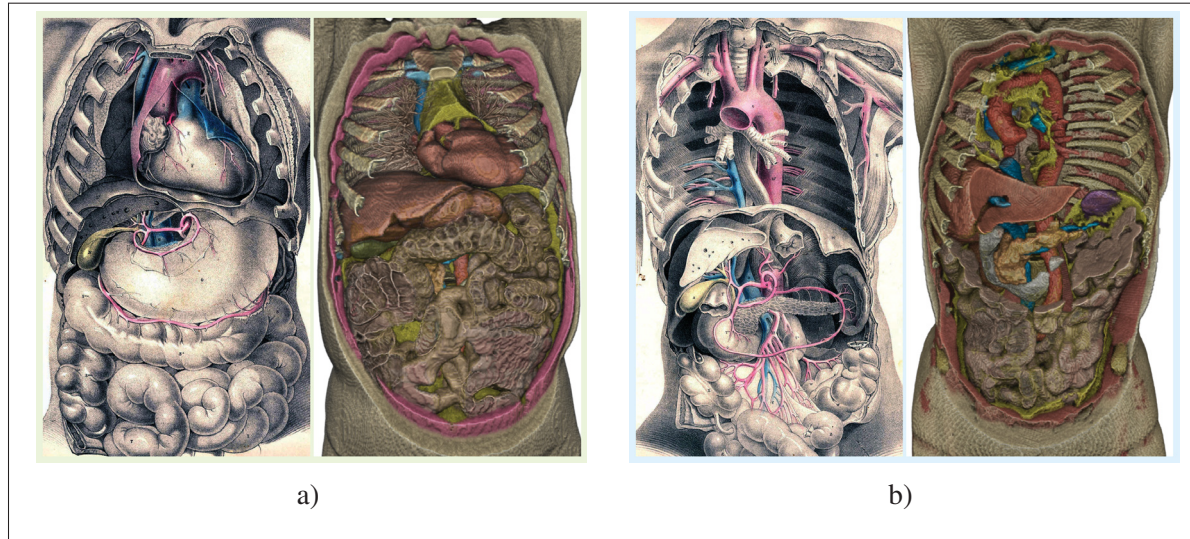


Figure 5.1 Comparison between anatomical book illustrations on the left panels in (a) and (b), and their recreation with our tool, *AnatomyCarve*, on the right panels in (a) and (b) correspondingly¹

To the best of our knowledge, no existing method enables per-organ or per-segment interactive clipping based on the concept of *stepwise dissection* (Koutsarnakis, Liakos, Kalyvas, Sakas & Stranjalis, 2015), which we adopt in *AnatomyCarve*. Stepwise dissection is a visualization strategy in which anatomical layers are removed sequentially to reveal deeper structures of interest (Komaitis *et al.*, 2022). Typically, this process begins with the removal of skin, followed by fat, fascia, and muscles, and finally location-specific structures such as organs or vessels (Cotofana & Lachman, 2020). *AnatomyCarve* translates this approach into an interactive 3D environment by allowing users to sequentially insert multiple clipping shapes, each targeting a distinct subset of anatomical structures, effectively emulating a real dissection process.

¹ Illustrations on the left panels of (a) and (b) adapted from J. Maclise, *Surgical Anatomy* (1859), public domain (Maclise, 1859)

By combining segment-specific clipping, stepwise dissection, and VR interactivity, AnatomyCarve provides a novel platform for both educational and clinical applications. It enables the creation of high-quality, textbook-style medical illustrations, supports immersive learning of complex anatomy, and offers surgeons a tool for exploring patient-specific anatomy in detail, potentially improving surgical planning and precision. In this paper, we demonstrate how AnatomyCarve enhances these tasks while maintaining real-time interactivity and spatial context.

5.2 Background and Related Work

5.2.1 Anatomical Illustration

Anatomical illustration has long been essential in medical education and communication, serving both as a pedagogical tool for conveying complex spatial relationships and as an artistic medium that bridges science with visual storytelling (Weiss & Casazza, 2023). Educationally, illustrations can simplify intricate anatomical concepts, enhance memory retention, and support learners at different stages of training (Papa, Varotto, Vaccarezza & Galassi, 2021). Artistically, they offer a unique lens for interpreting and representing the human body, contributing to both scientific understanding and cultural heritage (Calkins, Franciosi & Kolesari, 1999; Naicker, 2023).

Traditionally, such illustrations are task-driven, designed to highlight specific organs or tissues relevant to a given context (Preim & Bartz, 2007, p. 419). To manage occlusion and maintain clarity, less important elements may be clipped, removed, or deformed depending on their pedagogical role. The artist's decisions involve balancing visibility, spatial context, and visual simplicity, preserving critical reference structures, partially removing others, and completely omitting those unrelated to the focus. For example, in the left panels of Figure 5.1a and Figure 5.1b, ribs are partially retained to provide spatial context, while outer tissues such as muscles and skin are removed to avoid obstructing key organs. In some cases, organs are cut to reveal internal anatomy, as with the liver in Figure 5.1b. Deformation can also be employed; for instance, repositioning the bowel outside the body to expose underlying structures, as in the same figure. The emphasis changes with the instructional goal: Figure 5.1a focuses on the

digestive system, while Figure 5.1b centers on the cardiovascular system. AnatomyCarve enables the creation of similar task-dependent, context-preserving visualizations, as demonstrated in the right panels of Figure 5.1a and Figure 5.1b, translating the design strategies of traditional anatomical illustration into an interactive, 3D medium.

5.2.2 Occlusion Management Techniques

A number of solutions have been proposed for solving the self-occlusion problem, commonly referred to as occlusion management techniques (Elmqvist & Tsigas, 2008) (see reference for detailed review). In medical imaging, these techniques typically fall into two main categories identified by Elmqvist & Tsigas (2008): virtual X-ray methods, which reveal internal structures by simulating transparency or projection, and volumetric probe methods, which allow users to explore the interior of a volume interactively. Within both categories, techniques can further be distinguished based on whether they operate on segmented or unsegmented data.

5.2.2.1 Virtual X-ray

The Virtual X-ray paradigm includes all techniques that make occluding geometry semi-transparent or invisible (Elmqvist & Tsigas, 2008). One of the earliest and most basic methods of occlusion management in this category is slicing, which samples planar slices from a 3D volume and displays them in 2D (Joshi *et al.*, 2008). The simplest form, axis-aligned slicing, aligns slicing planes perpendicular to the three primary axes, sampling the volume at regular intervals. A variation, arbitrary slicing, allows for slices along any 2D plane (Laha & Bowman, 2013). Both approaches can handle segmented and unsegmented data, but they make it difficult to fully comprehend 3D structures, particularly elongated or convoluted anatomical features, such as electrodes or blood vessels (Preim & Bartz, 2007; Joshi *et al.*, 2008, p. 155).

Clipping is an alternative approach that hides portions of the dataset to focus on smaller, relevant subsets (Preim & Bartz, 2007, p. 291). By removing outer layers, clipping reduces visual complexity and exposes internal structures (Joshi *et al.*, 2008; Drouin & Collins, 2018). The

simplest form is axis-aligned cropping, which may remove too much surrounding context, reducing spatial understanding.

Joshi *et al.* (2008) explored irregular cropping for unsegmented anatomical volumes in neurosurgery. In their system, the user could insert 3D convex shapes into the volume; voxels inside the shape were excluded from rendering during ray casting. Irregular cropping preserves more context than axis-aligned slicing by allowing selective removal of data, but it only exposes surface geometry of internal structures rather than their full 3D shape (Figure 5.2).

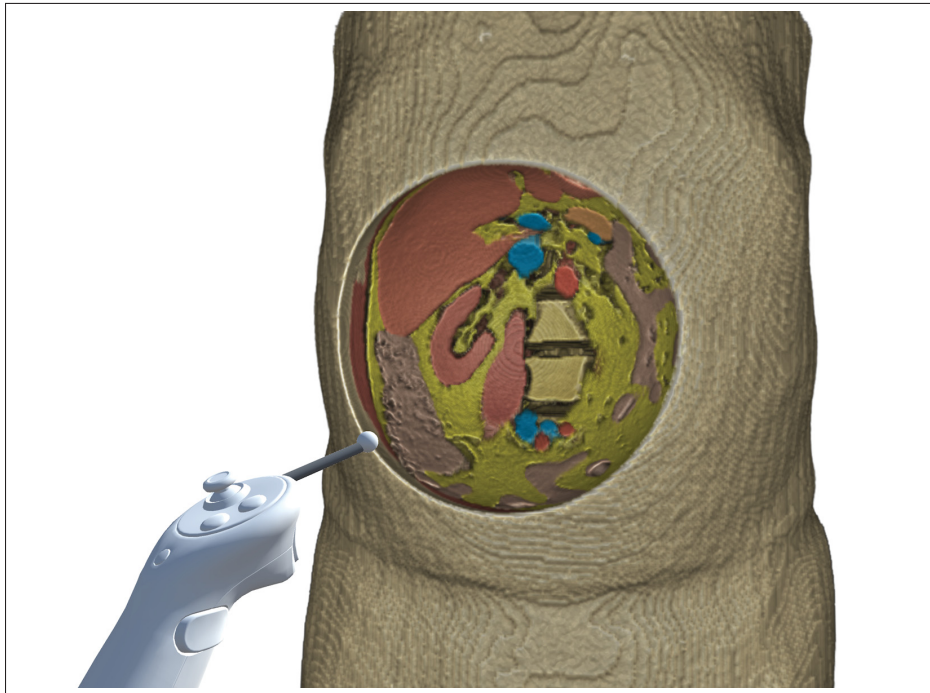


Figure 5.2 A simulation of the Joshi *et al.* (2008) clipping algorithm within AnatomyCarve. Note that the clipping does not discriminate between the segments and clips all the voxels within the spherical region

Segment-aware extensions further improve context preservation. Ajani, Bharadwaj & Krishnan (2018) introduced volumetric clipping surfaces for segmented data, where a 2D spline surface clips voxels on one side except for user-selected segments. The surface wraps selected segments and smoothly recedes into surrounding anatomy, like an invisible veil. Similarly, Díaz, Monclús, Navazo & Vázquez (2012) proposed interactive segment expansion within clipped planes: the

user inserts a clipping plane, hides voxels on one side, selects a visible pixel to identify its segment, and then expands that segment back into the clipped region. Users can combine multiple planes for complex visualizations. Both techniques, however, remain limited to planar-alike cropping for non-selected segments, a key limitation of the Virtual X-ray pattern.

Another approach, context-preserving “ghosting” (Bruckner, Grimm, Kanitsar & Groller, 2006), modulates opacity based on intensity, gradient, and eye distance to reveal internal structures without removing surrounding context. While effective at maintaining overall structure, overlapping semi-transparent layers can reduce shape clarity and hinder depth perception.

5.2.2.2 Volumetric Probe

The Volumetric Probe paradigm encompasses techniques where a user places a probe into a dataset to locally deform or remove geometry, generally classified as deformation-based methods. McGuffin *et al.* (2003) implemented a system allowing the user to select a region within a volume and deform either the entire volume or specific segments. This approach reveals hidden interior structures while preserving most context and occlusion cues. However, spatial relationships between structures are altered, making it harder to accurately perceive anatomical organization. He *et al.* (2017) extended this paradigm to hierarchical brain atlases, enabling exploded views of anatomical regions to facilitate visualization of densely packed structures. Users generate spline curves from controller positions and flip through segments along the trajectory. Similarly, Laha & Bowman (2013) designed Volume Cracker, a gesture-based system where a splitting plane positioned between the hands separates the volume into two parts. Initially designed for gloves and controllers, it was later adapted for bare-hand interaction (Laha *et al.*, 2016). Both systems inherit the limitations of Volumetric Probe methods: despite improved access to hidden structures, deformations alter original spatial positions, making it difficult to perceive accurate 3D relationships between anatomical segments.

Building on previous work, we present a novel interactive occlusion management method that combines the benefits of interactive clipping and deformation while avoiding their respective

limitations. Our method generalizes the approach of Díaz *et al.* (2012) in two key ways: (1) we support arbitrary 3D clipping shapes, rather than being restricted to 2D planes, and (2) we adopt a stepwise dissection interaction paradigm, rather than sequentially hiding and revealing segments. Using spherical clipping shapes allows users to expose precisely the internal structures of interest while preserving context and occlusion depth cues. Unlike deformation-based techniques, our approach does not alter the spatial positions of anatomical structures, fully maintaining their true spatial relationships. These innovations make our method uniquely suited for creating interactive, context-preserving visualizations of complex 3D anatomy.

5.3 Methods

5.3.1 Core Concept

The core concept of AnatomyCarve is illustrated in Figure 5.3. The system begins by rendering the complete anatomical volume. Users then define spherical clipping regions that selectively remove voxels belonging to a user-specified subset of anatomical segments. For example, in Figure 5.3, the first spherical region (indicated by the blue dotted outline) is configured to remove voxels corresponding to the skin, fat, muscle, and liver segments, while preserving voxels from the stomach, heart, and spine. Each spherical region is defined and positioned interactively, with the user able to modify the associated segment-selection set in real time before confirming the region’s parameters. Once fixed, additional spherical regions can be introduced, each with its own independent segment-selection configuration. The method imposes no restriction on the number of regions that can be created, enabling complex, task-specific occlusion management while maintaining spatial context for the preserved structures.

5.3.2 Interaction Paradigm

AnatomyCarve was developed specifically for VR to leverage its natural interaction and immersive 3D visualization capabilities. VR headsets with six degrees of freedom (6DoF) controllers enable precise positioning of clipping shapes, offering far greater flexibility than traditional 2D

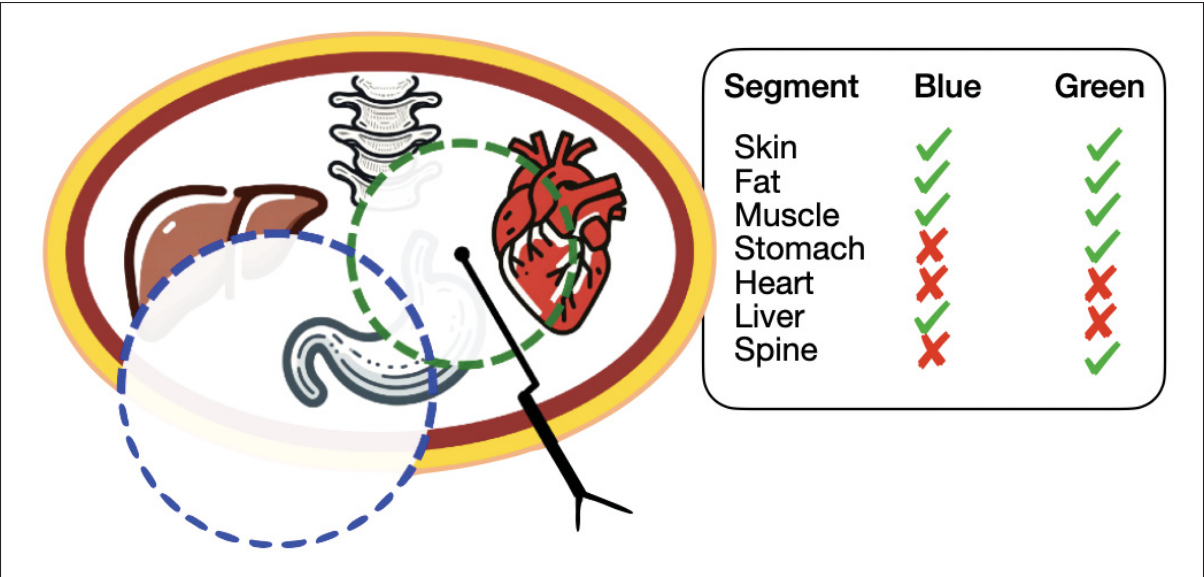


Figure 5.3 A carving configuration using two clipping spheres. The blue sphere removes the skin, fat, muscle, and liver segments. The green sphere also removes the skin, fat, and muscle, but instead of the liver, it removes the stomach and spine segments. The accompanying table lists the clipping masks for each sphere, where a check mark denotes a clipped segment and a cross denotes a segment that remains visible

input devices such as a mouse (Jerald, 2015). Depth cues from stereoscopy and motion parallax further improve the accuracy of clipping and segmentation (Vienne *et al.*, 2020), allowing users to create detailed, context-preserving views of anatomical volumes.

The system employs an asymmetrical interaction model, assigning distinct roles to each hand. One hand controls an invisible clipping sphere that hides all clippable voxels inside it, while the other hand selects which anatomical segments are affected by the sphere. A demonstration video accompanying this article illustrates these interactions in practice.

As shown in Figure 5.4a, the system operates on a segmented volume. The user can select anatomical segments from those present in the dataset (e.g., the colon, Figure 5.4b) and then insert a clipping sphere to remove the voxels inside it, following an approach similar to Joshi *et al.* (2008). Segments marked as unclippable remain fully visible (Figure 5.4c). Once positioned,

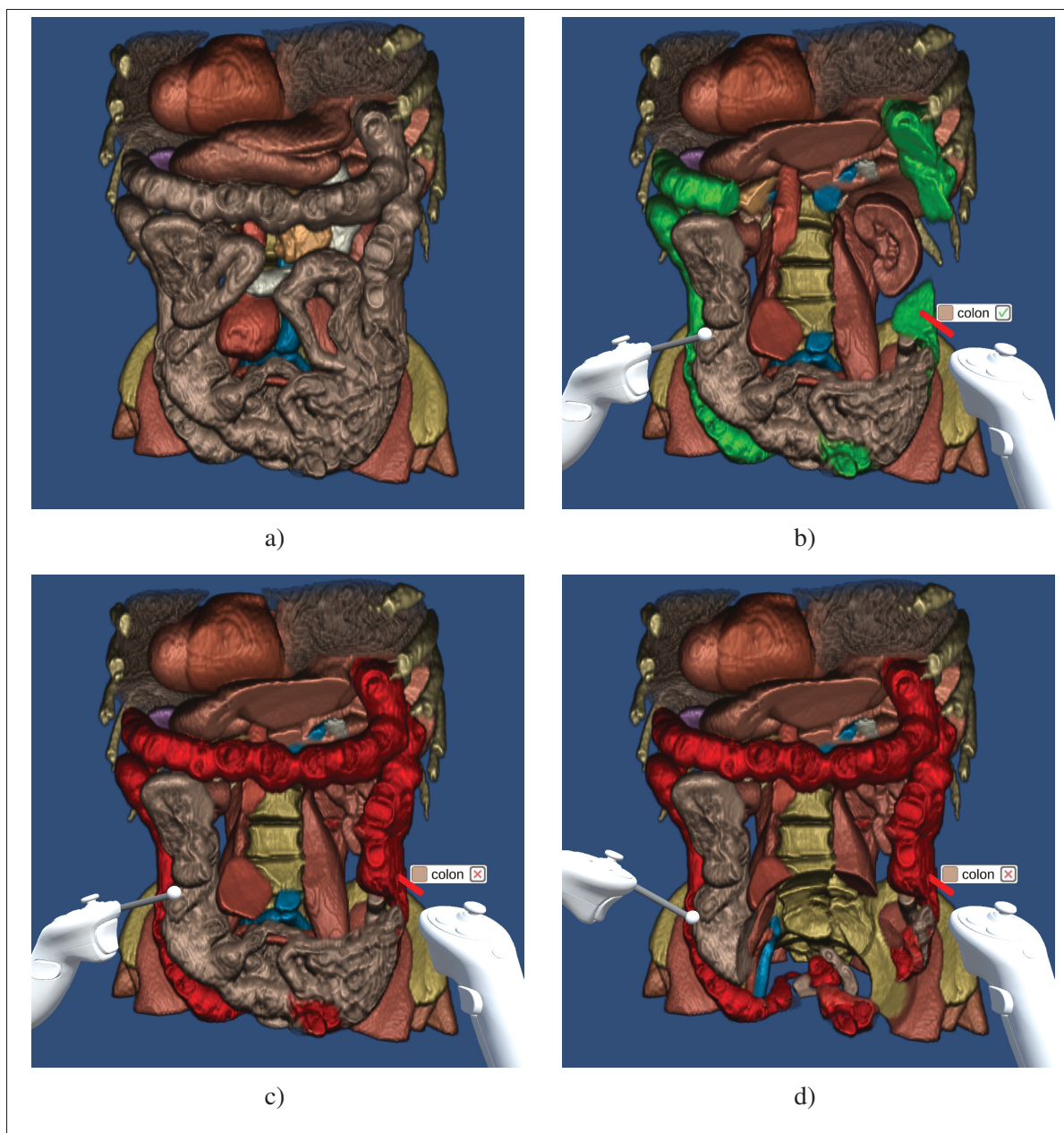


Figure 5.4 The basic functionality of AnatomyCarve. (a) A segmented medical volume where no clipping is applied is first shown. (b) A user inserts the clipping sphere in the volume, which hides all the voxels from all segments inside of it, and the user can simultaneously select segments on it. Then, in (c), the user makes the “colon” segment unclippable, so all of its voxels become visible, even if they are inside the clipping sphere. In (d), the user deposits the current clipping sphere and starts to manipulate a new one

the clipping sphere can be fixed in place, after which it continues to remove clippable voxels. Additional spheres can then be placed, each with its own segment mask (Figure 5.4d).

Segment selection uses a laser pointer (ray-casting) technique. The pointer highlights the first visible segment it intersects, turning it green (clippable) or red (unclippable) according to the current mask setting. The user toggles a segment's state via the trigger button, and a secondary button resets all mask values within the active sphere. Initially, all segments are clippable (Figure 5.2).

The sphere-manipulation hand controls the position and size of the clipping sphere using the joystick. Pressing the secondary button fixes the sphere in place, storing its mask. A new sphere is then instantiated with a copy of that mask, which can be modified independently. As illustrated in Figure 5.3, the first clipping sphere (blue) is fixed, while a second sphere (green) is actively manipulated.

Users can remove the most recently placed sphere using the primary button. Standard VR navigation controls allow volume translation and rotation by holding a grip button on either controller, or two-handed translation, rotation, and scaling by holding both grips simultaneously. These interactions enable ergonomic positioning of the volume for efficient and precise clipping.

5.3.3 Implementation

AnatomyCarve was implemented using the Unity Engine (Unity Technologies, United States, San Francisco), and uses a customized volume rendering algorithm. To clip parts of the volume, a calculation is performed for each voxel on the GPU using a compute shader to determine if that voxel should be hidden.

The algorithm requires the following inputs: (1) an intensity volume that holds the original values from the CT or MRI scan, (2) a 3D label map that associates each voxel in the intensity volume with a segment label, (3) an opacity transfer function that maps intensity values to opacity, and (4) a color transfer function that assigns a color to each segment label. For the

color transfer function, we adopted the predefined color mapping from 3D Slicer (Fedorov *et al.*, 2012).

The volume rendering process consists of four steps, each executed in a different compute shader (see Figure 5.5):

1. Shadow test volume clipping (Weiskopf *et al.*, 2003) is used to clip voxels from specific segments. For each voxel, we check if it is inside at least one clipping sphere, based on the clipping mask. If a voxel is inside at least one clipping sphere, it is marked as invisible; otherwise, it remains visible. This process can be formalized with the following equation:

$$Clipped(v) = \bigvee_{i=1}^n Inside(s_i, v) \wedge Mask(s_i, label(v)), \quad (5.1)$$

where v represents the voxel with (x, y, z) coordinates, n is the total number of spheres in the scene, s_i is the sphere at index i , $label(v)$ is the label of voxel v from the label map, $Inside(s_i, v)$ checks if voxel v is inside the sphere s_i , and $Mask(s_i, label(v))$ retrieves the clipping mask value for sphere s_i and segment corresponding to $label(v)$. If the voxel is not clipped, we apply the opacity transfer function to it and store the result in the opacity volume. If the voxel is clipped, an opacity value of 0 is stored instead.

2. A compute shader takes the opacity volume as input and performs three-dimensional anti-aliasing (FXAA) (Lottes, 2009) on it, producing an anti-aliased opacity volume.
3. A Sobel operator is applied to the anti-aliased opacity volume to compute the surface normals, which are stored in a separate volume.
4. The final rendering is performed in the visible frame buffer using direct volume rendering. The anti-aliased opacity volume is used as the per-voxel opacity, and the normals volume is used for Blinn-Phong (Blinn, 1977) shading to enhance the visual quality.

Optionally, Local Ambient Occlusion (LAO) (Hernell *et al.*, 2010) can be computed during clipping to improve depth perception. However, LAO was disabled during evaluations to maintain real-time interaction performance.

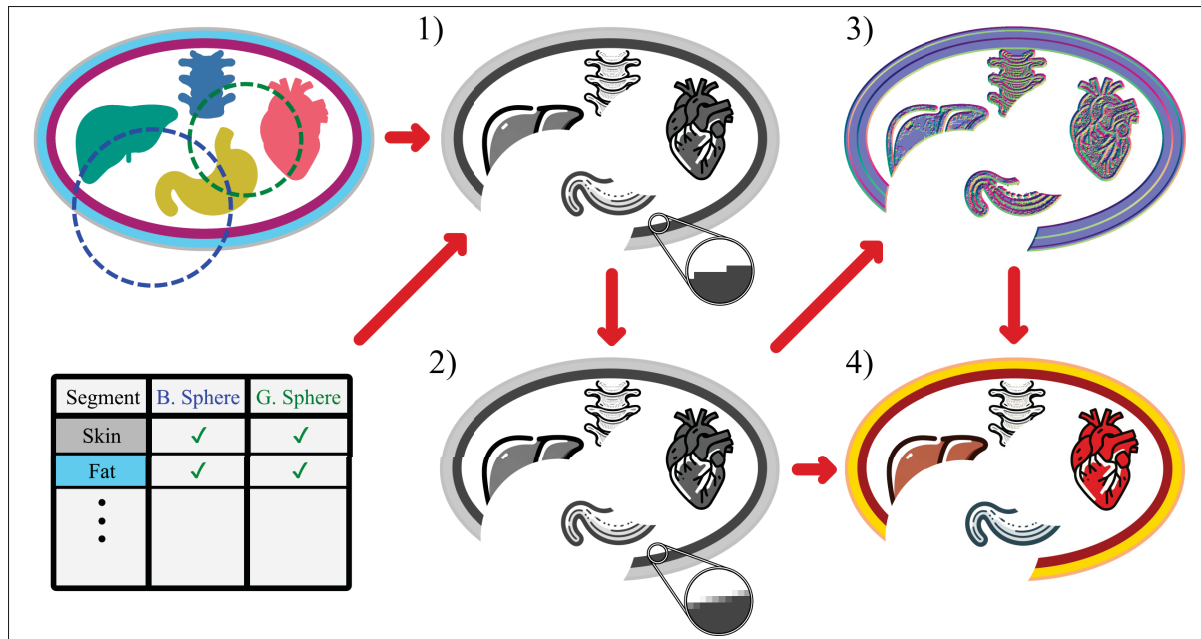


Figure 5.5 Volume rendering pipeline of AnatomyCarve, consists of four stages, each implemented in a separate compute shader. (1) Shadow test volume clipping removes voxels from specified segments according to the clipping mask. (2) Anti-aliasing is applied to the resulting opacity volume. (3) Surface normals are computed from the anti-aliased opacity volume. (4) Final direct volume rendering is performed using the computed opacities and normals

5.4 User Studies

We conducted two complementary user studies to evaluate AnatomyCarve. The first study was a usability evaluation with non-experts, designed to assess how easily new users could create complex visualizations using the tool. The second study involved neurosurgeons and neurosurgical residents, focusing on the system’s usefulness for a practical application—surgical planning. The following sections present the details and findings of each study.

5.4.1 Novice Usability Evaluation

The usability study aimed to assess how easily new users could achieve a desired visualization using AnatomyCarve. To this end, we manually created a set of visualizations inspired by anatomical drawings and asked participants to replicate them using the system. To evaluate the

quality of these replications, a second phase asked participants to rate how closely their results matched the reference illustrations.

The study design was inspired by Díaz *et al.* (2012), who validated their adaptive cross-sections algorithm by having participants copy presented images. We chose this approach because, to our knowledge, no other interactive occlusion management tool for segmented data allows users to create such highly customizable illustrations without being constrained by the focus-and-context visualization paradigm. Consequently, no comparable technique could serve as a fair benchmark without limiting AnatomyCarve’s capabilities. The study was conducted in two phases: the first took place in the laboratory using a VR headset running AnatomyCarve, and the second was completed online after participants finished the initial phase.

5.4.1.1 Data Preparation

For the study, we used medical volumes from the CT-ORG dataset (Rister, Yi, Shivakumar, Nobashi & Rubin, 2020), which were automatically segmented using TotalSegmentator (Wasserthal *et al.*, 2023; Isensee, Jaeger, Kohl, Petersen & Maier-Hein, 2021) in 3D Slicer (Fedorov *et al.*, 2012). From this dataset, we manually created 11 distinct 2D views using AnatomyCarve. Each view depicted multiple anatomical structures, with specific regions strategically removed to reveal hidden components. On average, creating these visualizations involved inserting 3.9 ± 1.2 (mean $\pm$ SD) spheres in the eight non-trial volumes, and toggling segment clippability 25.8 ± 28.8 (mean $\pm$ SD) times. Screenshots of all created views were captured and later presented to participants as ground truth visualizations to replicate.

5.4.1.2 Pre-tests Questionnaire and Tutorial

The user study began with informed consent, followed by a pre-test questionnaire querying participants about their experience with VR and anatomical medical images. Next, participants did a tutorial, which included a short presentation explaining what AnatomyCarve is and how it

can be used. Afterward, participants put on the VR headset, and we guided them through the system's various interactions, step by step, to help them learn how to use it effectively.

5.4.1.3 Task 1: Illustrative Recreation

After completing the tutorial and familiarizing themselves with the controls, participants began Task 1. In each trial, they were shown a 2D projection of a 3D anatomical illustration that had been manually created using AnatomyCarve. Participants were asked to reproduce this illustration by removing voxels from the appropriate segments in the 3D volume. They had to adjust the volume so that, when viewed from the same perspective, it closely matched the 2D reference image. Once satisfied, they pressed a button in the scene to proceed to the next trial.

The study included three guided practice trials followed by eight recorded trials of increasing difficulty, i.e., requiring more spheres and/or more segments to invert. During practice trials, participants received assistance to ensure they understood the system. For recorded trials, they worked independently, although they could ask questions if needed. Figure 5.6 shows the first and last (8th) volumes participants were asked to replicate in the recorded portion of the study.

5.4.1.4 Task 2: Illustration Ranking

In Task 2, our goal was to subjectively evaluate the quality of the visualizations created with AnatomyCarve in VR. After completing Task 1, participants were sent an online questionnaire. In this questionnaire, each participant was shown the 8 original reference illustrations they had been asked to replicate in Task 1, along with 7 anonymized recreations from other participants for each illustration. Their task was to rank the 7 recreations from most to least similar to the corresponding reference illustration.

Each participant evaluated only a subset of the total recreations. Specifically, for each of the 8 trials, participants were assigned 7 out of the 17 total recreations, selected in a pseudo-random but balanced manner so that each recreation was evaluated exactly 7 times across all participants.

All views were presented anonymously to protect participant privacy and avoid bias in the rankings.

5.4.1.5 Evaluation Measures

To evaluate the usability of the system, we used the following metrics and questionnaires for Task 1:

- Time per trial.
- Ratio of inserted clipping spheres by the participants compared to the number used when creating the reference illustrations.
- Ratio of toggled segments by the participants compared to the number toggled when creating the reference illustrations.
- System Usability Scale (SUS) (Brooke, 1996).
- NASA Task Load Index (TLX) (Hart & Staveland, 1988).

For Task 2, we measured:

- The root mean squared error (RMSE) of depth for each recreated view, quantifying the difference between the depth map of the participant-created view and that of the reference illustration.
- The mean absolute error (MAE) of the first encountered segment for each view, which evaluates, for each corresponding pixel, whether the first visible segment encountered during ray casting matches the reference illustration. This is a new metric introduced as an alternative to RMSE for comparing segmented medical images.
- The subjective global ranking of each view, rated by participants and computed using the Plackett-Luce model (Plackett, 1975).

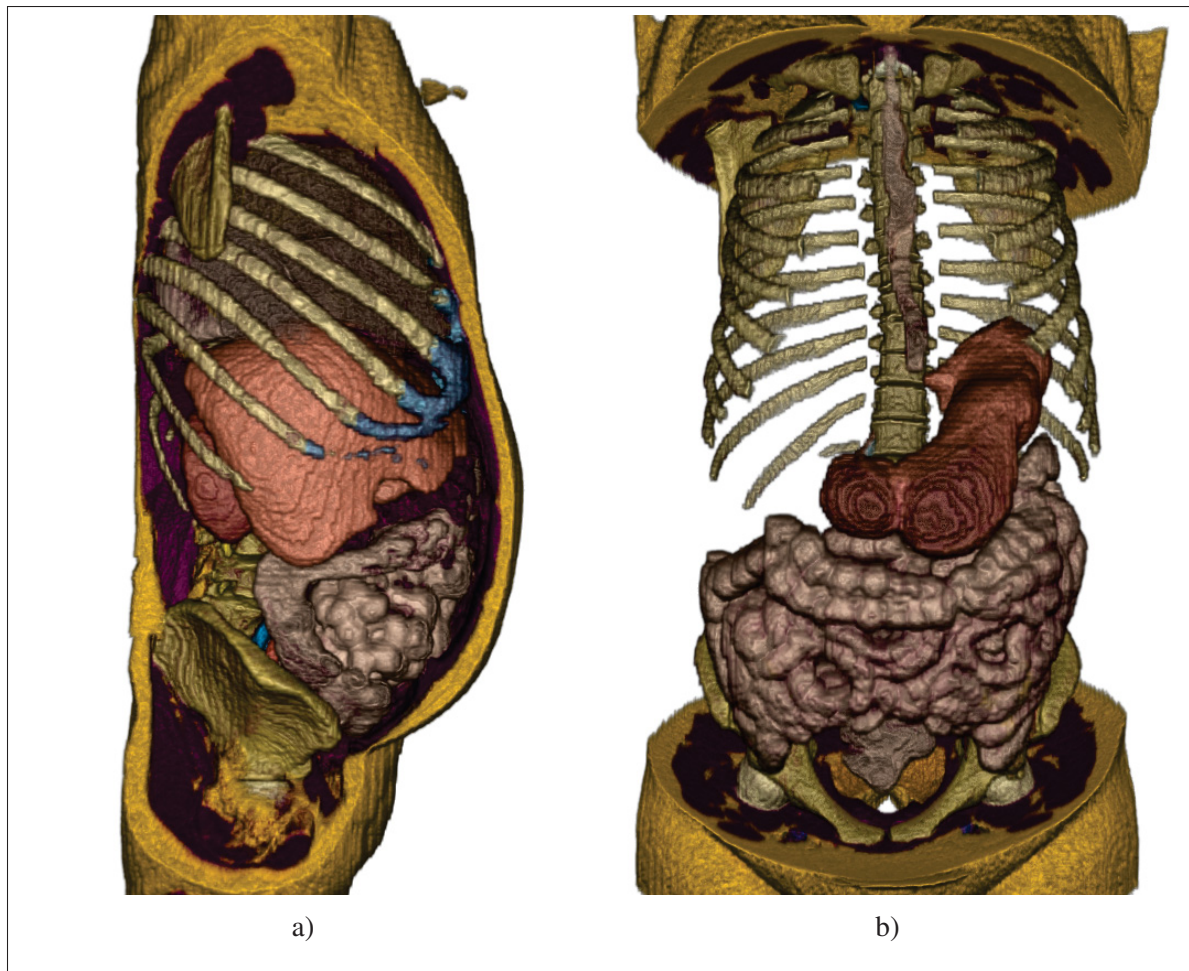


Figure 5.6 The first (a) and the last (b) volumes that participants had to copy during the usability user study

5.4.2 Expert Evaluation

The expert user study focused on surgical planning for the removal of spinal tumors. Six patient volumes were selected from the Centre hospitalier de l'Université de Montréal (CHUM): three with tumors in the thoracic spine, two in the cervical spine, and one in the intramedullary spine.

5.4.2.1 Data Preparation

Anatomical segmentation of bones and soft tissues was performed using TotalSegmentator (Wasserthal *et al.*, 2023; Isensee *et al.*, 2021), followed by manual refinement and validation by



Figure 5.7 The volume presented in the 5th trial. (a) The original volume shown to the participant. (b) The reference illustration created by the researchers, which participants were asked to replicate. (c) The participant-generated image ranked in the third sextile (S1, 3rd best out of 17). (d) The participant-generated image ranked in the fifth sextile (S5, 15th best out of 17)

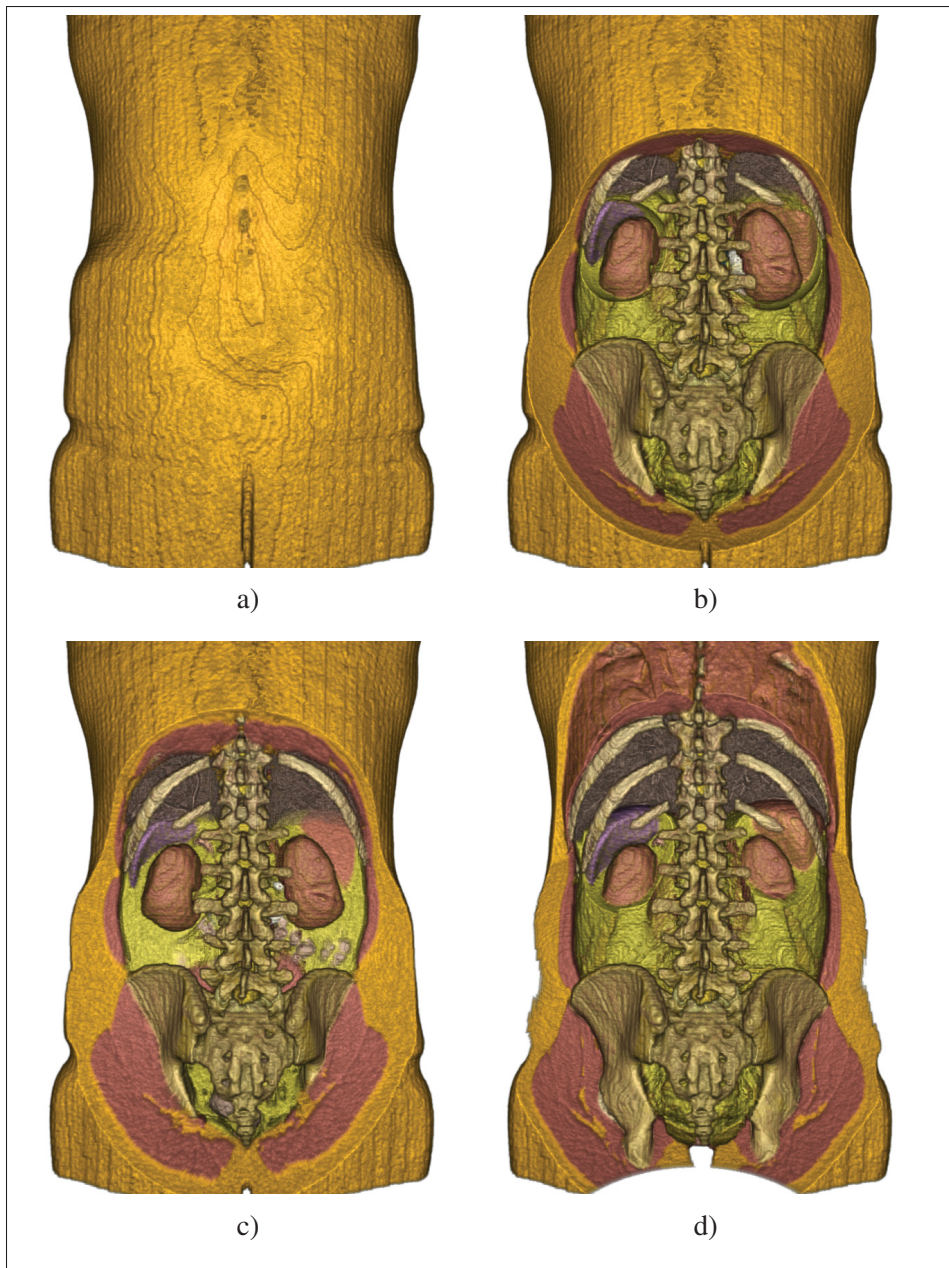


Figure 5.8 The volume presented in the 7th trial. (a) The original volume shown to the participant. (b) The reference illustration created by the researchers, which participants were asked to replicate. (c) The participant-generated image ranked in the third sextile (S1, 3rd best out of 17). (d) The participant-generated image ranked in the fifth sextile (S5, 15th best out of 17)

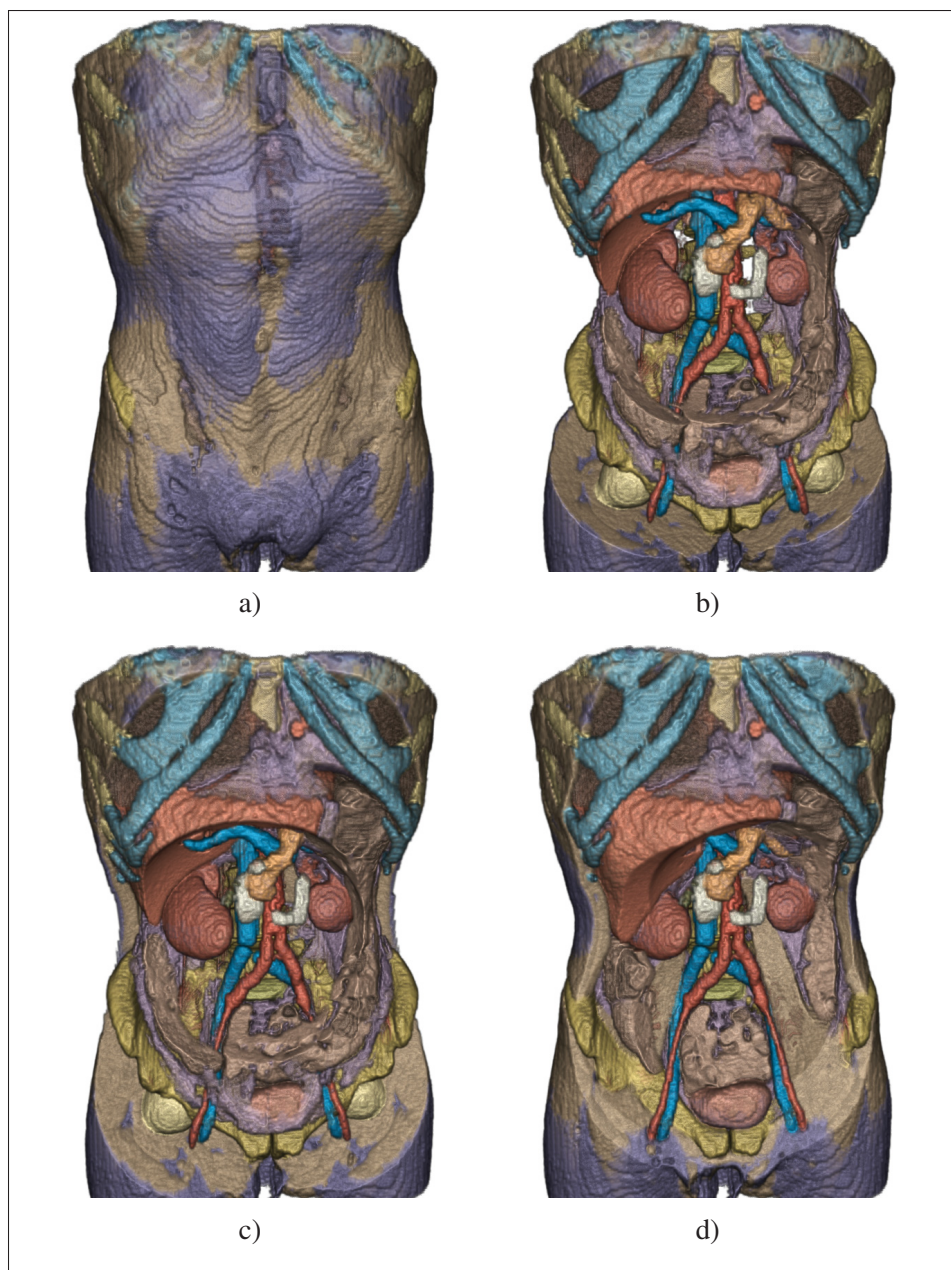


Figure 5.9 The volume presented in the 3rd trial. (a) The original volume shown to the participant. (b) The reference illustration created by the researchers, which participants were asked to replicate. (c) The participant-generated image ranked in the third sextile (S1, 3rd best out of 17). (d) The participant-generated image ranked in the fifth sextile (S5, 15th best out of 17)

a medical expert. Vascular structures were segmented semi-automatically using the “GrowCut” tool in 3D Slicer (Kikinis & Pieper, 2011), again by the same expert.

5.4.2.2 Pre-test Questionnaire and Tutorial

Prior to the study, participants provided informed consent and completed a pre-test questionnaire assessing their familiarity with VR and clinical experience. They were then given a tutorial on how to use AnatomyCarve.

5.4.2.3 Task

The primary objective of the expert study was to simulate traditional surgical planning. Participants used AnatomyCarve to plan trajectories for tumor resection, which involved: (1) identifying the tumor location within the segmented spine volume, (2) determining a safe and effective trajectory to access the tumor, and (3) identifying anatomical structures that would need to be traversed or avoided during the procedure.

5.4.2.4 Evaluation Measures

To assess usability and perceived effectiveness, participants completed the SUS questionnaire (Brooke, 1996) at the end of the session. They were also invited to provide qualitative feedback on the system’s strengths and weaknesses, limitations encountered during use, and suggestions for future improvements.

5.4.3 Apparatus

The studies were conducted on an Oculus Quest 2 headset (Meta Platforms, Inc., United States, Menlo Park). To ensure smooth rendering of the volumetric data, the Oculus Quest 2 was set to a refresh rate of 72 Hz. For the novice study we used a Windows 11 machine with an Intel Core i9-11900H processor, 32 GB of RAM, and an external AMD Radeon RX 6800 XT graphics card with 16 GB of VRAM. For the expert user study, we used a Windows 10 machine with an

AMD Ryzen 7 3700X processor, 32 GB of RAM, and an NVidia RTX 2080 Ti graphics card with 11 GB of VRAM. Ethical approval for the user studies was obtained from the École de Technologie Supérieure ethics committee.

5.5 Results

5.5.1 Novice Usability Evaluation

5.5.1.1 Participants

A total of 17 non-expert participants took part in the study (8 identifying as male, 8 as female, and 1 as non-binary), with ages ranging from 23 to 36 years ($M = 27.4$, $SD = 3.4$). Eighty-two percent reported at least moderate VR experience, with 41% describing themselves as extremely experienced. Regarding anatomical medical images, 41% of participants indicated moderate familiarity, while 53% reported being only slightly familiar or not familiar at all.

5.5.1.2 Task 1: Results

Participants were quick to recreate the anatomical visualizations, taking on average 176 ± 58 seconds (mean $\pm$ SD) per view. They inserted more clipping meshes than in the reference illustrations (1.88 ± 0.55), suggesting exploration or use of smaller, incremental edits, while segment toggling closely matched the original (0.98 ± 0.62), indicating appropriate structure selection.

Novices rated the system highly on the SUS, scoring 77.6, above the average benchmark of 68 (Bangor, Kortum & Miller, 2008), reflecting ease of use and quick learnability. Participants were confident in their ability to learn and use the system quickly, and did not feel the need for significant technical support (see Table 5.1).

The NASA-TLX workload score averaged 32.7, indicating a generally manageable task load. Mental demand and effort were moderately high (53 ± 21 and 52 ± 25 , respectively), reflecting

Table 5.1 System Usability Scale (SUS) results comparing Novice and Expert groups across their respective user studies. Responses are on a 5-point Likert scale (1 = Strongly Disagree to 5 = Strongly Agree). Odd items are positive (higher scores are optimal), and even items are negative (lower scores are optimal)

Question	Novices	Experts
I think that I would like to use this system frequently.	4.06 $\pm$ 0.73	4.13 $\pm$ 0.78
I found the system unnecessarily complex.	1.76 $\pm$ 0.64	1.63 $\pm$ 0.48
I thought the system was easy to use.	4.24 $\pm$ 0.73	4.38 $\pm$ 0.86
I think that I would need the support of a technical person to be able to use this system.	2.53 $\pm$ 1.14	3.13 $\pm$ 0.78
I found the various functions in this system were well integrated.	4.35 $\pm$ 0.76	4.25 $\pm$ 0.66
I thought there was too much inconsistency in this system.	1.12 $\pm$ 0.32	1.75 $\pm$ 0.66
I would imagine that most people would learn to use this system very quickly.	3.82 $\pm$ 0.92	4.13 $\pm$ 0.78
I found the system very cumbersome to use.	1.71 $\pm$ 0.82	2.00 $\pm$ 0.71
I felt very confident using the system.	3.65 $\pm$ 0.97	4.13 $\pm$ 0.93
I needed to learn a lot of things before I could get going with this system.	1.94 $\pm$ 0.87	2.38 $\pm$ 1.11

the cognitive engagement required to replicate the reference illustrations accurately. In contrast, physical demand (21 ± 17), temporal demand (26 ± 21), and frustration (16 ± 28) were low, suggesting that participants did not feel rushed, physically strained, or stressed while interacting with the VR system. Performance scored 28 ± 18 , indicating participants were relatively confident in completing the task. Figure Figure 5.10 summarizes these scores across all six dimensions for novice participants. Overall, these results suggest that AnatomyCarve is usable, imposing minimal physical or emotional burden, while requiring moderate mental effort.

5.5.1.3 Task 2: Illustration Ranking

In the second part of the study, participants were asked to rank recreated views based on their similarity to the reference illustration. To combine these individual rankings into a single consensus, we applied the Plackett-Luce model (Plackett, 1975), which generated a global

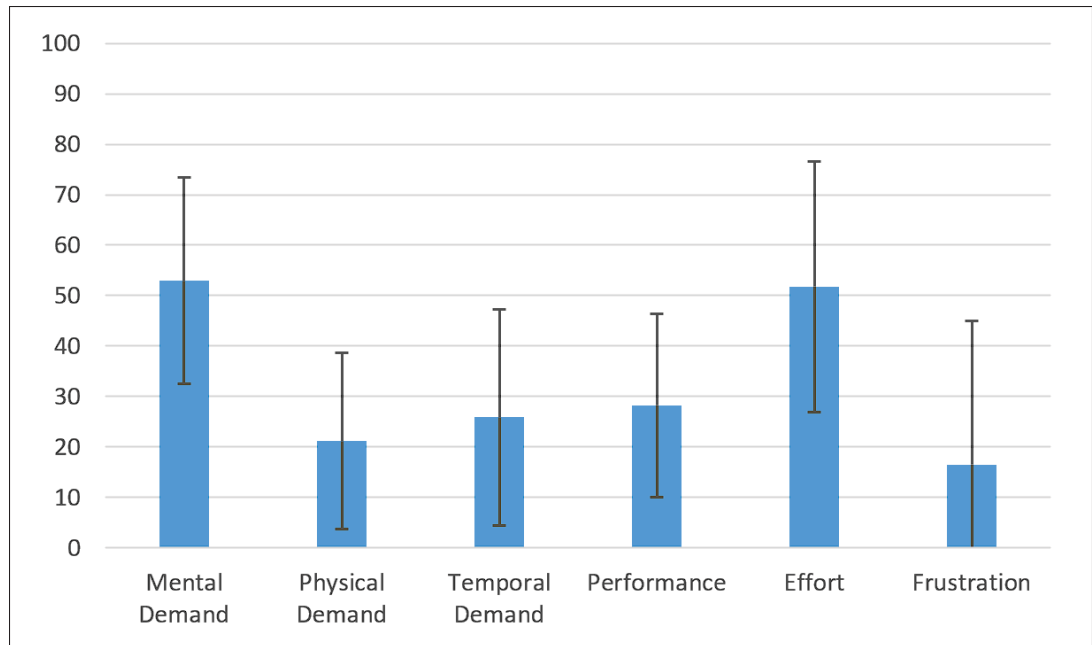


Figure 5.10 NASA TLX score for each of the 6 scales, averaged for all novice participants. The error bars represent the standard deviation

ranking of all participant-created views for each trial. In addition to these subjective rankings, we computed two objective similarity metrics comparing each user-created view to the ground truth: the root mean square error (RMSE) of depth and the mean absolute error (MAE) of the first encountered segment.

Figure 5.11 illustrates the relationship between participant rankings and these metrics, averaged over all eight trials. The trend line for the average MAE of the first encountered segment showed a strong linear relationship with rank ($R^2 = 0.93$, slope = 0.059), whereas the RMSE trend line exhibited a weaker correlation ($R^2 = 0.70$, slope = 0.034), suggesting that MAE better captures the agreement between participant perception and ground truth.

To further characterize performance, we computed the average MAE and RMSE values across all participants for each trial. Trial 5 exhibited the lowest average MAE of the first encountered segment (0.066 ± 0.040 ; Figure 5.7), while trial 7 had the highest error (0.167 ± 0.058 ; Figure 5.8). Interestingly, trial 3 (Figure 5.9) demonstrated the steepest slope and highest R^2 values when

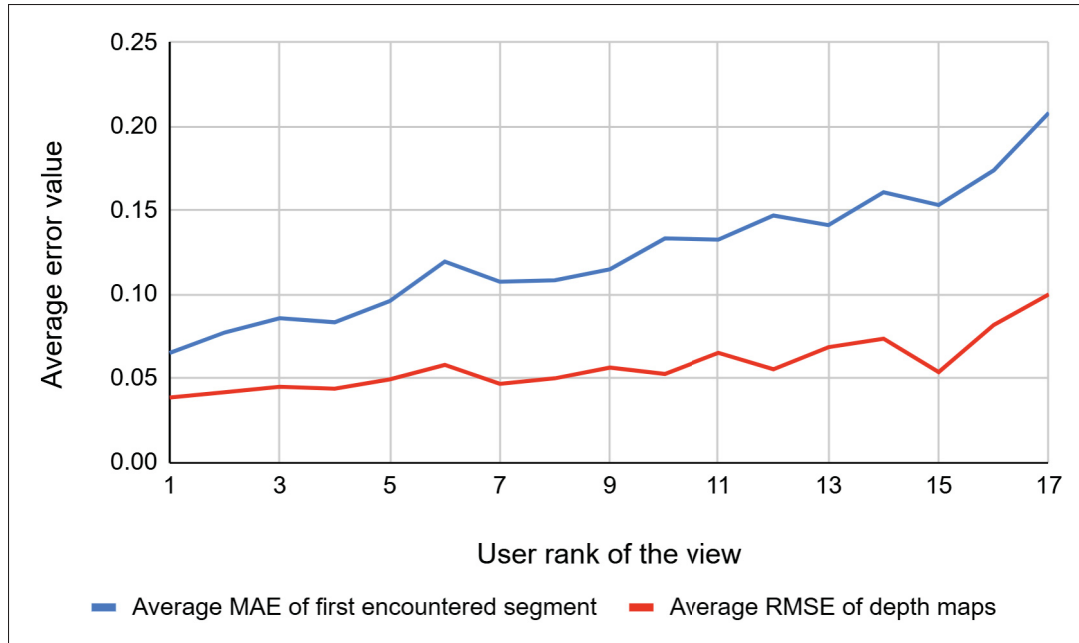


Figure 5.11 Average MAE of the first encountered segment for each pixel, plotted by rank and averaged across all trials

mapping user rankings to both MAE and RMSE ($R^2 = 0.84$ for MAE, $R^2 = 0.625$ for RMSE). This pattern indicates a pronounced disparity in participant performance, highlighting that this task was particularly polarizing in terms of difficulty.

5.5.2 Expert Evaluation

5.5.2.1 Participants

The usefulness of AnatomyCarve for surgical planning was evaluated by 8 practicing neurosurgeons or neurosurgical residents (6 identifying as male, 2 as female) from CHUM. Participants were between 26 and 46 years old ($M = 34.9$, $SD = 6.9$) and had an average of 7.4 ± 5.6 (mean $\pm$ SD) years of clinical experience. Regarding VR usage, 3 participants reported using VR at least once per month, while 5 reported never having used VR.

5.5.2.2 Results

Expert participants' evaluation of AnatomyCarve combined both subjective usability ratings and qualitative feedback. In terms of SUS scores, experts rated the system slightly lower (75.3) than novices (77.6), yet they generally reported higher confidence in using the system, finding it easier to learn and operate, even though they perceived it as slightly more inconsistent and requiring more technical support than novices (see Table 5.1).

Across all participants, the system's three-dimensional visualization of anatomical structures was widely appreciated, with 7 out of 8 participants noting it as a significant improvement over traditional surgical planning methods, which typically rely on 2D slices (axial, coronal, sagittal) or basic 3D volumetric renderings. Specifically, four participants emphasized that the 3D views were especially valuable for preoperative evaluation and planning, while another four highlighted that the visualization helped clarify the spatial arrangement and connections between anatomical features, enhancing their understanding of anatomical relationships during surgery.

Despite these strengths, several limitations were reported. Five participants found that the resolution and precision of the visualized volumes were insufficient, making it challenging to identify finer anatomical structures. Three participants noted that the lack of integrated 2D imaging made it harder to relate traditional imaging modalities to the 3D views, and another three commented on a learning curve, indicating that users may require time to become proficient with the system.

Participants also offered suggestions for improvement. Five recommended increasing the system's resolution to better visualize smaller anatomical details, while four suggested incorporating 2D image viewing alongside the 3D visualization to improve usability. Additionally, one participant proposed adding an eraser-like tool to continuously clip elements from different segments, which could enhance flexibility during surgical planning.

Overall, these results suggest that AnatomyCarve provides meaningful improvements over traditional methods, particularly in conveying 3D anatomical relationships, while highlighting areas for refinement in resolution, 2D integration, and workflow flexibility.

5.6 Discussion

The evaluation of AnatomyCarve demonstrates strong potential for both novice and expert users, indicating that the system is effective for creating anatomical visualizations and supporting surgical planning, though several areas could be improved.

In terms of usability, non-expert and expert participants reported similar SUS scores (77.6 and 75.3, respectively), suggesting that the interface is generally intuitive and easy to use. The NASA-TLX score of 32.7 for non-experts indicates that, despite overall usability, a modest learning curve exists. This observation aligns with expert feedback and likely reflects both the differences between AnatomyCarve and traditional visualization methods, and the inherent complexity of the interface, which requires users to remember multiple button actions and gestures.

Task 2 of the novice study provides additional insight into system effectiveness. Participants ranked recreated views based on similarity to the ground truth, and objective metrics, particularly the MAE at the first encountered segment, correlated strongly with participant rankings. The MAE trend line demonstrated higher R^2 and slope values than RMSE, suggesting it is a reliable measure of visualization fidelity. For example, the volume with the lowest MAE (Figure 5.7) closely matched the ground truth, accurately representing key anatomical structures. Even in the volume with the highest MAE (Figure 5.8), major features such as the kidneys and pelvis were preserved, though some details (e.g., skin and exposed muscles) were absent. These findings indicate that novices can effectively generate accurate 3D visualizations using AnatomyCarve, and that MAE at the first encountered segment serves as a valuable objective complement to subjective rankings and usability assessments.

Experts, including neurosurgeons and neurosurgical residents, reported high satisfaction with the system's 3D visualization capabilities, particularly its ability to convey spatial relationships between anatomical structures, a challenge when using traditional 2D imaging modalities. However, experts identified limitations related to the resolution and precision of visualized volumes, which affected identification of smaller structures such as nerves. Importantly, these limitations stem from the underlying dataset rather than the system itself. Enhancing dataset resolution and employing more precise segmentation techniques, potentially via automatic or semi-automatic methods, could mitigate this issue.

The learning curve was another area highlighted by experts. Several participants recommended integrating 2D imaging alongside the 3D views to facilitate interpretation, particularly for users accustomed to traditional planning tools. This integration would allow cross-referencing between conventional and novel representations, reducing the learning curve and increasing overall system intuitiveness in clinical settings.

Overall, AnatomyCarve demonstrates substantial promise as both an illustrative tool for anatomical exploration and education, as well as a preoperative planning tool, providing rapid and high-quality 3D anatomical visualizations. Its utility could be further enhanced by refining resolution, improving segmentation, and incorporating complementary 2D imaging. Additional developments, such as implementing segment deformation capabilities to simulate occlusion management techniques commonly used in anatomical textbooks, introducing a continuous "eraser"-like tool for inserting clipping shapes while holding a button, and adopting advanced rendering techniques like Contextual Ambient Occlusion (Titov *et al.*, 2024) to enable real-time interactive clipping, could further improve usability, flexibility, and visual fidelity, making the system an even more powerful resource for surgical planning, particularly in neurosurgery.

5.7 Conclusion

In this paper, we introduced AnatomyCarve, a novel interactive occlusion management tool that enhances the visualization of segmented medical images in VR through a stepwise dissection

paradigm. By allowing users to selectively clip and manipulate segments, the system enables the creation of complex 3D visualizations while preserving the spatial relationships between anatomical structures.

Our evaluation demonstrates that AnatomyCarve is both intuitive and effective: novices were able to quickly generate high-quality custom anatomical images, while experts highlighted its utility for surgical planning by providing clear spatial understanding of complex anatomy.

AnatomyCarve represents a significant advancement in medical visualization, transforming traditional 2D anatomical illustrations into immersive, interactive 3D experiences. By supporting highly customizable exploration of anatomical structures, it enhances both preoperative planning and medical education, allowing users to gain deeper insight into complex anatomical relationships that static images or conventional clipping techniques cannot provide. This work establishes a new paradigm for interactive 3D medical visualization, bridging the gap between educational tools and clinical applications.

Code Availability

The full source code for AnatomyCarve is available as a Unity3D package: <https://github.com/andrey-titov/AnatomyCarve>. A full desktop re-implementation as a 3D Slicer module is available at: <https://github.com/andrey-titov/SlicerAnatomyCarve>.

Acknowledgments

This work was funded by the Natural Sciences and Engineering Research Council of Canada, Discovery program (RGPIN-2020-05084), as well as by the Fonds de recherche du Québec – Nature et technologies (B2X Doctoral Scholarship), Application 334501 (<https://doi.org/10.69777/334501>).

CONCLUSION AND RECOMMENDATIONS

The purpose of our work is to investigate how modern VR headsets could enhance the visualization of medical images. We examine the roles of depth perception, rendering quality, and interactivity to assess how these aspects impact the interpretation of angiographic data, as well as segmented and unsegmented anatomical datasets that exhibit significant self-occlusion. The results of our research indicate that VR is a promising technology for medical image visualization, and differs significantly from standard desktop computers.

The first objective of this research was to determine how depth cues developed for a standard desktop setup translate to virtual reality. For this objective, we conducted a user study and found that adding manual interactivity to any shading cues significantly reduced participants' head movement, which may indicate that the users relied less on the motion parallax depth cue inherently present when using a VR HMD. Regarding the depth cue itself (Blinn-Phong shading, chromadepth or aerial perspective) performance was on average unchanged. Specifically, it did not affect comparative decision time, global depth perception (judging the depth between two vessels that are far apart), or local depth perception (judging depth around a given vessel). This contrasts with 2D monitor visualization, where these cues yield substantial differences in depth disambiguation (Kersten-Oertel *et al.*, 2014; Drouin *et al.*, 2020). Further, in terms of local depth perception, we have found that the Blinn-Phong shading (Blinn, 1977) cue performs better than both fog and chromadepth.

This work provided design guidance for VR medical visualization. When global depth ordering is the priority, any static cue (Blinn-Phong shading, chromadepth or aerial perspective) is suitable, as existing VR depth-enhancement methods leave little room for further improvement. However, when local manipulation and fine targeting matter, more realistic Blinn-Phong shading is preferable, as we hypothesize that its combination with head movements provides better local depth perception. Additionally, the result that dynamic cues yielded significantly less

head movement than static cues suggests that in settings where ergonomics or fixed viewpoints restrict head motion, for example, a surgical microscope, a seated view, or prolonged procedures, they reduce head path. This may in turn lessen fatigue and improve comfort. Participants also preferred dynamic cues, which may suggest that perceived comfort and effort warrant consideration when selecting cues. Further, this work also enabled us to establish a hierarchy of cues in VR. First, binocular disparity and motion parallax set the baseline, then the shading cue refines local perception. Finally, chromadepth and fog can be optionally added, but they do not contribute to depth perception in any measurable manner.

The second objective of the thesis was to determine how VR could improve the visualization of unsegmented volumes by combining the interactivity provided by modern VR headsets with specialized custom rendering. Specifically, we aimed to improve occlusion management in VR through interactive clipping and to enhance depth perception with global-illumination cues. For this objective, we developed a novel interactive visualization technique called Contextual Ambient Occlusion, which is based upon the Local Ambient Occlusion algorithm (Hernell *et al.*, 2010), retaining its quality, while offering significantly better performance when clipping the volume. This performance increase makes it possible to be used with real-time clipping changes in the volume, while still providing global illumination in the form of volumetric ambient occlusion. The algorithm supports the insertion of multiple clipping shapes, each of which are taken into account during rendering. In this rendering technique, the interactivity provided by VR makes it easier to insert 3D shapes in the volume at the desired position and orientation, while the incorporation of global illumination reveals more information about the dataset's structure than is achievable using only local illumination.

This work on real-time ambient occlusion unlocks true interactive clipping with global cues, which hasn't been done previously. AO is often disabled, simplified or amortized in real-time interaction due to its computational overhead, compromising depth perception precisely when

spatial understanding is most needed (Lindemann & Ropinski, 2011; Kroes, Post & Botha, 2012). Our system delivers stable high-quality AO every frame while a user manipulates clipping shapes, eliminating temporal blur and letting the user trust what they see. This reliability is what makes CAO stand out from other techniques that rely on screen-space visualization or gradual calculation of the image, like in the case of the original LAO algorithm (Hernell *et al.*, 2010). Our approach offers scalable performance gains that increase with dataset size and sampling complexity. The method maintains interactive frame rates even with multiple clipping shapes, indicating that multi-shape clipping is a practical extension. Moreover, because computation occurs in data space, the technique integrates naturally into multi-view environments such as stereo and collaborative VR, avoiding stereo inconsistencies and supporting shared exploration of volumetric data.

The third objective was to understand how VR could assist occlusion management in the visualization of a segmented volume by focusing on interactivity combined with clipping. To complete this objective, we drew inspiration from medical illustrations featured in human anatomy books, which often employ organ-specific clipping to reveal inner anatomy obstructed by outer layers. Building on this idea, we created an interactive visualization technique called AnatomyCarve that allows users to clip desired segments in the segmented image. The user could insert a clipping shape into the volume that would only clip the segments previously selected for clipping. It featured a “stepwise dissection” concept, allowing the insertion of multiple clipping shapes within the medical image, each clipping different segments. This technique enables users to easily create custom visualizations, as confirmed by a user study conducted with non-experts. Moreover, the tool was tested by neurosurgeons and neurosurgical residents for its usefulness in the clinical domain. The system was found to be simple and efficient for creating custom medical views, and effective for medical planning. Thus, due to its occlusion management capabilities, it may be advantageous compared with visualizations traditionally used for preoperative planning.

AnatomyCarve applies a focus-and-context visualization paradigm to manage occlusion by preserving surrounding structures that convey spatial relationships while selectively removing material that hides the target tissues or organs. This principle is useful for medical imaging, and more specifically, anatomical education, where students could exercise their knowledge by reconstructing anatomical illustrations to better learn the location of organs and tissues. Further, it is useful for communication, as it gives the possibility to create reproducible views. In other words, AnatomyCarve makes it possible to save the "recipe" of the view, and to alter it slightly, meaning that the views are easy to replicate. Furthermore, findings from the user study indicate that AnatomyCarve is useful for clinical planning, as it avoids deformation-based volumetric browsing methods (such as the cut-open or peel operations described by McGuffin *et al.* (2003). Such volumetric probes, as classified by Elmqvist & Tsigas (2008), can distort geometry and spatial location, disrupting reliable trajectories and adjacency. By avoiding these operations, AnatomyCarve preserves these relationships when spatial precision is essential.

Finally, in evaluating AnatomyCarve, we introduced two metrics for comparing medical visualizations: the Mean Absolute Error of the first encountered segment, and a subjective global ranking derived from participants' judgments. The MAE of the first encountered segment is a transferable, semantic measure for two-dimensional images of segmented volumes that, for each corresponding pixel, compares the first visible segment in one image to that in the other and then computes the mean absolute error over the image. This allows similarity assessment that accounts for semantic content within the volume, which is useful for education and automated view generation, and is more informative than purely geometric comparisons. The second metric is a subjective global ranking based on participants' judgments that orders visualizations by similarity to a reference ground truth view. This ranking can also serve as an indicator of perceived usability, since excluding a small number of outlier views reveals the range of visualizations produced by participants. It also offers a comparable, unit-agnostic summary aligned with human perception and enables uncertainty analysis based on participant agreement.

6.1 Impact and Contributions

VR is a promising technology that can efficiently support the visualization of medical images. Regarding depth perception, because VR headsets inherently provide cues such as stereoscopy and motion parallax, the shading cues designed for 2D screens have less impact in VR than interactivity, which positively influences performance. Additionally, we have determined that clipping is a potent data visualization strategy for visualizing datasets that exhibit significant self-occlusion, both segmented and unsegmented. This is because clipping requires positioning of shapes within the 3D scenes with a certain precise rotation, which is easier to achieve with VR controllers with 6 degrees of freedom (DoF) than with a traditional computer mouse.

The main impact of this work is a deeper understanding of how virtual reality can enhance medical image visualization. From the angiography study, we learned that VR's intrinsic depth cues, binocular disparity and motion parallax, already provide strong depth information. As a result, cue choice affects decision time and perceived depth far less than on 2D displays. Nevertheless, shading continues to improve local spatial accuracy, and dynamic, controller-based cues reduce head movement, indicating that immersive setups can make angiographic exploration both more precise and more ergonomic in clinical contexts. From the Contextual Ambient Occlusion project, we learned that tailoring shading for interactive VR is practical by recomputing ambient occlusion only near the clipping boundary. This approach preserves LAO-level image quality, eliminates stereo inconsistencies, and achieves a two- to fivefold speedup, removing a key bottleneck for real-time VR clipping and enabling truly interactive volume exploration. From AnatomyCarve, we found that segment-aware clipping designed around a 6 DoF controller input preserves spatial context while revealing target structures. The technique was well received by both non-experts and neurosurgeons, demonstrating that VR-specific interaction can generate illustration-like, task-oriented views that support education and surgical planning beyond what standard desktop tools can offer.

Additionally, our work made exploration of 3D medical images more accessible and deepened our understanding of effective visualization in virtual reality. Across the three systems we developed, volume exploration became more direct and hands-on, reducing reliance on menus, switches, and complex interfaces, and thereby lowering users' cognitive load. Our studies identified clipping as a strong fit for VR, where handheld 6 DoF controllers allow for the precise placement of 3D clipping shapes, and this approach was well-received in AnatomyCarve evaluations. Our studies also introduced two practical metrics to compare segmented 2D renderings: an accuracy measure for the first encountered segment and a global ranking from participant judgments, which support consistent view ranking and usability analysis. Finally, we advanced the concept of a shareable view state in the AnatomyCarve system. Defined by the inserted clipping shapes, this state allows collaborators to reopen and explore the same spatial view rather than relying on approximate screenshots.

6.2 Future Work

The research on the usefulness of VR HMDs for medical image perception is far from complete, and additional studies are warranted. In particular, the proposed Contextual Ambient Occlusion algorithm should be examined more extensively for the visualization of segmented data, and it could be incorporated into AnatomyCarve to further improve rendering quality. We hypothesize that improved shading fidelity, including high-quality ambient occlusion, will better support the visualization of segmented medical images by enhancing depth perception within segmented volumes, which in turn may strengthen surgeons' preoperative planning.

Another direction is to study how the VR techniques translate to augmented reality for intraoperative guidance. Future work should evaluate whether segment-aware clipping in AnatomyCarve and CAO-based shading remain usable under AR constraints such as registration stability, correct occlusion, and fixed-view microscope modes. This matters in the clinic: for example, AR is already used at the microscope to overlay indocyanine green angiography

and patient-specific vascular models during aneurysm surgery, helping with clip placement and real-time flow assessment (Carl, Bopp, Benescu, Saß & Nimsky, 2020). It would also be valuable to compare common AR display types and to examine how room lighting and instrument motion influence cue visibility. Reports point to registration error, weak depth perception, and occlusion conflicts as ongoing issues (Roethe, Rösler, Misch, Vajkoczy & Picht, 2022), which directly motivate our focus on occlusion-correct shading and robust registration. Recent work on real-time instrument segmentation for AR overlays also suggests practical ways to keep the depth ordering correct when tools occlude the target (Tanzi, Piazzolla, Porpiglia & Vezzetti, 2021). Using the same tasks and perception measures as in VR will show whether these planning techniques generalize to the operating room and how to adapt the cues for AR. Porting the work to AR would create a direct bridge from preoperative planning to surgical guidance, allowing the same interactive actions to support decisions during the procedure without leaving the clinical workflow.

Moreover, AnatomyCarve should be tested on other types of tasks, such as anatomy learning. It could support layered dissection activities in which learners insert clipping shapes to expose selected segments, examine spatial relations, and save views for later study. These activities would record the time needed to expose target segments and the accuracy of anatomical identification to measure the learning curve compared with traditional rendering.

In addition, recent progress on Gaussian-splatting rendering shows support for real-time clipping (Li *et al.*, 2025). Compared with standard volume rendering that traces rays through every voxel, Gaussian splatting draws the scene as many small soft blobs that sum to the full shape (Kerbl, Kopanas, Leimkuehler & Drettakis, 2023). Because only the important blobs need to be drawn and each blob is rendered as a smooth screen-space ellipse, this approach is efficient, supports fast hiding and revealing of structures, and keeps thin vessels looking smooth during motion. Integrating this capability into future versions of AnatomyCarve has the potential to enhance

rendering quality and to facilitate volume interpretation. Future work should assess whether splatting-based rendering preserves stereoscopic consistency, supports segment-aware clipping, and interoperates with CAO in both VR and AR.

Finally, the proposed algorithms should be evaluated in collaborative multi-user settings. These settings should let teams modify and merge surgical plans, add annotations, and compare approaches on the same case. The system should also synchronize views in real-time, keep a history of actions, and offer role-specific interfaces for surgeons, trainees, and observers. Future studies should measure plan quality and time to consensus, and assess perceived workload and trust in the system. They should also test whether remote collaboration offers benefits comparable to in-person work.

APPENDIX I

MANUSCRIPT 2 ADDITIONAL NOTES

1. Additional notes for algorithm 4.2

- Line 3: The algorithm starts by executing the `ShadowTestClipping` function on the 9 samples around the voxel (x, y, z) , where one sample is taken at its center, and 8 samples are taken at the corners of the voxel.
- Line 18: The function `ShadowTestClipping` iterates through the samples that were passed to it and performs shadow test clipping on them. All samples are given in the local space, hence a single sample is referred to as (x_l, y_l, z_l) to differentiate it from the voxel (x, y, z) .
- Line 19: The sample is transformed to the coordinate space of the clipping camera. The resulting (x_c, y_c) coordinates will correspond to a pixel position in the FBO, while the z_c component corresponds to the depth of the sample in the clipping camera coordinate space.
- Lines 20 and 21: The near Z_{front} and far Z_{back} depth maps obtained in the precalculation step II are sampled at the 2D position obtained in Line 19, which gives two depth values z_{front} and z_{back} .
- Lines 22 and 23: The D_{front} and D_{back} dilation maps obtained in the precalculation step III are sampled at the 2D position obtained in Line 19, which gives two depth values d_{front} and d_{back} .
- Lines 24 and 25: The depth of the sample obtained in Line 19 is compared to the depths obtained on Lines 20 and 21. If the depth of the sample is between the depth values of z_{front} and z_{back} , the sample is considered to be invisible, otherwise it is considered as visible. If the voxel is visible, a counter $c_{visible}$ is incremented that keeps track of the number of visible samples.
- Lines 26 and 27: The depth of the sample obtained in Line 19 is compared to the depths obtained in Lines 22 and 23. If the depth of the sample is between the depth values of d_{front} and d_{back} , a counter $c_{dilated}$ is incremented that keeps track of the number of samples inside the dilation shape.

After this, a check determines whether MSAA antialiasing should be used.

- Lines 4 and 5: If the algorithm determines that the 9 samples are all visible, then it means that the voxel (x, y, z) was not affected by clipping, and its value is used directly. In that case, the opacity transfer function TF_o is applied on its value and the result is saved in the V_o volume.
- Lines 6 and 7: If the algorithm determines that the 9 samples are all invisible, then it means that the voxel is fully clipped, so an opacity of 0 is stored in V_o for the current voxel (x, y, z) .
- Lines 8 and 9: If some of the samples are visible and some are not, then N additional samples are taken at predefined positions in the $[-1, -1, -1]$ to $[1, 1, 1]$ neighborhood of the voxel, for a total of $N + 9$ samples. For these samples, the function `ShadowTestClipping` is executed again.
- Line 10: After the calculation in Line 9 is done, the anti-aliased opacity is obtained using the indicated formula and then stored in V_o for the current voxel (x, y, z) .

In addition to calculating the anti-aliased opacity for each voxel, the compute shader also outputs a 3D mask V_m that would indicate to which of the following categories the voxel corresponds: clipped (category a), not clipped, but LAO factor affected by clipping (category b) and not affected by clipping (category c). To determine this, simple checks are executed on the counters.

- Lines 11 and 12: If $c_{dilated}$ is equal to 0, it means that the voxel is completely outside the dilation shape, which means that it corresponds to category c , and a value of 1 is stored in V_m for the current voxel.
- Lines 13 and 14: Otherwise, if $c_{visible}$ is bigger than 0, it means that the voxel is outside the clipping shape but inside the dilation shape, so it corresponds to category b , and a value of 0.5 is stored.
- Lines 15 and 16: Otherwise, the voxel is completely clipped (category a), so a value of 0 is stored.

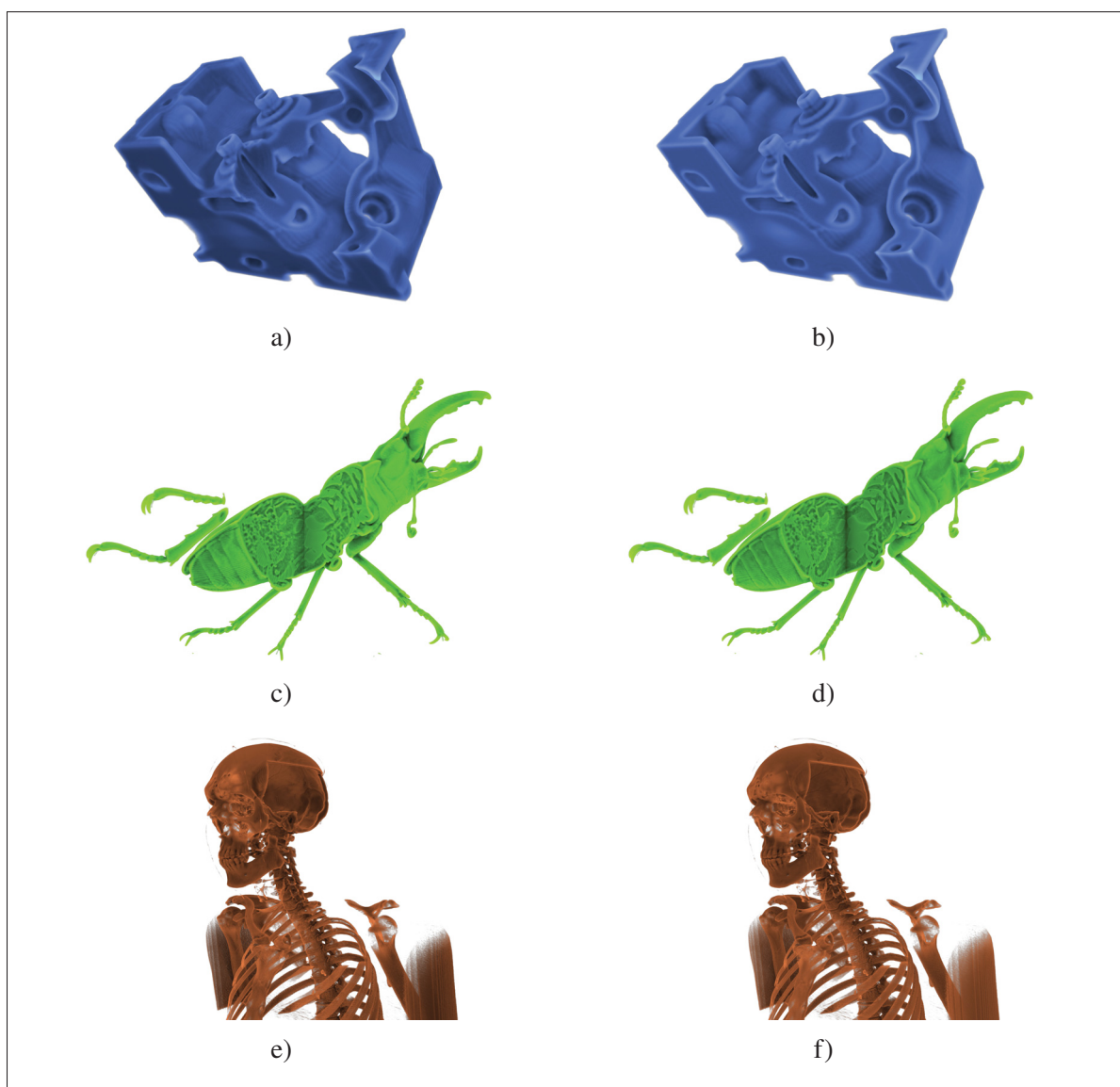


Figure-A I-1 Comparison between the images created using CAO for the 3 scenes using a 6-ray and a 54-ray spherical ray cast. The top, middle and bottom rows demonstrate the Engine, Beetle and Skeleton scenes correspondingly

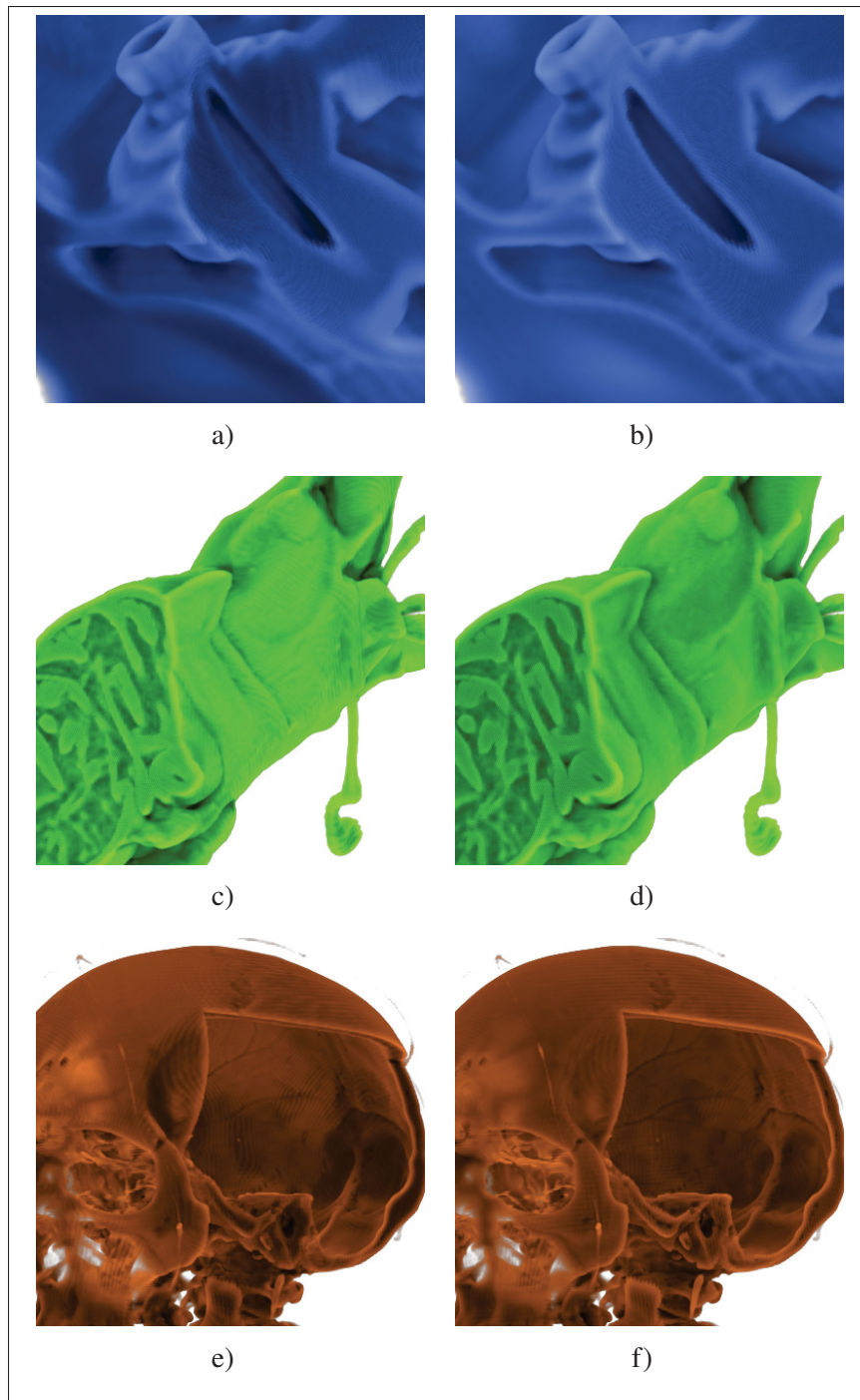


Figure-A I-2 Close-up of the images created using CAO for the 3 scenes using a 6-ray and a 54-ray spherical ray cast. The top, middle and bottom rows demonstrate the Engine, Beetle and Skeleton scenes correspondingly

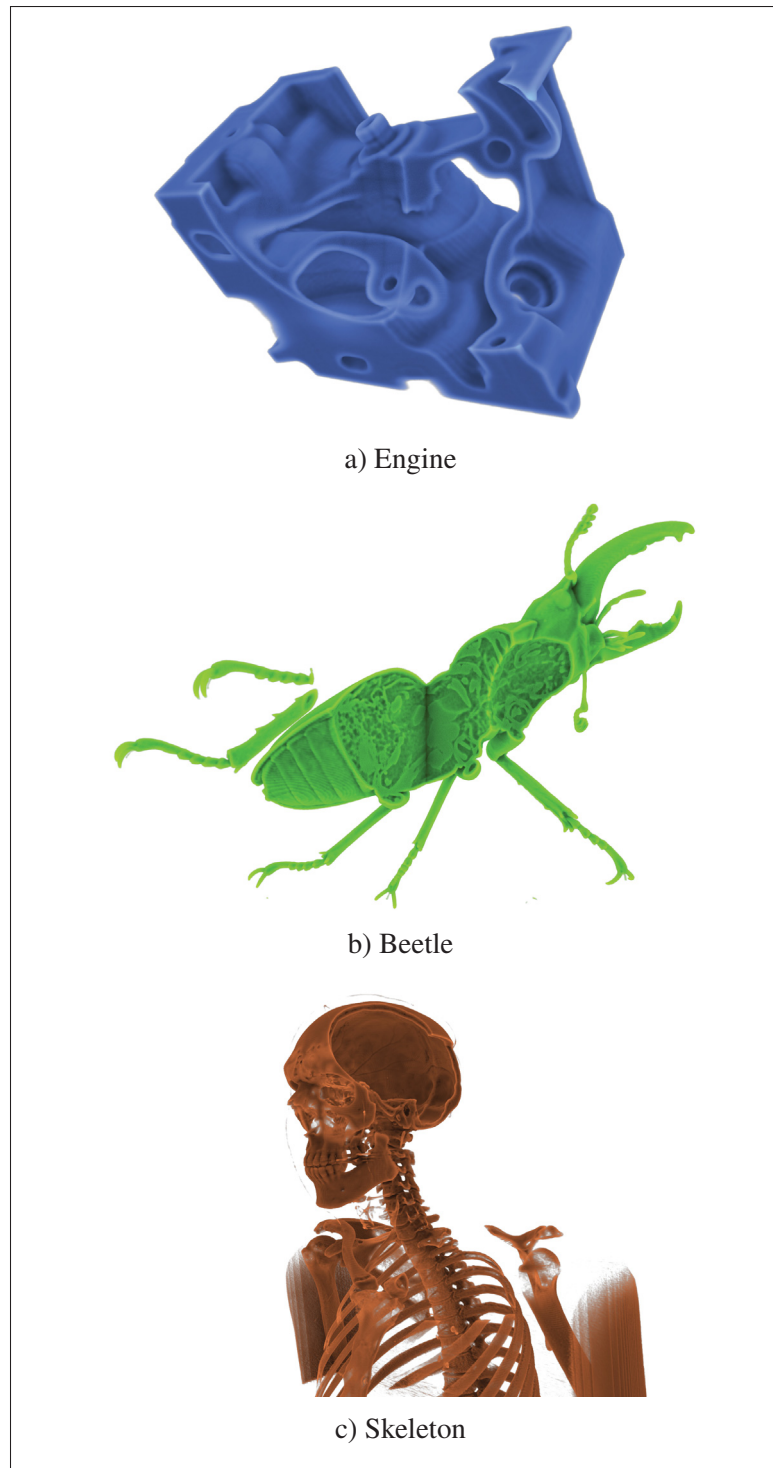


Figure-A I-3 Modified scenes with 2 clipping meshes. A sphere was added to each scene as a second clipping mesh



Figure-A I-4 Per-pixel 2D error between images created with a 6-ray, 14-ray, 26-ray and 54-ray spherical ray cast and the ground truth image created with 512 rays. Each pixel color aggregates the average error between the RGB components

LIST OF REFERENCES

- Abbara, S., Blanke, P., Maroules, C. D., Cheezum, M., Choi, A. D., Han, B. K., Marwan, M., Naoum, C., Norgaard, B. L., Rubinshtein, R., Schoenhagen, P., Villines, T. & Leipsic, J. (2016). SCCT guidelines for the performance and acquisition of coronary computed tomographic angiography: A report of the Society of Cardiovascular Computed Tomography Guidelines Committee: Endorsed by the North American Society for Cardiovascular Imaging (NASCI). *Journal of Cardiovascular Computed Tomography*, 10(6), 435–449. doi: 10.1016/j.jcct.2016.10.002.
- Abbas, J. R., O'Connor, A., Ganapathy, E., Isba, R., Payton, A., McGrath, B., Tolley, N. & Bruce, I. A. (2023). What is Virtual Reality? A healthcare-focused systematic review of definitions. *Health Policy and Technology*, 12(2), 100741. doi: 10.1016/j.hlpt.2023.100741.
- Abhari, K., Baxter, J. S. H., Eagleson, R., Peters, T. & Ribaupierre, S. d. (2012). Perceptual enhancement of arteriovenous malformation in MRI angiography displays. *Medical Imaging 2012: Image Perception, Observer Performance, and Technology Assessment*, 8318, 831809. doi: 10.1117/12.911687.
- Ajani, B., Bharadwaj, A. & Krishnan, K. (2018). Volumetric Clipping Surface: Un-occluded Visualization of Structures Preserving Depth Cues into Surrounding Organs. *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, pp. 291–298. doi: 10.1007/978-3-030-00937-3_34.
- Akenine-Möller, T., Haines, E. & Hoffman, N. (2019). *Real-Time Rendering* (ed. 3). New York: A K Peters/CRC Press. doi: 10.1201/9781315365459.
- Arthur, K. (2000). *Effects of Field of View on Performance with Head-Mounted Displays*. (Ph.D. thesis, University of North Carolina at Chapel Hill, Chapel Hill, North Carolina, United States). Retrieved from: https://www.researchgate.net/publication/2243094_Effects_of_Field_of_View_on_Performance_with_Head-Mounted_Displays.
- Bailey, M. & Clark, D. (1998). Using ChromaDepth to Obtain Inexpensive Single-image Stereovision for Scientific Visualization. *Journal of Graphics Tools*, 3(3), 1–9. doi: 10.1080/10867651.1998.10487491. _eprint: <https://doi.org/10.1080/10867651.1998.10487491>.
- Baliyan, V., Shaqdan, K., Hedgire, S. & Ghoshhajra, B. (2019). Vascular computed tomography angiography technique and indications. *Cardiovascular Diagnosis and Therapy*, 9(Suppl 1), S14–S1S27. doi: 10.21037/cdt.2019.07.04.

- Balsa Rodríguez, M., Gobbetti, E., Iglesias-Guitián, J. A., Makhinya, M., Marton, F., Pajarola, R. & Suter, S. K. (2013). A Survey of Compressed GPU-Based Direct Volume Rendering. Retrieved from: <http://hdl.handle.net/2183/41921>.
- Bangor, A., Kortum, P. T. & Miller, J. T. (2008). An Empirical Evaluation of the System Usability Scale. *International Journal of Human-Computer Interaction*, 24(6), 574–594. doi: 10.1080/10447310802205776. _eprint: <https://doi.org/10.1080/10447310802205776>.
- Barhorst-Cates, E. M., Rand, K. M. & Creem-Regehr, S. H. (2016). The Effects of Restricted Peripheral Field-of-View on Spatial Learning while Navigating. *PLoS ONE*, 11(10), e0163785. doi: 10.1371/journal.pone.0163785.
- Bavoil, L. & Sainz, M. (2008). *Screen Space Ambient Occlusion*.
- Bhattacharya, P. & Wild, D. (1996). A new edge detector for gray volumetric data. *Computers in Biology and Medicine*, 26(4), 315–328. doi: 10.1016/0010-4825(96)00003-0.
- Blinn, J. F. (1977). Models of light reflection for computer synthesized pictures. *Proceedings of the 4th annual conference on Computer graphics and interactive techniques*, (SIGGRAPH '77), 192–198. doi: 10.1145/563858.563893.
- Brooke, J. (1996). SUS: A "quick and dirty" usability scale. In *Usability Evaluation in Industry* (pp. 189–194). Taylor & Francis. Retrieved from: <https://www.doi.org/10.1201/9781498710411-35>.
- Bruckner, S., Grimm, S., Kanitsar, A. & Groller, M. (2006). Illustrative Context-Preserving Exploration of Volume Data. *IEEE Transactions on Visualization and Computer Graphics*, 12(6), 1559–1569. doi: 10.1109/TVCG.2006.96.
- Calkins, C. M., Franciosi, J. P. & Kolesari, G. L. (1999). Human anatomical science and illustration: The origin of two inseparable disciplines. *Clinical Anatomy*, 12(2), 120–129. doi: 10.1002/(SICI)1098-2353(1999)12:2<120::AID-CA7>3.0.CO;2-V. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/%28SICI%291098-2353%281999%2912%3A2%3C120%3A%3AAID-CA7%3E3.0.CO%3B2-V>.
- Callahan, S. P., Callahan, J. H., Scheidegger, C. E. & Silva, C. T. (2008). Direct Volume Rendering: A 3D Plotting Technique for Scientific Data. *Computing in Science Engineering*, 10(1), 88–92. doi: 10.1109/MCSE.2008.7. Conference Name: Computing in Science Engineering.

- Carl, B., Bopp, M., Benescu, A., Saß, B. & Nimsky, C. (2020). Indocyanine Green Angiography Visualized by Augmented Reality in Aneurysm Surgery. *World Neurosurgery*, 142, e307–e315. doi: 10.1016/j.wneu.2020.06.219.
- Chen, W., Hua, W., Bao, H. & Peng, Q. (2003). Real-time ray casting rendering of volume clipping in medical visualization. *Journal of Computer Science and Technology*, 18(6), 804–814. doi: 10.1007/BF02945470.
- Chiang, P.-Y., Chen, C.-C. & Hsia, C.-H. (2019). A touchless interaction interface for observing medical imaging. *Journal of Visual Communication and Image Representation*, 58, 363–373. doi: 10.1016/j.jvcir.2018.12.004.
- Chu, Y., Li, X., Yang, X., Ai, D., Huang, Y., Song, H., Jiang, Y., Wang, Y., Chen, X. & Yang, J. (2018). Perception enhancement using importance-driven hybrid rendering for augmented reality based endoscopic surgical navigation. *Biomedical Optics Express*, 9(11), 5205–5226. doi: 10.1364/BOE.9.005205.
- Cohen, J. (2013). *Statistical Power Analysis for the Behavioral Sciences* (ed. 2). New York: Routledge. doi: 10.4324/9780203771587.
- Conn, C., Lanier, J., Minsky, M., Fisher, S. & Druin, A. (1989). Virtual environments and interactivity: windows to the future. *ACM SIGGRAPH 89 Panel Proceedings*, (SIGGRAPH '89), 7–18. doi: 10.1145/77276.77278.
- Correa, C. & Ma, K.-L. (2008). Size-based Transfer Functions: A New Volume Exploration Technique. *IEEE Transactions on Visualization and Computer Graphics*, 14(6), 1380–1387. doi: 10.1109/TVCG.2008.162. Conference Name: IEEE Transactions on Visualization and Computer Graphics.
- Correa, C. & Ma, K.-L. (2009). The Occlusion Spectrum for Volume Classification and Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 15(6), 1465–1472. doi: 10.1109/TVCG.2009.189.
- Correa, C., Silver, D. & Chen, M. (2006). Feature Aligned Volume Manipulation for Illustration and Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 12(5), 1069–1076. doi: 10.1109/TVCG.2006.144.
- Cotofana, S. & Lachman, N. (2020). The Superficial Face Dissection as an Example for Integrating Clinical Approaches, Authentic Learning, and Changing Perspectives in Anatomy Dissection. *Anatomical Sciences Education*, 13(1), 117–121. doi: 10.1002/ase.1881. _eprint: <https://anatomypubs.onlinelibrary.wiley.com/doi/pdf/10.1002/ase.1881>.

- Cullip, T. J. & Neumann, U. (1993). *Accelerating volume reconstruction with 3D texture hardware*.
- Cutting, J. E. (1997). How the eye measures reality and virtual reality. *Behavior Research Methods, Instruments, & Computers*, 29(1), 27–36. doi: 10.3758/BF03200563.
- Cutting, J. E. & Vishton, P. M. (1995). Chapter 3 - Perceiving Layout and Knowing Distances: The Integration, Relative Potency, and Contextual Use of Different Information about Depth*. In Epstein, W. & Rogers, S. (Eds.), *Perception of Space and Motion* (pp. 69–117). San Diego: Academic Press. doi: 10.1016/B978-012240530-3/50005-5.
- Drouin, S. & Collins, D. L. (2018). PRISM: An open source framework for the interactive design of GPU volume rendering shaders. *PLOS ONE*, 13(3), e0193636. doi: 10.1371/journal.pone.0193636.
- Drouin, S., Kersten-Oertel, M. & Louis Collins, D. (2015). Interaction-Based Registration Correction for Improved Augmented Reality Overlay in Neurosurgery. *Augmented Environments for Computer-Assisted Interventions*, (Lecture Notes in Computer Science), 21–29. doi: 10.1007/978-3-319-24601-7_3.
- Drouin, S., Collins, L. & Kersten-Oertel, M. (2018). Interaction in Augmented Reality Image-Guided Surgery. In *Mixed and Augmented Reality in Medicine* (ed. 1st ed., pp. 99–114). Boca Raton: CRC Press. doi: 10.1201/9781315157702-7.
- Drouin, S., Giovanni, D. A. D., Kersten-Oertel, M. & Collins, D. L. (2020). Interaction Driven Enhancement of Depth Perception in Angiographic Volumes. *IEEE Transactions on Visualization and Computer Graphics*, 26(6), 2247–2257. doi: 10.1109/TVCG.2018.2884940. Conference Name: IEEE Transactions on Visualization and Computer Graphics.
- Díaz, J., Monclús, E., Navazo, I. & Vázquez, P. (2012). Adaptive Cross-sections of Anatomical Models. *Computer Graphics Forum*, 31(7), 2155–2164. doi: 10.1111/j.1467-8659.2012.03208.x. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8659.2012.03208.x>.
- Díaz, J., Vázquez, P.-P., Navazo, I. & Duguet, F. (2010). Real-time ambient occlusion and halos with Summed Area Tables. *Computers and Graphics*, 34(4), 337–350. doi: 10.1016/j.cag.2010.03.005.
- Elmqvist, N. & Tsigas, P. (2008). A Taxonomy of 3D Occlusion Management for Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 14(5), 1095–1109. doi: 10.1109/TVCG.2008.59.

- Engel, D. & Ropinski, T. (2021). Deep Volumetric Ambient Occlusion. *IEEE Transactions on Visualization and Computer Graphics*, 27(2), 1268–1278. doi: 10.1109/TVCG.2020.3030344. Conference Name: IEEE Transactions on Visualization and Computer Graphics.
- Everitt, C. (2001). *Interactive Order-Independent Transparency*. Retrieved on 2022-06-09 from: <https://www.semanticscholar.org/paper/Interactive-Order-Independent-Transparency-Everitt/99b8940b5a6dab8527198e966c0eb7e2a02ee28c>.
- Fedorov, A., Beichel, R., Kalpathy-Cramer, J., Finet, J., Fillion-Robin, J.-C., Pujol, S., Bauer, C., Jennings, D., Fennessy, F., Sonka, M., Buatti, J., Aylward, S., Miller, J. V., Pieper, S. & Kikinis, R. (2012). 3D Slicer as an image computing platform for the Quantitative Imaging Network. *Magnetic Resonance Imaging*, 30(9), 1323–1341. doi: 10.1016/j.mri.2012.05.001.
- Fisher, S. S., McGreevy, M., Humphries, J. & Robinett, W. (1987). Virtual environment display system. *Proceedings of the 1986 workshop on Interactive 3D graphics*, (I3D '86), 77–87. doi: 10.1145/319120.319127.
- Fulvio, J. M. & Rokers, B. (2017). Use of cues in virtual reality depends on visual feedback. *Scientific Reports*, 7(1), 16009. doi: 10.1038/s41598-017-16161-3.
- Gaeta, M., Galletta, K., Cavallaro, M., Mormina, E., Cannizzaro, M. T., Lanzafame, L. R. M., D'Angelo, T., Blandino, A., Vinci, S. L. & Granata, F. (2024). T1 relaxation: Chemo-physical fundamentals of magnetic resonance imaging and clinical applications. *Insights into Imaging*, 15(1), 200. doi: 10.1186/s13244-024-01744-2.
- Gallagher, T. A., Nemeth, A. J. & Hacein-Bey, L. (2008). An Introduction to the Fourier Transform: Relationship to MRI. *American Journal of Roentgenology*, 190(5), 1396–1405. doi: 10.2214/AJR.07.2874.
- Gibson, E., Gibson, J., Smith, O. W. & Flock, H. (1959). Motion parallax as a determinant of perceived depth. *Journal of experimental psychology*. doi: 10.1037/H0043883.
- Guibal, C. R. C. & Dresch, B. (2004). Interaction of color and geometric cues in depth perception: When does “red” mean “near”? *Psychological Research*, 69(1), 30–40. doi: 10.1007/s00426-003-0167-0.
- Hale, K. S. & Stanney, K. M. (2014). *Handbook of Virtual Environments: Design, Implementation, and Applications, Second Edition*. CRC Press.
- Hansen, C. D. & Johnson, C. R. (2011). *Visualization Handbook*. Elsevier.

- Harman, K. L., Humphrey, G. K. & Goodale, M. A. (1999). Active manual control of object views facilitates visual recognition. *Current Biology*, 9(22), 1315–1318. doi: 10.1016/S0960-9822(00)80053-6.
- Hart, S. G. & Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In Hancock, P. A. & Meshkati, N. (Eds.), *Advances in Psychology* (vol. 52, pp. 139–183). North-Holland. doi: 10.1016/S0166-4115(08)62386-9.
- He, L., Guayaquil-Sosa, A. & McGraw, T. (2017). Medical Image Atlas Interaction in Virtual Reality. *IEEE Vis.*, pp. 5.
- Heinrich, F., Bornemann, K., Lawonn, K. & Hansen, C. (2020). Interacting with Medical Volume Data in Projective Augmented Reality. *Medical Image Computing and Computer Assisted Intervention – MICCAI 2020*, (Lecture Notes in Computer Science), 429–439. doi: 10.1007/978-3-030-59716-0_41.
- Heinrich, F., Apilla, V., Lawonn, K., Hansen, C., Preim, B. & Meuschke, M. (2021). Estimating depth information of vascular models: A comparative user study between a virtual reality and a desktop application. *Computers & Graphics*, 98, 210–217. doi: 10.1016/j.cag.2021.05.014.
- Hernell, F., Ljung, P. & Ynnerman, A. (2007). *Efficient Ambient and Emissive Tissue Illumination using Local Occlusion in Multiresolution Volume Rendering*. The Eurographics Association. doi: 10.2312/VG/VG07/001-008.
- Hernell, F., Ljung, P. & Ynnerman, A. (2010). Local Ambient Occlusion in Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 16(4), 548–559. doi: 10.1109/TVCG.2009.45. Conference Name: IEEE Transactions on Visualization and Computer Graphics.
- Hibbard, P. B., van Dam, L. C. & Scarfe, P. (2020). The Implications of Interpupillary Distance Variability for Virtual Reality. *2020 International Conference on 3D Immersion (IC3D)*, pp. 1–7. doi: 10.1109/IC3D51119.2020.9376369.
- Hoffman, D. M., Girshick, A. R., Akeley, K. & Banks, M. S. (2008). Vergence-accommodation conflicts hinder visual performance and cause visual fatigue. *Journal of Vision*, 8(3), 33.1–30. doi: 10.1167/8.3.33.
- Isensee, F., Jaeger, P. F., Kohl, S. A. A., Petersen, J. & Maier-Hein, K. H. (2021). nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, 18(2), 203–211. doi: 10.1038/s41592-020-01008-z.

- Jayakrishnan, V. K., White, P. M., Aitken, D., Crane, P., McMahon, A. D. & Teasdale, E. M. (2003). Subtraction Helical CT Angiography of Intra- and Extracranial Vessels. *AJNR: American Journal of Neuroradiology*, 24(3), 451–455. Retrieved from: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7973630/>.
- Jerald, J. (2015). *The VR Book: Human-Centered Design for Virtual Reality*. Association for Computing Machinery and Morgan & Claypool. Retrieved from: <http://doi.org/10.1145/2792790>.
- Johnson, P. T. & Fishman, E. K. (2010). CT Angiography of Coronary Artery Aneurysms: Detection, Definition, Causes, and Treatment. *American Journal of Roentgenology*, 195(4), 928–934. doi: 10.2214/AJR.09.3517.
- Joshi, A., Scheinost, D., Vives, K., Spencer, D., Staib, L. & Papademetris, X. (2008). Novel interaction techniques for neurosurgical planning and stereotactic navigation. *IEEE Transactions on Visualization and Computer Graphics*, 14(6), 1587–1594. doi: 10.1109/TVCG.2008.150. Conference Name: IEEE Transactions on Visualization and Computer Graphics.
- Keinert, B., Innmann, M., Sanger, M. & Stamminger, M. (2015). Spherical fibonacci mapping. *ACM Transactions on Graphics*, 34(6), 193:1–193:7. doi: 10.1145/2816795.2818131.
- Kerbl, B., Kopanas, G., Leimkuehler, T. & Drettakis, G. (2023). 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.*, 42(4), 139:1–139:14. doi: 10.1145/3592433.
- Kersten-Oertel, M., Chen, S. J.-S. & Collins, D. L. (2014). An Evaluation of Depth Enhancing Perceptual Cues for Vascular Volume Visualization in Neurosurgery. *IEEE Transactions on Visualization and Computer Graphics*, 20(3), 391–403. doi: 10.1109/TVCG.2013.240.
- Kikinis, R. & Pieper, S. (2011). 3D Slicer as a tool for interactive brain tumor segmentation. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, 2011, 6982–6984. doi: 10.1109/IEMBS.2011.6091765.
- Komaitis, S., Stranjalis, G., Kalamatianos, T., Drosos, E., Kalyvas, A. V., Skandalakis, G. P., Liouta, E., Charalampopoulou, E., Mazarakis, N. & Koutsarnakis, C. (2022). A stepwise laboratory manual for the dissection and illustration of limbic and paralimbic structures: lessons learned from the Klingler’s technique. *Surgical and Radiologic Anatomy*, 44(7), 1045–1061. doi: 10.1007/s00276-022-02981-0.

- Koulouris, G. A., Akşit, K., Stengel, M., Mantiuk, R. K., Mania, K. & Richardt, C. (2019). Near-Eye Display and Tracking Technologies for Virtual and Augmented Reality. *Computer Graphics Forum*, 38(2), 493–519. doi: 10.1111/cgf.13654. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13654>.
- Koutsarnakis, C., Liakos, F., Kalyvas, A. V., Sakas, D. E. & Stranjalis, G. (2015). A Laboratory Manual for Stepwise Cerebral White Matter Fiber Dissection. *World Neurosurgery*, 84(2), 483–493. doi: 10.1016/j.wneu.2015.04.018.
- Kreiser, J., Hermosilla, P. & Ropinski, T. (2018). Void Space Surfaces to Convey Depth in Vessel Visualizations. *arXiv:1806.07729 [cs]*. Retrieved from: <http://arxiv.org/abs/1806.07729>. arXiv: 1806.07729.
- Kroes, T., Post, F. H. & Botha, C. P. (2012). Exposure Render: An Interactive Photo-Realistic Volume Rendering Framework. *PLOS ONE*, 7(7), e38586. doi: 10.1371/journal.pone.0038586.
- Kruger, J. & Westermann, R. (2003). Acceleration techniques for GPU-based volume rendering. *IEEE Visualization, 2003. VIS 2003.*, pp. 287–292. doi: 10.1109/VISUAL.2003.1250384.
- Kumamaru, K. K., Hoppel, B. E., Mather, R. T. & Rybicki, F. J. (2010). CT Angiography: Current Technology and Clinical Use. *Radiologic clinics of North America*, 48(2), 213–235. doi: 10.1016/j.rcl.2010.02.006.
- Lacroute, P. & Levoy, M. (1994). Fast volume rendering using a shear-warp factorization of the viewing transformation. *Proceedings of the 21st annual conference on Computer graphics and interactive techniques, (SIGGRAPH '94)*, 451–458. doi: 10.1145/192161.192283.
- Lagos, K. & McInerney, T. (2019). Fast Contextual View Generation in 3D Medical Images Using a 3D Widget User Interface and Super-Ellipsoids. *Advances in Visual Computing, (Lecture Notes in Computer Science)*, 455–468. doi: 10.1007/978-3-030-33723-0_37.
- Laha, B. & Bowman, D. A. (2013). Volume cracker: a bimanual 3D interaction technique for analysis of raw volumetric data. *Proceedings of the 1st symposium on Spatial user interaction, (SUI '13)*, 61–68. doi: 10.1145/2491367.2491368.
- Laha, B., Bowman, D. A. & Socha, J. J. (2016). Bare-Hand Volume Cracker for Raw Volume Data Analysis. *Frontiers in Robotics and AI*, 3. doi: 10.3389/frobt.2016.00056.
- LaValle, S. M., Yershova, A., Katsev, M. & Antonov, M. (2014). Head tracking for the Oculus Rift. *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 187–194. doi: 10.1109/ICRA.2014.6906608.

- Lawonn, K., Luz, M., Preim, B. & Hansen, C. (2015). Illustrative Visualization of Vascular Models for Static 2D Representations. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, pp. 399–406.
- Lawonn, K., Luz, M. & Hansen, C. (2017). Improving spatial perception of vascular models using supporting anchors and illustrative visualization. *Computers & Graphics*, 63, 37–49. doi: 10.1016/j.cag.2017.02.002.
- Lescop, L. (2017). 360° vision, from panoramas to VR. *ENVISIONING ARCHITECTURE: SPACE / TIME / MEANING*, 1(ENVISIONING ARCHITECTURE: SPACE / TIME / MEANING), 226–232. Retrieved from: <https://hal.science/hal-01584883>.
- Levoy, M. (1988). Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, 8(3), 29–37. doi: 10.1109/38.511.
- Li, C., Tong, Y., Chen, K., Yang, Z., Li, R., Qiu, S., Chan, J. Y.-K., Heng, P.-A. & Dou, Q. [arXiv:2507.06647 [cs]]. (2025). ClipGS: Clippable Gaussian Splatting for Interactive Cinematic Visualization of Volumetric Medical Data. arXiv. Retrieved on 2025-09-26 from: <http://arxiv.org/abs/2507.06647>.
- Lindemann, F. & Ropinski, T. (2011). About the Influence of Illumination Models on Image Comprehension in Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 17(12), 1922–1931. doi: 10.1109/TVCG.2011.161. Conference Name: IEEE Transactions on Visualization and Computer Graphics.
- Livitcuka, R., Alksnis, R. & Pladere, T. (2023). Impact of interpupillary distance mismatch on visual aftereffects of virtual reality gameplay. *Optical Architectures for Displays and Sensing in Augmented, Virtual, and Mixed Reality (AR, VR, MR) IV*, 12449, 377–380. doi: 10.1117/12.2666238.
- Ljung, P., Krüger, J., Groller, E., Hadwiger, M., Hansen, C. D. & Ynnerman, A. (2016). State of the Art in Transfer Functions for Direct Volume Rendering. *Computer Graphics Forum*, 35(3), 669–691. doi: 10.1111/cgf.12934. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12934>.
- Lorensen, W. E. & Cline, H. E. (1998). Marching cubes: a high resolution 3D surface construction algorithm. In *Seminal graphics: pioneering efforts that shaped the field, Volume 1* (vol. Volume 1, pp. 347–353). New York, NY, USA: Association for Computing Machinery. Retrieved from: <https://doi.org/10.1145/280811.281026>.
- Lottes, T. (2009). *FXAA*.

- Lu, Y., Ota, K. & Dong, M. (2024). An Empirical Study of VR Head-Mounted Displays Based on VR Games Reviews. *ACM Games*, 2(3), 26:1–26:20. doi: 10.1145/3665988.
- Lugauer, F. & Wetzl, J. (2018). Magnetic Resonance Imaging. In Maier, A., Steidl, S., Christlein, V. & Hornegger, J. (Eds.), *Medical Imaging Systems: An Introductory Guide*. Cham (CH): Springer. Retrieved from: <http://www.ncbi.nlm.nih.gov/books/NBK546152/>.
- MacKenzie, I. S., Soukoreff, R. W. & Pal, C. (1997). A two-ball mouse affords three degrees of freedom. *CHI '97 Extended Abstracts on Human Factors in Computing Systems*, (CHI EA '97), 303–304. doi: 10.1145/1120212.1120405.
- MacLise, J. (1859). *Surgical Anatomy*. Blanchard and Lea. Retrieved from: <https://www.gutenberg.org/ebooks/24440>.
- Max, N. (1995). Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2), 99–108. doi: 10.1109/2945.468400. Conference Name: IEEE Transactions on Visualization and Computer Graphics.
- Mazuryk, T. & Gervautz, M. (2015). History, Applications, Technology and Future.
- McGuffin, M., Tancu, L. & Balakrishnan, R. (2003). Using deformations for browsing volumetric data. *IEEE Visualization, 2003. VIS 2003.*, pp. 401–408. doi: 10.1109/VISUAL.2003.1250400.
- Meijer, F. J. A., Schuijf, J. D., de Vries, J., Boogaarts, H. D., van der Woude, W.-J. & Prokop, M. (2019). Ultra-high-resolution subtraction CT angiography in the follow-up of treated intracranial aneurysms. *Insights into Imaging*, 10(1), 2. doi: 10.1186/s13244-019-0685-y.
- Melzer, J. E., Brozoski, F. T., Letowski, T. R., Harding, T. H. & Rash, C. E. [Institution: American Psychological Association]. (2009). Guidelines for HMD design: (614362011-018). Retrieved on 2025-10-06 from: <https://doi.apa.org/doi/10.1037/e614362011-018>.
- Miller, G. (1994). Efficient algorithms for local and global accessibility shading. *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, (SIGGRAPH '94), 319–326. doi: 10.1145/192161.192244.
- Mueller, K., Shareef, N., Huang, J. & Crawfis, R. (1999). High-quality splatting on rectilinear grids with efficient culling of occluded voxels. *IEEE Transactions on Visualization and Computer Graphics*, 5(2), 116–134. doi: 10.1109/2945.773804. Conference Name: IEEE Transactions on Visualization and Computer Graphics.

- Naicker, J. (2023). Medical Illustration in Anatomy. In Shapiro, L. (Ed.), *Graphic Medicine, Humanizing Healthcare and Novel Approaches in Anatomical Education* (pp. 63–83). Cham: Springer Nature Switzerland. doi: 10.1007/978-3-031-39035-7_4.
- Opolski, M. P., Staruch, A. D., Jakubczyk, M., Min, J. K., Gransar, H., Staruch, M., Witkowski, A., Kepka, C., Kim, W.-K., Hamm, C. W., Möllmann, H. & Achenbach, S. (2016). CT Angiography for the Detection of Coronary Artery Stenoses in Patients Referred for Cardiac Valve Surgery: Systematic Review and Meta-Analysis. *JACC: Cardiovascular Imaging*, 9(9), 1059–1070. doi: 10.1016/j.jcmg.2015.09.028.
- Pai, A., Shetty, R., Hodis, B. & Chowdhury, Y. S. (2025). Magnetic Resonance Imaging Physics. In *StatPearls*. Treasure Island (FL): StatPearls Publishing. Retrieved from: <http://www.ncbi.nlm.nih.gov/books/NBK564320/>.
- Palmer, S. E. (1999). *Vision Science: Photons to Phenomenology*. Cambridge, MA, USA: A Bradford Book.
- Papa, V., Varotto, E., Vaccarezza, M. & Galassi, F. M. (2021). Teaching Anatomy Through Images: The Power of Anatomical Drawings. *Anthropologie (1962-)*, 59(2), 145–154. doi: 10.26720/anthro.21.03.29.2.
- Pfister, H. (2005). Hardware-Accelerated Volume Rendering. In *Visualization Handbook* (pp. 229–258). Elsevier. doi: 10.1016/B978-012387582-2/50013-7.
- Pham, D. L., Xu, C. & Prince, J. L. (2000). Current Methods in Medical Image Segmentation1. *Annual Review of Biomedical Engineering*, 2(Volume 2, 2000), 315–337. doi: 10.1146/annurev.bioeng.2.1.315.
- Phong, B. T. (1975). Illumination for computer generated pictures. *Communications of the ACM*, 18(6), 311–317. doi: 10.1145/360825.360839.
- Plackett, R. L. (1975). The Analysis of Permutations. *Applied Statistics*, 24(2), 193. doi: 10.2307/2346567.
- Preim, B. & Bartz, D. (2007). *Visualization in Medicine: Theory, Algorithms, and Applications*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
- Preim, B. & Botha, C. P. (2013). *Visual Computing for Medicine: Theory, Algorithms, and Applications*. Morgan Kaufmann.

- Preim, B., Baer, A., Cunningham, D., Isenberg, T. & Ropinski, T. (2016). A Survey of Perceptually Motivated 3D Visualization of Medical Image Data. *Computer Graphics Forum*, 35(3), 501–525. doi: <https://doi.org/10.1111/cgf.12927>. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.12927>.
- Prince, J. L. & Links, J. (2014). *Medical Imaging Signals and Systems*. Pearson Education.
- Qian, N. (1997). Binocular Disparity and the Perception of Depth. *Neuron*, 18(3), 359–368. doi: 10.1016/S0896-6273(00)81238-6.
- Rister, B., Yi, D., Shivakumar, K., Nobashi, T. & Rubin, D. L. (2020). CT-ORG, a new dataset for multiple organ segmentation in computed tomography. *Scientific Data*, 7(1), 381. doi: 10.1038/s41597-020-00715-8.
- Roethe, A. L., Rösler, J., Misch, M., Vajkoczy, P. & Picht, T. (2022). Augmented reality visualization in brain lesions: a prospective randomized controlled evaluation of its potential and current limitations in navigated microneurosurgery. *Acta Neurochirurgica*, 164(1), 3–14. doi: 10.1007/s00701-021-05045-1.
- Ropinski, T., Steinicke, F. & Hinrichs, K. (2006). Visually Supporting Depth Perception in Angiography Imaging. *Smart Graphics*, (Lecture Notes in Computer Science), 93–104. doi: 10.1007/11795018_9.
- Ropinski, T., Meyer-Spradow, J., Diepenbrock, S., Mensmann, J. & Hinrichs, K. (2008). Interactive Volume Rendering with Dynamic Ambient Occlusion and Color Bleeding. *Computer Graphics Forum*, 27(2), 567–576. doi: 10.1111/j.1467-8659.2008.01154.x. _eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-8659.2008.01154.x>.
- Ruia, S. & Tripathy, K. (2025). Humphrey Visual Field. In *StatPearls*. Treasure Island (FL): StatPearls Publishing. Retrieved from: <http://www.ncbi.nlm.nih.gov/books/NBK585112/>.
- Schlegel, P., Makhinya, M. & Pajarola, R. (2011). Extinction-Based Shading and Illumination in GPU Volume Ray-Casting. *IEEE Transactions on Visualization and Computer Graphics*, 17(12), 1795–1802. doi: 10.1109/TVCG.2011.198. Conference Name: IEEE Transactions on Visualization and Computer Graphics.
- Schneider, R. (2013). Minkowski addition. In *Convex Bodies: The Brunn–Minkowski Theory* (ed. 2, pp. 139–207). Cambridge: Cambridge University Press. doi: 10.1017/CBO9781139003858.006.

- Serra, L., Kockro, R. A., Guan, C. G., Hern, N., Lee, E. C. K., Lee, Y. H., Chan, C. & Nowinski, W. L. (1998). Multimodal volume-based tumor neurosurgery planning in the virtual workbench. *Medical Image Computing and Computer-Assisted Intervention — MICCAI'98*, (Lecture Notes in Computer Science), 1007–1015. doi: 10.1007/BFb0056289.
- Shi, P., Billeter, M. & Eisemann, E. (2022). Stereo-consistent screen-space ambient occlusion. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 5(1), 2:1–2:12. doi: 10.1145/3522614.
- Shinohara, R. T., Sweeney, E. M., Goldsmith, J., Shiee, N., Mateen, F. J., Calabresi, P. A., Jarso, S., Pham, D. L., Reich, D. S. & Crainiceanu, C. M. (2014). Statistical normalization techniques for magnetic resonance imaging. *NeuroImage: Clinical*, 6, 9–19. doi: 10.1016/j.nicl.2014.08.008.
- Simons, D. J., Wang, R. F. & Roddenberry, D. (2002). Object recognition is mediated by extraretinal information. *Perception & Psychophysics*, 64(4), 521–530. doi: 10.3758/BF03194723.
- Standring, S. (Ed.). (2015). *Gray's Anatomy*. Elsevier. Retrieved from: <https://shop.elsevier.com/books/gray-s-anatomy/standring/978-0-7020-5230-9>.
- Stewart, A. (2003). Vicinity shading for enhanced perception of volumetric data. *IEEE Visualization, 2003. VIS 2003.*, pp. 355–362. doi: 10.1109/VISUAL.2003.1250394.
- Sutherland, I. E. (1968). A head-mounted three dimensional display. *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, (AFIPS '68 (Fall, part I)), 757–764. doi: 10.1145/1476589.1476686.
- Sutherland, J., Belec, J., Sheikh, A., Chepelev, L., Althobaity, W., Chow, B. J. W., Mitsouras, D., Christensen, A., Rybicki, F. J. & La Russa, D. J. (2019). Applying Modern Virtual and Augmented Reality Technologies to Medical Images and Models. *Journal of Digital Imaging*, 32(1), 38–53. doi: 10.1007/s10278-018-0122-7.
- Sweet, B. T. & Kaiser, M. K. (2013). Choosing Your Poison: Optimizing Simulator Visual System Selection as a Function of Operational Tasks. Retrieved from: <https://ntrs.nasa.gov/citations/20130014740>.
- Tanzi, L., Piazzolla, P., Porpiglia, F. & Vezzetti, E. (2021). Real-time deep learning semantic segmentation during intra-operative surgery for 3D augmented reality assistance. *International Journal of Computer Assisted Radiology and Surgery*, 16(9), 1435–1445. doi: 10.1007/s11548-021-02432-y.

- Teittinen, M. (1993). Depth Cues in the Human Visual System. *The encyclopedia of virtual environments*, 1. Retrieved from: http://www.hitl.washington.edu/projects/knowledge_base/virtual-worlds/EVE/III.A.1.c.DepthCues.html.
- Thamrongrat, P., Khundam, C., Pakdeebun, P. & Nizam, D. N. M. (2023). Desktop vs. Headset: A Comparative Study of User Experience and Engagement for Flexibility Exercise in Virtual Reality. *Emerging Science Journal*, 7(4), 1063–1082. doi: 10.28991/ESJ-2023-07-04-03.
- Thompson, P., May, K. & Stone, R. (1993). Chromostereopsis: a multicomponent depth effect? *Displays*, 14(4), 227–234. doi: 10.1016/0141-9382(93)90093-K.
- Titov, A., Kersten-Oertel, M. & Drouin, S. (2022). The Effect of Interactive Cues on the Perception of Angiographic Volumes in Virtual Reality. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 10(4), 357–365. doi: 10.1080/21681163.2021.1999332. _eprint: <https://doi.org/10.1080/21681163.2021.1999332>.
- Titov, A., Kersten-Oertel, M. & Drouin, S. (2023). Contextual Ambient Occlusion. Retrieved from: <https://openreview.net/forum?id=5bd-6wyFWg5>.
- Titov, A., Kersten-Oertel, M. & Drouin, S. (2024). Contextual Ambient Occlusion: A volumetric rendering technique that supports real-time clipping. *Computers & Graphics*, 119, 103884. doi: 10.1016/j.cag.2024.01.011.
- Toulouse, A., Thiele, S. & Herkommer, A. (2019). Virtual reality headset using a gaze-synchronized display system. *Optical Design Challenge 2019*, 11040, 1104009. doi: 10.1117/12.2523920.
- van Ooijen, P. M. A., van Geuns, R. J. M., Rensing, B. J. W. M., Bongaerts, A. H. H., de Feyter, P. J. & Oudkerk, M. (2003). Noninvasive Coronary Imaging Using Electron Beam CT: Surface Rendering Versus Volume Rendering. *American Journal of Roentgenology*, 180(1), 223–226. doi: 10.2214/ajr.180.1.1800223.
- Vienne, C., Masfrand, S., Bourdin, C. & Vercher, J.-L. (2020). Depth Perception in Virtual Reality Systems: Effect of Screen Distance, Environment Richness and Display Factors. *IEEE Access*, 8, 29099–29110. doi: 10.1109/ACCESS.2020.2972122. Conference Name: IEEE Access.
- Viola, I., Kanitsar, A. & Groller, M. (2004). Importance-driven volume rendering. *IEEE Visualization 2004*, pp. 139–145. doi: 10.1109/VISUAL.2004.48.

- Wang, L., Zhao, Y., Mueller, K. & Kaufman, A. (2005). The magic volume lens: an interactive focus+context technique for volume rendering. *VIS 05. IEEE Visualization, 2005.*, pp. 367–374. doi: 10.1109/VISUAL.2005.1532818.
- Wang, X. M., Southwick, D., Robinson, I., Nitsche, M., Resch, G., Mazalek, A. & Welsh, T. N. (2024). The geometry of the vergence-accommodation conflict in mixed reality systems. *Virtual Reality*, 28(2), 95. doi: 10.1007/s10055-024-00991-4.
- Ware, C. & Mitchell, P. (2008). Visualizing graphs in three dimensions. *ACM Transactions on Applied Perception*, 5(1), 2:1–2:15. doi: 10.1145/1279640.1279642.
- Wasserthal, J., Breit, H.-C., Meyer, M. T., Pradella, M., Hinck, D., Sauter, A. W., Heye, T., Boll, D. T., Cyriac, J., Yang, S., Bach, M. & Segeroth, M. (2023). TotalSegmentator: Robust Segmentation of 104 Anatomic Structures in CT Images. *Radiology: Artificial Intelligence*, 5(5), e230024. doi: 10.1148/ryai.230024.
- Weiskopf, D., Engel, K. & Ertl, T. (2002). Volume clipping via per-fragment operations in texture-based volume visualization. *IEEE Visualization, 2002. VIS 2002.*, pp. 93–100. doi: 10.1109/VISUAL.2002.1183762.
- Weiskopf, D., Engel, K. & Ertl, T. (2003). Interactive clipping techniques for texture-based volume visualization and volume shading. *IEEE Transactions on Visualization and Computer Graphics*, 9(3), 298–312. doi: 10.1109/TVCG.2003.1207438. Conference Name: IEEE Transactions on Visualization and Computer Graphics.
- Weiss, V. & Casazza, K. (2023). Leveraging the Art of Medical Illustration to Enhance Anatomy Instruction. In Stephen, J. S., Kormpas, G. & Coombe, C. (Eds.), *Global Perspectives on Higher Education: From Crisis to Opportunity* (pp. 213–235). Cham: Springer International Publishing. doi: 10.1007/978-3-031-31646-3_15.
- Williams, L. (1978). Casting curved shadows on curved surfaces. *Proceedings of the 5th annual conference on Computer graphics and interactive techniques*, (SIGGRAPH '78), 270–274. doi: 10.1145/800248.807402.
- Xiao, L., Zhao, Y., Lindberg, D., Hegland, J., Moczydowski, S., Penner, E., Tebbs, D., Terpstra, D., Ender, I., Lin, Y.-J., Majors, J. & Lanman, D. (2025). Wide Field-of-View Mixed Reality. *ACM SIGGRAPH 2025 Emerging Technologies*, (SIGGRAPH '25), 1–2. doi: 10.1145/3721257.3734021.
- Zhai, S., Buxton, W. & Milgram, P. (1996). The partial-occlusion effect: utilizing semitransparency in 3D human-computer interaction. *ACM Transactions on Computer-Human Interaction*, 3(3), 254–284. doi: 10.1145/234526.234532.

Zheng, J., Chan, K. & Gibson, I. (1998). Virtual reality. *Potentials, IEEE*, 17, 20–23.
doi: 10.1109/45.666641.