Linköping studies in science and technology.
Dissertation, No. 1789

ENHANCING SALIENT FEATURES IN VOLUMETRIC DATA
USING ILLUMINATION AND TRANSFER FUNCTIONS

Daniel Jönsson

Division of Media and Information Technology
Department of Science and Technology
Linköping University, SE-601 74 Norrköping, Sweden
Norrköping, October 2016

**Enhancing Salient Features in Volumetric Data Using Illumination
and Transfer Functions**

Copyright © 2016 Daniel Jönsson (unless otherwise noted)

*Division of Media and Information Technology
Department of Science and Technology
Campus Norrköping, Linköping University
SE-601 74 Norrköping, Sweden*

ISBN: 978-91-7685-689-5

ISSN: 0345-7524

Printed in Sweden by LiU-Tryck, Linköping, 2016

Acknowledgments

The work resulting in this thesis would not have been possible without my enthusiastic and incredibly smart supervisors, colleagues and co-authors. Hours of interesting discussions have been spent over coffee and many whiteboards have been filled during those discussions. I would like to express my sincere gratitude to my supervisor Anders Ynnerman who has provided guidance and inspiration. Thanks also goes to my co-supervisors Timo Ropinski and Patric Ljung. Timo was my co-supervisor during my first years as a PhD and was very supportive and helpful during those initial confusing years. Patric took over the role as co-supervisor during the last time and has acted as an excellent sounding board.

Colleagues with whom work and non-work related discussions have been made include Alexander Bock, who is the kindest metal fan I know, Andrew Gardner who always has a fist bump at hand, Erik Sundén who I would like to thank for applying the VIS-police approval approach on the papers, Joel Kronander who is bringing a theoretic point of view into the discussions, Martin Falk who has provided both climbing and paper experiences, Peter Steneteg who has contributed with his programming mastermind, Rickard Englund who may be the secret chancellor?, Sathish Kotravel who has an amazing ability to come up with one-liners, Stefan Lindholm with whom I had thorough discussions on the discreteness of nature and Tan Khoa Nguen who taught me how a proper bribing of a Vietnamese border control officer should be done. Our administrator Eva Skärblom deserves a special thanks for her immaculate work. I would also like to thank the Center of Medical Imaging and Visualization (CMIV) for providing interesting data to visualize. I have also had the opportunity to work together with Isabelle Wegmann Hachette and Gunnar Låthen at Context Vision AB, Claes Lundström at Sectra AB and Katarina Sperling at the Norrköping Visualization Center. These companies show that my work as a PhD student extend beyond the published papers. Several people have proofread this work and made comments that significantly improved it. Thank you Martin Falk, Andrew Gardner, Joel Kronander, Erik Sundén and Anders Ynnerman.

This work would not have been possible without funding agencies. Many thanks to the Swedish e-Science Research Centre (SeRC), the Swedish Research Council (VR), the Knut and Alice Wallenberg Foundation (KAW) and the Excellence Center at Linköping and Lund in Information Technology (ELLIIT).

I am also grateful for the time spent abroad in Vancouver together with Torsten Möller and collaborations with the people at the SCI institute in Salt Lake City, especially Tiago Etienne.

Finally, I would like to thank my family for illuminating my time out-of-office, <3 Malin and Edvin!

Abstract

The visualization of volume data is a fundamental component in the medical domain. Volume data is used in the clinical work-flow to diagnose patients and is therefore of uttermost importance. The amount of data is rapidly increasing as sensors, such as computed tomography scanners, become capable of measuring more details and gathering more data over time. Unfortunately, the increasing amount of data makes it computationally challenging to interactively apply high quality methods to increase shape and depth perception. Furthermore, methods for exploring volume data has mostly been designed for experts, which prohibits novice users from exploring volume data. This thesis aims to address these challenges by introducing efficient methods for enhancing salient features through high quality illumination as well as methods for intuitive volume data exploration.

Humans are interpreting the world around them by observing how light interacts with objects. Shadows enable us to better determine distances while shifts in color enable us to better distinguish objects and identify their shape. These concepts are also applicable to computer generated content. The perception in volume data visualization can therefore be improved by simulating real-world light interaction. However, realistic light simulation is a computationally challenging problem. This thesis presents efficient methods that are capable of interactively simulating realistic light propagation in volume data. In particular, this work shows how a multi-resolution grid can be used to encode the attenuation of light from all directions using spherical harmonics and thereby enable advanced interactive dynamic light configurations. Two methods are also presented that allow photon mapping calculations to be focused on visually changing areas. The results demonstrate that photon mapping can be used in interactive volume visualization for both static and time-varying volume data.

Efficient and intuitive exploration of volume data requires methods that are easy to use and reflect the objects that were measured. A value that has been collected by a sensor commonly represents the material existing within a small neighborhood around a location. Recreating the original materials is difficult since the value represents a mixture of them. This is referred to as the partial-volume problem. A method is presented that derives knowledge from the user in order to reconstruct the original materials in a way which is more in line with what the user would expect. Sharp boundaries are visualized where the certainty is high while uncertain areas are visualized with fuzzy boundaries. The volume exploration process of mapping data values to optical properties through the transfer function has traditionally been complex and performed by expert users. In this thesis, a dynamic gallery of the data is combined with touch interaction to allow novice users to explore volume data. A study at a science center showed that visitors favor the presented dynamic gallery method compared to the most commonly used transfer function editor.

Populärvetenskaplig Sammanfattning

Medicinska verktyg som till exempel datortomografskannern kan idag samla in tusentals bilder av kroppens inre och tillsammans formar de en volym av data. Visualisering av volymdata är idag en grundläggande komponent i det kliniska arbetsflödet där det används för att ställa diagnoser på patienter. Visualiseringen gör att radiologer kan se inre organ och är därför ett viktigt verktyg i det kliniska arbetsflödet. Mängden information ökar kraftigt i takt med att skannerns förbättrar sin detaljrikedom och förmåga att samla in information över tid. Den ökade detaljrikedomen gör att rätt diagnoser kan ställas med större säkerhet medan information över tid gör det möjligt att analysera organens funktion i kroppen. Det är en stor utmaning att hantera den ökande mängden volymdata och förbättra möjligheterna för användare att tolka informationen. Den här avhandlingen introducerar därför effektiva metoder för att framhäva viktiga delar i datan genom virtuell belysning samt metoder för att förenkla utforskningen av datan.

Människor har sedan födseln tränats i att tolka världen omkring oss genom att observera hur ljus interagerar med objekt i vår omgivning. Skuggor gör att vi kan bedöma avstånd bättre medan färgskiftningar gör att vi kan separera objekt och bedöma dess form. Visualisering av volymdata kan därför förbättras genom att introducera en realistisk belysningsmiljö där vi kan använda vårt synsinne fullt ut för att tolka informationen. Realistisk ljussimulering kräver dock stora mängder beräkningar vilket hindrar den från att användas vid interaktiv visualisering. Det krävs därmed effektiva metoder för att interaktivt kunna simulera realistiskt ljus i volymdata. I den här avhandlingen visas hur avancerade dynamiska ljuskonfigurationer interaktivt kan simuleras. Ljussimulering av högre kvalitet kan uppnås genom att representera ljuset med små energipartiklar kallade fotoner. Att använda fotoner för ljussättning i volymdata är beräkningstungt och har därför inte kunnat göras interaktivt tidigare. Två metoder presenteras som fokuserar beräkningarna på områden som förändras under utforskningen vilket gör att mängden beräkningar som behöver utföras kan reduceras. Resultaten visar att realistisk ljussimulering med fotoner därmed kan användas vid interaktiv visualisering av volymdata.

För att uppnå effektiv och intuitiv utforskning av volymdata krävs metoder som är enkla att använda och återspeglar objekten som mäts. Ett mätvärde som har samlats in av en skanner representerar samtliga objekt inom ett visst område. Att återskapa objekten inom området är svårt eftersom mätvärdet representerar en blandning av objekten. I avhandlingen presenteras en metod som härleder kunskap från användaren för att återskapa objekten på ett sätt som ligger mer i linje med vad användaren förväntar sig. Gränser där objekten kan bestämmas med hög säkerhet återskapas som skarpa medan områden där osäkerheten är hög blir suddiga. Utforskning av volymdata har traditionellt varit en komplex process utförd av experter. Ett dynamiskt bildgalleri av volymdatan kombineras med

pekskärmsinteraktion för att oerfarna användare ska kunna utforska datan. En studie på ett museum visade att fler besökare föredrog det föreslagna dynamiska bildgalleriet jämfört den vanligaste metoden för att utforska volymdata.

Metoderna presenterade i den här avhandlingen är främst demonstrerade på medicinsk data men de kan också appliceras på volymdata från andra områden.

Publications

The following publications are included in this dissertation:

- Paper A:** D. Jönsson, E. Sundén, A. Ynnerman, and T. Ropinski. A Survey of Volumetric Illumination Techniques for Interactive Volume Rendering. *Computer Graphics Forum*, 33(1):27–51, 2014
- Paper B:** J. Kronander, D. Jönsson, J. Löw, P. Ljung, A. Ynnerman, and J. Unger. Efficient Visibility Encoding for Dynamic Illumination in Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 18(3):447–462, 2012
- Paper C:** D. Jönsson, J. Kronander, T. Ropinski, and A. Ynnerman. Historygrams: Enabling Interactive Global Illumination in Direct Volume Rendering Using Photon Mapping. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2364–2371, 2012
- Paper D:** D. Jönsson and A. Ynnerman. Correlated Photon Mapping for Interactive Global Illumination of Time-Varying Volumetric Data. *IEEE Transactions on Visualization and Computer Graphics*, 23, 2017, in press
- Paper E:** S. Lindholm, D. Jönsson, C. Hansen, and A. Ynnerman. Boundary Aware Reconstruction of Scalar Fields. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2447–2455, 2014
- Paper F:** D. Jönsson, M. Falk, and A. Ynnerman. Intuitive Exploration of Volumetric Data Using Dynamic Galleries. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):896–905, 2016

The following publications, listed in reverse chronological order, have been published in relation to the work performed during the thesis but are not included in the dissertation:

- D. Jönsson, E. Sundén, G. Låthén, and I. W. Hachette. Method and System for Volume Rendering of Medical Images, patent pending 2016. [Online]. Available: <http://patentscope.wipo.int/search/en/detail.jsf?docId=US173535325>, Accessed: 2016-09-05
- E. Sundén, A. Bock, D. Jönsson, A. Ynnerman, and T. Ropinski. Interaction Techniques as a Communication Channel when Presenting 3D Visualizations. In *IEEE VIS International Workshop on 3DVis*, pages 2–6, 2014
- J. Parulek, D. Jönsson, T. Ropinski, S. Bruckner, A. Ynnerman, and I. Viola. Continuous Levels-of-Detail and Visual Abstraction for Seamless Molecular Visualization. *Computer Graphics Forum*, 33(6):276–287, 2014
- J. Kronander, J. Dahlin, D. Jönsson, M. Kok, T. Schön, and J. Unger. Real-Time Video Based Lighting Using GPU Raytracing. In *European Signal Processing Conference (EUSIPCO 2014)*, pages 1627–1631, 2014
- T. Etienne, D. Jönsson, T. Ropinski, C. Scheidegger, J. L. D. Comba, L. G. Nonato, R. M. Kirby, A. Ynnerman, and C. T. Silva. Verifying Volume Rendering using Discretization Error Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 20(1):140–154, 2014
- S. Lindholm, D. Jönsson, H. Knutsson, and A. Ynnerman. Towards Data Centric Sampling for Volume Rendering. In *SIGRAD*, pages 55–60, 2013
- N. Rostamzadeh, D. Jönsson, and T. Ropinski. Comparison of Volumetric Illumination Methods by Considering the Underlying Optical Models. In *SIGRAD*, pages 35–40, 2013
- D. Jönsson, E. Sundén, A. Ynnerman, and T. Ropinski. State of The Art Report on Interactive Volume Rendering with Volumetric Illumination. In *EG - State of the Art Reports*, volume 1, pages 53–74, 2012
- D. Jönsson, P. Ganestam, M. Doggett, A. Ynnerman, and T. Ropinski. Explicit Cache Management for Volume Ray-Casting on Parallel Architectures. In *Eurographics Symposium on Parallel Graphics and Visualization*, pages 31–40, 2012

Contributions

Paper A:

A Survey of Volumetric Illumination Techniques for Interactive Volume Rendering

Presents an overview of the state-of-the art within the field of volumetric illumination. Methods within the field are compared with respect to their memory consumption, computational load and capabilities of producing high fidelity illumination when changing the transfer function, light sources or camera.

Paper B:

Efficient Visibility Encoding for Dynamic Illumination in Direct Volume Rendering

Proposes the use of an efficient data structure to store and compute local and global visibility within the data set using spherical harmonics. The computational gain enables complex dynamic light setups to be used within interactive volume visualization. As a second author I was primarily involved in both the development and implementation of the method. The method was presented at the IEEE VIS TVCG track.

Paper C:

Historygrams: Enabling Interactive Global Illumination in Direct Volume Rendering Using Photon Mapping

Introduces a representation of the data history of a photon and view ray that can be efficiently queried and stored. The data history is used to determine whether a photon or view ray is affected by a parameter change and therefore needs to be recomputed. The method increases the performance of the high fidelity photon mapping illumination technique such that it can be used for interactive volume exploration using the transfer function. The method was presented at IEEE VIS (SciVis).

Paper D:

Correlated Photon Mapping for Interactive Global Illumination of Time-Varying Volumetric Data

Utilizes correlation between time-steps in volumetric data to determine and prioritize the subset of photons that need to be updated due to changes. A visual importance function is proposed that takes the data into account as well as the transfer function in order to determine the correlation of visible content between time steps. An approximate photon gathering approach is presented, which allows

the photon density to be retrieved during camera movements using hardware accelerated filtering. The method advances the state-of-the-art performance in volumetric photon mapping such that it can be used for interactive volume exploration of time-varying volumetric data. The method received an honorable mention and is to be presented at IEEE VIS (SciVis).

Paper E:
Boundary Aware Reconstruction of Scalar Fields

Improves the visualization of features in transition areas and at discontinuous material boundaries using encoded domain knowledge. The domain knowledge is supplied through a material classification and combined with the visual classification to reconstruct values that are more in line with the user's understanding of the nature of the data. The results show that visualizations which capture the nature of the data can be created with decreased interaction complexity compared to previous work. As a second author I was involved in developing the idea, the method and writing the manuscript. The method was presented at IEEE VIS (SciVis).

Paper F:
Intuitive Exploration of Volumetric Data Using Dynamic Galleries

Makes volume data exploration for novice users easier through the use of dynamically generated gallery images of the data set. Each gallery image displays a sub-range of the data set's value range. Zooming and panning is used to change the displayed data value range, causing the gallery to dynamically update, and enables the user to get both an overview as well as details on demand. A user study showed that the presented method was preferred compared to a traditional approach without gallery images. The method was presented at IEEE VIS (SciVis).

Contents

Acknowledgments	v
Abstract	vii
Populärvetenskaplig Sammanfattning	ix
List of publications	xi
Contributions	xiii
1 Motivation	1
1.1 Visual analysis of complex information	2
1.2 Enhancing salient features	3
1.3 Visualizing volume data	3
2 Introduction	5
2.1 The visualization pipeline	5
2.2 Static volume data	6
2.3 Time-varying volume data	7
2.4 Volume data generation	7
2.5 Volume data visualization	9
2.5.1 Direct volume rendering	10
2.6 Transfer functions	11
2.6.1 One-dimensional transfer functions	12
2.6.2 Multi-dimensional transfer functions	12
3 Selected challenges in volume data exploration	15
3.1 Background	15
3.2 Light transport theory for volumetric data	17
3.2.1 Material scattering	18
3.2.2 Solving the light transport equation	20
3.3 Enhancing salient features using illumination	21
3.3.1 Spherical harmonics encoding	22
3.3.2 Photon mapping	22
3.3.3 Illumination of time-varying volumetric data	23
3.3.4 Summary of challenges in volumetric illumination	23
3.4 Data exploration using the transfer function	24
3.4.1 Feature separation	24
3.4.2 Feature reconstruction	25
3.4.3 Interaction complexity	26

3.4.4	Summary of challenges in transfer function interaction	27
4	Enhancing and exploring salient features (contributions)	29
4.1	Overview	29
4.2	Unconstrained volume illumination	30
4.2.1	Dynamic light sources	30
4.2.2	Dynamic transfer function	34
4.2.3	Dynamic volume data	36
4.3	Reconstructing what the user wants	41
4.3.1	Feature-based reconstruction	41
4.3.2	Dealing with the partial volume effect	42
4.4	Volume data exploration for novice users	44
4.4.1	Spatial connection for intuitive understanding	44
4.4.2	Improving intuitiveness	45
5	Concluding remarks	49
5.1	Interactive volumetric illumination	49
5.2	Volumetric data exploration	50
5.3	Outlook	51
	Bibliography	55
	Paper A	65
	Paper B	95
	Paper C	111
	Paper D	123
	Paper E	137
	Paper F	149

Motivation

A human is trained from the moment it is born to interpret the world through the visual sense. We are therefore remarkably good at forming a mental image of the information that is fed through our eyes. Examples from the Gestalt principles in Figure 1.1 demonstrate how quickly and easy we understand information through the visual sense [1]. We can identify groups depending on how close features are to each other and fill in missing information without consciously thinking about it [74]. Providing visual representations of complex information is therefore increasingly important to allow humans to understand and reason about the large amounts of information available today.

Figure 1.1: Examples of human interpretation capabilities given by the Gestalt principles. (a) Objects perceived close to each other form groups. (b) Objects are perceived as whole even though they are incomplete.

Figure 1.2: We can quickly understand the contents of a single image (a). But it takes more time to understand the relation between images when the number of images grows (b). By displaying the images in a different way (c), or adding additional information, it becomes easier to spot the relation between the images.

1.1 Visual analysis of complex information

Information can become complex due to a range of reasons. For example, the information can consist of many different parameters or it can have intricate interconnections. It can also become complex due to the amount of information. Take the example of visualizing the light information projected onto a camera at a single instance in time. It is an easy task for a human to understand the contents of the single image captured by the camera. With an increasing number of images it becomes hard and time-consuming to grasp the content of all the images. The information is still the same, only the amount has changed. The information has, in a sense, become complex.

It may be possible to assist the human, i.e. the user, to understand the contents of the images if the connection between them can be emphasized. The images in Figure 1.2, taken by the author, illustrate this concept. Looking at the information from a different point of view, or emphasizing salient features, can help us understand that the images in the figure are about a trip in South America. We may even draw the conclusion that it is likely that the author of this thesis likes to travel. These types of problems are hard for a computer to solve while humans can solve them quickly without much effort. While this may change with future developments in machine learning and artificial intelligence, it has been one of the reasons why humans often are involved in analyzing complex information and why

visualization can help us understand it.

1.2 Enhancing salient features

Humans rely on illumination to better understand their surroundings. The location or shape of an object can be understood by interpreting shadows and specular highlights [13]. Light obstructed by another object can help us to understand if the object is in front of or behind the other object [4], commonly referred to as occlusion. The colors of objects can help us instantly separate or group them as shown in the Gestalt principles [72]. Salient features can therefore be enhanced by assigning different colors to them, applying specular highlights and allowing them to cast shadows. These concepts have been applied by artists to bring out features in their paintings for many centuries.

1.3 Visualizing volume data

This work deals with visualizing volumes of information. The volume can be seen as series of images put into a stack and are thus spatially connected and complex in a sense. The information is often physical measurements from the real world such as the radiodensity in the human body.

Humans are not particularly good at understanding shapes or relations between organs when analyzing millions of radiodensity measurements in a sheet. Instead, we combine the humans' phenomenal visual analysis capabilities of complex information and understanding of light interaction by visualizing the volume data as light interacting media in three spatial dimensions. This allows us to see the relations between the radiodensity measurements, i.e. organs, in the same way as they are arranged in the measured human body. A physician can thereby examine the inside of the human body as if looking at it in the real world.

This thesis uses the light interacting media representation for volume data and presents methods for enhancing salient features through simulated illumination as well as interaction methods for specifying occlusion and color. These methods support the users' in understanding and analyzing volume data using visual analysis.

Introduction

The visualization process often begins with two components, the data and the user. The type of data decides what kinds of visualizations can be offered, while the user provides knowledge about the salient features desired to be discovered and analyzed. This work deals with enhancing salient features when visualizing volume data. A brief introduction on volume data, how it can be generated, and what kinds of visualizations that can be created using this type of data, is provided in this chapter. The focus of this work is on visualizing volume data using the volume rendering equation. The basic model for volume rendering is therefore given in this chapter as an introduction, while an extended explanation is given in the next chapter along with common ways of solving the equation. Paper [A](#) can be seen as a more extensive introduction to the methods used to solve the volume rendering equation. The final section within this chapter explains how users can encode their knowledge to find the salient features within the data.

The interested reader is referred to books on the topic, such as *Visual Computing for Medicine* by Preim and Botha [\[57\]](#) or *Real-Time Volume Graphics* by Engel et al. [\[11\]](#).

2.1 The visualization pipeline

The workflow of most visualization techniques follows the visualization pipeline shown in Figure [2.1](#). Raw data is typically retrieved using sensors or generated by a simulation. Examples of sensors used in a medical setting are Computed Tomography (CT) scanners, which use X-rays to measure the radiodensity inside the body, and Magnetic Resonance Imaging (MRI) scanners, which use strong

Figure 2.1: An overview of the visualization pipeline adapted from Haber and McNabb [15]. Raw data is analyzed, filtered, mapped and finally rendered to produce an image on the screen that can be analyzed by the user. The user gains insight and changes parameters to further explore the data, which causes the pipeline to be re-evaluated.

magnetic fields to measure the distribution of hydrogen in the body. Both of these techniques allow physicians to examine the inside of patients without performing surgery and to possibly diagnose the patient’s condition.

The raw data needs to be transformed into a representation that can be used by the visualization algorithm. This is done in the data analysis step. The data analysis may be a computationally expensive task which derives new features from raw data, or a more light-weight procedure performed during exploration, such as reconstructing an estimate of the original signal from discrete raw data. The process of measuring the signal is therefore closely connected to the data analysis, which consequently needs to reflect the filters used to construct the raw data in order to accurately reconstruct the original signal.

The exploratory phase of the visualization often begins at the filtering and mapping stages, where the interesting parts of the signal are retained and transformed into visual representations. The user can have control over the filter such that features within the data that are currently not important can be removed. Features of interest can, on the other hand, be separated by, for example, mapping them to different colors. The filtering and mapping steps are therefore key components within the visualization pipeline.

The last stage within the pipeline turns the mapped data into pixels on the screen. The pipeline is part of an iterative process, which is re-executed as soon as the user changes the parameters. The faster an iteration can be performed, the faster it is to gain insights, which is one of the reasons why interactivity is essential during the exploratory phase within the visualization pipeline.

2.2 Static volume data

The raw data in the visualization pipeline is, in our case, a 3D array of scalar values. We refer to the 3D array of scalar values as volume data, which can be seen as a stack of 2D images as exemplified on the left in Figure 2.2. Static volume

Figure 2.2: Structured 3D data can be seen as a stack of images put together to form a volume of data (left). Visualization of the stack of computed tomography images to the left (right).

data reflects the fact that the volume data is recorded at a single point in time and thereby represent a static snapshot of the measured objects. The scalar values are organized in a structured grid, which means that the spatial location of each value is implicitly given by its position in the array. The structured grid uses a fixed distance between the values in each dimension such that the position $\mathbf{x}$ of element integer indices (i, j, k) can be computed using $\mathbf{x} = \mathbf{o} + (i \cdot dx, j \cdot dy, k \cdot dz)$, where $\mathbf{o}$ is the origin of the grid and (dx, dy, dz) is the distance between samples in each dimension. The structured grid is the standard format within the medical domain.

2.3 Time-varying volume data

Data measured over time is commonly referred to as time-varying data. In this work, we refer to data recorded in 3D over time as time-varying volume data. It is a 4D signal, where the fourth dimension corresponds to time.

By analyzing volume data over time, it becomes possible to examine and investigate the function of objects within the data in addition to their spatial location, size and shape. This means that the function of the human organs can be examined, enabling a new set of patient diagnosis in the medical domain. For example, the heart beat cycle can be analyzed to understand if the flow into the ventricles is normal, which is not possible to determine using static volume data.

2.4 Volume data generation

There are a range of devices available to generate both static and time-varying volume data. Computed tomography (CT) scanners measure the attenuation of x-rays within the subject, called radiodensity. The radiodensity values produced by the scanner are mapped to Hounsfield units (HU). The medical field commonly

Figure 2.3: MRI scan of a brain visualized in brown together with the fMRI signal in yellow-orange. The MRI scan supplies spatial context for the low resolution fMRI signal, which allows us to visualize where the active areas are in the brain when performing a particular task.

use a HU scale where -1000 corresponds to air, 0 to distilled water and 3000 to compact bone. CT produces best results for hard tissues. Contrast agents with high density can be injected into the blood to highlight blood vessel structures. An example of a CT scan of a human head, where a contrast agent has been injected, is shown on the right side in Figure 2.2.

Magnetic resonance imaging (MRI) scanners measure the radio frequency emitted by hydrogen atoms. First, a strong magnetic field is applied to line up the hydrogen atoms in the same direction in the body. Second, a radio pulse is emitted to excite the nuclear spin of the hydrogen atoms. The atoms will realign when turning off the radio pulse. The realigning will emit a radio frequency which is used to locate the hydrogen in the body. Different tissues are identified by observing the time it takes for the atoms to realign. Soft tissues are better captured by MRI since they commonly contain hydrogen atoms and MRI is therefore used to examine, for example, the structures within the brain [9]. The output values of the same material varies between MRI scans. The values from MRI scans can, in contrast to CT, therefore not be accurately mapped to a specific material [9]. Functional MRI (fMRI) measures blood flow and can be used to examine the activity in the brain under the assumption that increased blood flow indicates higher brain activity. The fMRI scans are of lower resolution than the MRI scans but can be combined with an MRI scan to improve the spatial context of the signal as shown by Nguyen et al. [52]. An example of a fusion of fMRI and MRI scans is depicted in Figure 2.3.

Positron emission tomography (PET) measures metabolic processes in the body. Similar to fMRI, PET scans are generally of low resolution and are therefore often combined with CT or MRI scans in order to provide a spatial reference for the PET signal [71]. PET is commonly used together with a tracer that is injected into the body, which yield a high signal response in areas which the tracer flows to.

Figure 2.4: Time-varying volumetric ultrasound scan of a fetus. The head of the fetus can be seen in the top part of the images. Ultrasound generally produces noisy data, which makes it more difficult to discern features. Data courtesy of Context Vision AB.

Ultrasound is used to measure bones or soft tissues and is often used for point-of-care testing, i.e. diagnosis at the same time and place as the patient is being treated. The time it takes for the ultrasound waves to be reflected are measured and used as a basis to produce an image or volume. The ultrasound measurements cannot be used to classify a given signal value into a tissue but it is possible to analyze the structure of the tissues. Ultrasound is inexpensive but generates noisier data compared to the previously mentioned techniques.

Techniques generating time-varying volume data have been developing rapidly during the last decade. There were no commercial CT-scanners available to extract time-varying data until recently and scanning resolution has increased while the X-ray dosage has decreased at the same time. This means that 4D-CT volume data is starting to be used in the clinical workflow. Other techniques, such as ultrasound, have been used to analyze time-dependent 2D data. However, it has not been possible to produce time-varying volume data using ultrasound in hospitals until recently. An example of a rendering of time-varying volume ultrasound data can be seen in Figure 2.4.

2.5 Volume data visualization

Some of the most common visualization techniques for the data described above are iso-surfacing, maximum intensity projection (MIP) and direct volume rendering (DVR). In Figure 2.5, these three techniques are applied to a CT scan of the shoulder and head of a human. The iso-surface technique allows the user to see where in the data set a given value exists and is therefore useful when an exact value is of interest to the user. The MIP technique allows the user to find the largest values within the data by projecting them onto the screen, which generates a result similar to traditional X-ray images. The MIP technique is useful when large density values are of interest, such as bone in a CT scan. However, spatial relations between objects are lost due to lack of occlusion but they can partially be regained by changing the point of view. Direct volume rendering uses light as an information carrier by simulating the light transport within volumetric data.

Figure 2.5: Examples of three different volume visualization techniques applied to a CT scan of the head and shoulders. (a) Iso-surface algorithm visualizing a specific data value. (b) Maximum intensity projection (MIP) visualizing the maximum data value on the way to the camera. (c) Emission-absorption model interpreting the data values as light which is emitted and absorbed toward on the way to the camera.

The visualization is controlled by filtering and mapping the scalar data into visual properties such as light emission and absorption. While iso-surfaceing and MIP perform binary decisions in what to visualize, the volume rendering technique captures uncertainty through the occlusion resulting from absorption. Enhancing the features within the data using direct volume rendering is the main focus of this work. The following subsection provides an introduction to the key concepts of volume rendering.

2.5.1 Direct volume rendering

The most basic way of applying direct volume rendering is to assume that light is only emitted and absorbed by small particles in the volume. A camera is used to denote the position $\mathbf{x}_c$ and direction $\mathbf{a}$ a user is viewing the scene. The light traveling towards the camera and ending up on the screen forms the visualization. Here, we will give a short introduction to the emission-absorption model, which is a simplified version of the full light transport model explained in Section 3.2.

The amount of light reaching the camera can be computed by integrating the radiance L , the amount of energy flowing per unit area, in the direction towards the camera. The radiance reaching the camera from the initial position $\mathbf{x}_0$ in the direction $\vec{\omega} = \frac{\mathbf{x}_c - \mathbf{x}_0}{\|\mathbf{x}_c - \mathbf{x}_0\|}$, is given by integrating the effects of attenuation and emitted light along the ray:

$$L(\mathbf{x}_c, \vec{\omega}) = T(\mathbf{x}_0, \mathbf{x}_c) \underbrace{L_0(\mathbf{x}_0, \vec{\omega})}_{\text{background}} + \int_0^D \underbrace{T(\mathbf{x}_c, \mathbf{x})}_{\text{attenuation}} \underbrace{\sigma_a(\mathbf{x}) L_e(\mathbf{x}, \vec{\omega})}_{\text{emission}} ds, \quad (2.1)$$

where $D = \|\mathbf{x}_c - \mathbf{x}_0\|$ is the length of the ray and $\mathbf{x} = \mathbf{x}_0 + s\vec{\omega}$, $s \in [0, D]$, is the position along the ray. The first term describes the amount of light reaching the camera from the initial background position. The second part accounts for the attenuated emitted radiance $L_e(\mathbf{x}, \vec{\omega})$ along the ray. The absorption coefficient $\sigma_a(\mathbf{x})$ specifies the amount of particles that are emitting and absorbing light. Attenuation of energy between two points $\mathbf{x}_1$ and $\mathbf{x}_2$ is given by the exponential of the extinction $\sigma_t(\mathbf{x})$ between them:

$$T(\mathbf{x}_1, \mathbf{x}_2) = e^{-\int_0^{\|\mathbf{x}_2 - \mathbf{x}_1\|} \sigma_t(\mathbf{x}_1 + s\vec{\omega}) ds}, \quad (2.2)$$

where $\vec{\omega} = \frac{\mathbf{x}_2 - \mathbf{x}_1}{\|\mathbf{x}_2 - \mathbf{x}_1\|}$ is the normalized direction between the two points and $\sigma_t(\mathbf{x}_1 + s\vec{\omega})$ describes the amount of particles absorbing light at the given position. The attenuation of energy, also referred to as transmittance or transparency, is key to describing the amount of light finally reaching a location and is discussed further in Section 3.3.

2.6 Transfer functions

One of the core components of the volume rendering visualization technique is the transfer function. The transfer function enables users to encode their domain knowledge of the data by mapping the data into optical properties. The user can thereby decide which data that is of interest and how it should appear. The mapping of data can also be referred to as classification. A common approach is to let the user specify the optical properties in terms of opacity $\alpha \in [0, 1]$ and color [8]. The opacity is related to the transparency T according to $\alpha = 1 - T$. The color is usually inserted into Equation 2.1 as the emitted radiance. An opacity of zero will make a feature completely transparent while an opacity of one will make a feature entirely opaque. The opacity of a data value is in this case implicitly representing a constant extinction coefficient for a given length l :

$$\alpha = 1 - e^{-\int_0^l \sigma_t(s) ds} = 1 - e^{-\sigma_t l}. \quad (2.3)$$

The extinction coefficient can be derived from the equation above by taking the logarithm of the two sides and solving for $\sigma_t = \log(1 - \alpha)/l$. Note that an opacity of one will translate to an infinite extinction coefficient, which needs to be dealt with when solving the equations numerically. However, many algorithms use the opacity directly and adapt it to the desired length l' using opacity correction:

$$\alpha' = 1 - (1 - \alpha)^{\frac{l'}{l}}. \quad (2.4)$$

A selection of methods used to specify the optical properties are explained in the following subsections while a thorough overview on transfer functions is given in the work of Ljung et al. [46].

Figure 2.6: Transfer function editors for material classification. (Left) Intensity values are classified into a color and opacity. (Right) Intensity and gradient magnitude can also be combined to classify a material. The intensity-gradient magnitude histogram is shown in the background.

2.6.1 One-dimensional transfer functions

The most commonly used method for exploring volume data is the one-dimensional transfer function. It is simple to implement and does not require any additional attributes to be computed. A scalar value is mapped to optical properties by applying the transfer function. The user can explore the data by altering the function through a graphical interface. A common graphical interface for specifying color and opacity for different scalar values can be seen in Figure 2.6, left. The height of the curve indicates which opacity a given intensity value will be mapped to. The color for a particular intensity value is given by interpolating between the colors of the neighboring handles. The user can adjust the curve and colors to separate or hide features within the data.

2.6.2 Multi-dimensional transfer functions

The scalar based classification is only capable of classifying the material correctly if there is a one-to-one mapping between the scalar value and the material. However, the same value at different locations in the data can, in many cases, reside from different objects. Different attributes, often based on statistical properties, can be added to improve the classification. Note that the output of the multi-dimensional transfer functions are the same as for the one-dimensional ones, i.e. optical properties such as opacity and a color.

Transfer functions based on additional attributes require different user interfaces to deal with the higher dimensionality. A widget-based interface for manipulating the two-dimensional intensity-gradient magnitude transfer function, introduced by Kniss et al. [33], is depicted in Figure 2.6, right. The opacity is encoded as a third dimension, which in this case is mapped to the transparency of the widgets shown in yellow and blue.

This concludes the introduction of core components in volume visualization. This chapter started with describing the visualization pipeline and its application to

volume data. What volume data is and how it can be generated was introduced along with basic techniques to visualize it. We dived into the mathematical theory behind the emission-absorption model to gain a better understanding of the DVR technique, which is the main focus of this work. Finally, it was described how salient features in the data can be shown using the transfer function. The following parts of the thesis will be directed towards challenges addressed and solutions supplied in the included papers.

Selected challenges in volume data exploration

This chapter serves as an introduction to the research contributions presented in this thesis and also presents the corresponding challenges in the volume visualization field with respect to illumination and data exploration. The chapter begins with a general background followed by sections on theory and addressed challenges in the field.

3.1 Background

Sensors producing volumetric data in the medical domain have, during the last decades, substantially increased their output in terms of resolution and quality. The increased resolution and quality enable detailed structures, such as blood vessels, to be examined and diagnosed with greater certainty. Screen resolution has also increased in parallel with the data resolution. This increases the quality of the visualization, but also increases computational complexity and need for performance. The advent of specialized graphics hardware, the graphics processing unit (GPU), has been one of the enabling factors for dealing with higher screen resolution and the increasing amount of data. However, current computation capabilities of the GPU are still not enough to apply brute force methods for visualizing volumetric data of increasing resolution. While each computation performed on a GPU may be slower than on a central processing unit (CPU), the GPU can reduce the total computation time by performing many operations in parallel. However, algorithms and data structures need to be designed specifically for GPUs in order to fully utilize

Figure 3.1: The spatial perception in DVR (a) can be enhanced by applying local shading (b) and global illumination (c).

their parallel nature. Almost all new techniques dealing with volume visualization need to utilize GPUs in order to become interactive and therefore need to be designed accordingly.

The DVR technique allows volume data to be examined using a range of exploration techniques. The point of view can be changed to view the data from different angles and positions. Regions can be clipped to remove obstructing objects. Clipping of exterior parts is often used to explore the interior regions of the data. The transfer function can be changed to reveal or hide features within the data as well as separate features by assigning different colors to them. The position of light sources can be changed to emphasize structures and reveal spatial differences between features. All of these techniques are used to gain an understanding of the data and need to be combined and performed interactively to enable a smooth exploration experience for the user. Each technique presents a challenge in itself. More specifically, an illumination algorithm needs to support clipping and should be fast enough to handle camera and transfer function changes interactively. The perceptual benefit of applying illumination is restricted by the quality, type, placement or number of light sources that can be used, or the material properties which can be applied [66, 68, 76]. The perceptual difference when applying three different illumination approaches is demonstrated in Figure 3.1, where the vessel structure inside a head is visualized. Achieving interactive high quality global illumination that is not restricted by the number or type of light sources is challenging due to the computationally expensive operations required. Another challenge lies in making the transfer function easy to use while improving the material separation. Volume exploration will never reach a wide audience unless all of these aspects are combined with an intuitive interface. The work presented in this thesis is touching all of these aspects and has taken them into account during the design of the methods and algorithms within.

Figure 3.2: Light can be absorbed, emitted and scattered on its way towards the eye. Absorption reduces the amount of light while emission increases it as indicated by the thickness of the arrows. Scattering may increase or decrease the light towards the eye depending on the amount of incoming light and the reflectance properties of the material.

3.2 Light transport theory for volumetric data

Light can be described as particles or waves of energy traveling through space. Emitted light travels unobstructed until it hits an object, which can cause it to change its path or transform into another form of energy. The term scattering is used when light changes its path, while the term absorption is used when transforming into another type of energy that cannot be seen by the eye, such as heat. These three types of events, emission, absorption and scattering, are illustrated in Figure 3.2.

The radiance will be attenuated due to absorption and scattering as it travels towards the eye. The probability that light is absorbed or scattered when traveling an infinitesimal distance ds is given by $\sigma_t(\mathbf{x}) ds$, where σ_t is the extinction coefficient at position $\mathbf{x}$. The extinction coefficient describes the material's density of particles that the light may hit and can be divided into the absorption coefficient $\sigma_a(\mathbf{x})$ and scattering coefficient $\sigma_s(\mathbf{x})$, such that $\sigma_t(\mathbf{x}) = \sigma_a(\mathbf{x}) + \sigma_s(\mathbf{x})$. The absorption coefficient determines how much light is absorbed by the material while the scattering coefficient determines how much light is scattered. The absorbing particles are assumed to be emitting light as well.

The emission-absorption model presented in Section 2.5.1 is extended to include scattering in the integration of radiance along the ray:

$$L(\mathbf{x}_c, \vec{\omega}) = T(\mathbf{x}_0, \mathbf{x}_c) \underbrace{L_0(\mathbf{x}_0, \vec{\omega})}_{\text{background}} + \int_0^D \underbrace{T(\mathbf{x}_c, \mathbf{x})}_{\text{attenuation}} \left(\underbrace{\sigma_s(\mathbf{x}) L_i(\mathbf{x}, \vec{\omega})}_{\text{scattering}} + \underbrace{\sigma_a(\mathbf{x}) L_e(\mathbf{x}, \vec{\omega})}_{\text{emission}} \right) ds. \quad (3.1)$$

The second part integrates all radiance scattered and emitted into the direction towards the camera at each point along the ray from the initial position to the camera position. The outgoing radiance at each position in the integration is attenuated by the transmittance from the camera position to the point being evaluated, given by $T(\mathbf{x}_c, \mathbf{x})$. Energy emitted from a material or light source along the ray is captured by the emission term $L_e(\mathbf{x}, \vec{\omega})$, while the in-scattered radiance along the ray is captured by $L_i(\mathbf{x}, \vec{\omega})$. An illustration of the components

Figure 3.3: The radiance arriving at the location on the screen is the sum of the attenuated background radiance L_0 and integrated emitted and scattered light along the ray towards the eye. Single scattering accounts for the radiance scattered once from the light source(s) towards the eye, indicated by the large yellow arrow, while multiple scattering considers light that has bounced inside the medium as indicated by the small yellow arrows.

in Equation 3.1 is given in Figure 3.3. The integration can also be performed from the camera position to the initial position by switching the position $\mathbf{x} = \mathbf{x}_0 + s\vec{\omega}$ with $\mathbf{x} = \mathbf{x}_c - s\vec{\omega}$.

The scattered radiance $L_i(\mathbf{x}, \vec{\omega})$ takes the light arriving from all directions into account by integrating over the incoming directions towards the point $\mathbf{x}$ on a sphere $\Omega_{4\pi}$:

$$L_i(\mathbf{x}, \vec{\omega}) = \int_{\Omega_{4\pi}} s(\mathbf{x}, \vec{\omega}', \vec{\omega}) L(\mathbf{x}, \vec{\omega}') d\vec{\omega}', \quad (3.2)$$

where $s(\mathbf{x}, \vec{\omega}', \vec{\omega})$ is the material reflectance function, which models the amount of light coming from direction $\vec{\omega}'$ and is reflected into direction $\vec{\omega}$ at position $\mathbf{x}$ (see Figure 3.4). The combination of Equations 3.1 and 3.2 introduces the recursive nature of a global illumination computation. In order to determine the outgoing radiance at a position $\mathbf{x}$ it is necessary to know how much light is sent from all other points in the scene towards $\mathbf{x}$.

3.2.1 Material scattering

The reflectance function is used to simulate how light interacts with real-world materials ranging from plastics to milk, and therefore has a large impact on the visual appearance of the objects in the scene. It describes how the incident light at a point will reflect and is often referred to as local illumination. The reflectance function can improve our ability to perceive the shape of an object. The function

Figure 3.4: The proportion of incoming light reflected from direction $\vec{\omega}'$ into direction $\vec{\omega}$ at location $\mathbf{x}$ is determined by the material reflectance function $s(\mathbf{x}, \vec{\omega}', \vec{\omega})$. (a) A reflectance model that represents a surface material, which depend on the normal $\vec{n}$ orthogonal to the surface. (b) Volumetric material functions are commonly modeled based on the incoming light direction, here using the Henyey-Greenstein phase function ($g = 0.5$).

needs to be energy conserving in order to be physically correct. This means that the amount of incoming light equals the amount of outgoing light:

$$\int_{\Omega_{4\pi}} s(\mathbf{x}, \vec{\omega}', \vec{\omega}) d\vec{\omega}' = 1.$$

The models describing light reflection on a surface, where the incoming and outgoing light direction can be exchanged without changing the result, i.e. $s(\mathbf{x}, \vec{\omega}', \vec{\omega}) = s(\mathbf{x}, \vec{\omega}, \vec{\omega}')$, are called *bidirectional reflectance distribution functions* (BRDFs). These surface based models assume that the light is reflected on a hemisphere centered around the normal axis of the surface as seen in Figure 3.4(a).

Scattering models for participating media generally do not depend on the surface normal. Instead, they model how the light scatters based on the incoming light direction. These functions, describing the scattering of incoming light into any direction on the sphere, are referred to as phase functions. The most straight forward phase function is the isotropic phase function, which scatters light equally in all directions, i.e. $s(\mathbf{x}, \vec{\omega}', \vec{\omega}) = \frac{1}{4\pi}$. Scattering in anisotropic materials, on the other hand, can be approximated by the Henyey-Greenstein [18] model depending on the anisotropy parameter $g \in [-1, 1]$:

$$G(\vec{\omega}', \vec{\omega}, g) = \frac{1 - g^2}{4\pi(1 + g^2 - 2g \vec{\omega}' \cdot \vec{\omega})^{\frac{3}{2}}} \quad (3.3)$$

This equation models the probability that light scatters from direction $\vec{\omega}'$ into direction $\vec{\omega}$, where $g = -1$ means that the light will scatter back into the direction it came from, $g = 0$ results in anisotropic scattering, while $g = 1$ causes the light to continue in the same direction it came from. The Henyey-Greenstein phase function is illustrated in Figure 3.4(b) using $g = 0.5$.

3.2.2 Solving the light transport equation

Light transport equations are generally too complex and computationally expensive to be solved analytically for all but the simplest cases. Interactive volume rendering methods commonly use a numerical solver based on a Riemann sum. The derivation for the emission-absorption model is given here, while the interested reader is referred to the work of Max [51] for the full model, and Paper A for an extensive overview of methods for solving these equations.

The ray along which the light transport equation is evaluated is separated into $i \in [0, N-1]$ intervals over the length D of the ray. Each segment in Equation 3.1, having a length of $\Delta s = D/N$, is assumed to have a constant emission and absorption. The attenuation from the start to the end of the ray can, under this assumption, be approximated using the exponential of the Riemann sum:

$$e^{-\int_0^D \sigma_t(\mathbf{x}_0+s\vec{\omega})ds} \approx e^{-\sum_{i=0}^{N-1} \sigma_t(\mathbf{x}_0+i\Delta s\vec{\omega})\Delta s} = \prod_{i=0}^{N-1} t_i, \quad (3.4)$$

where $t_i = e^{-\sigma_t(\mathbf{x}_0+i\Delta s\vec{\omega})\Delta s}$ and $\prod_{i=k}^l t_i$ refers to the multiplication of all terms t_i (for $k \leq i \leq l$). Similarly, the emission for segment i is defined as $g_i = \sigma_a(\mathbf{x}_0 + i\Delta s\vec{\omega})L_e(\mathbf{x}_0 + i\Delta s\vec{\omega}, \vec{\omega})\Delta s$. The attenuation of the emission from segment i along the ray to the end is approximated using $e^{-\sum_{j=i+1}^{N-1} \sigma_t(\mathbf{x}_0+j\Delta s\vec{\omega})\Delta s} = \prod_{j=i+1}^{N-1} t_j$. The emission-absorption parts in the volume rendering integral in Equation 3.1, excluding background light, can thus be approximated using:

$$\int_0^D T(\mathbf{x}_c, \mathbf{x})\sigma_a(\mathbf{x})L_e(\mathbf{x}, \vec{\omega})ds \approx \sum_{i=0}^{N-1} g_i \prod_{j=i+1}^{N-1} t_j. \quad (3.5)$$

The equation above can be solved with the initial conditions for the attenuation $T = 1$ and radiance $L = 0$ by iteratively applying the front-to-back compositing algorithm for segments $i = N-1$ to $i = 0$:

$$\begin{aligned} L &= T \cdot L + g_i \\ T &= T \cdot t_i \end{aligned} \quad (3.6)$$

Popular technical realizations using the compositing scheme are texture-slicing [5] and ray-casting [39]. In texture-slicing, the iterations along all rays are synchronized through a plane sweeping the volume data, and the intermediate results between each iteration are stored in textures. In ray-casting, each ray sent from the camera is treated independently.

Light scattering can be added by extending g_i to take the incoming radiance in Equation 3.2 into account. As previously mentioned, this turns the expression into a recursive equation that is not easily solved and has been an active research topic during the last two decades. The methods addressing this challenge are further discussed in Section 3.3.

3.3 Enhancing salient features using illumination

One of the main goals within the volumetric illumination field has been to come up with techniques that can handle advanced illumination effects with high quality, have a small memory footprint and are fast enough to compute without losing interactivity. Salient features are specified by the user through the transfer function and are therefore a key aspect to consider when applying volumetric illumination to enhance them. Producing high quality illumination to enhance salient features is a challenging goal and each technique presented during the last two decades has made a trade-off with respect to one or more of the goals in order to reach interactive frame-rates.

Some techniques are tailored towards a specific rendering paradigm for volume data. The half-angle slicing technique presented by Kniss et al. [34] has a low memory overhead and produces advanced illumination effects but is bound to the texture slicing rendering technique [5]. Texture slicing has the advantage that it works on old hardware but has been shown to produce lower quality than ray-casting [65].

Many illumination algorithms use fast pre-computation techniques of illumination information which often allow faster rendering speeds for camera movements at the expense of additional memory requirements [2, 19, 58, 59, 60, 75]. One example is the summed area table-based technique presented by Schlegel et al. [63]. They precompute a summed area table which is co-registered with the volume data and use it to improve the performance of estimating the attenuation of light during rendering. By storing the summed area table they are able to reuse computations that would need to be performed each time the incident light is evaluated at a point in the volume. Most techniques that rely on pre-computed illumination information need to perform a considerable amount of computations as soon as the light sources change, which may prevent interactive dynamic lights from being used. One way to circumvent this problem is to decouple light information from attenuation. Spherical harmonics can be used to encode the attenuation from all directions [41, 64], which means that the lights can be changed without recomputing the attenuation.

A recent trend has been to enhance the methods traditionally used for offline computation within the computer graphics field to work for interactive volume data exploration purposes. Techniques used for offline computation are able to produce high quality illumination and have less restrictions on the type of illumination effects that can be achieved. Kroes et al. [36] presented a framework adapted to use the GPU for path-tracing of volume data. They are able to produce high quality direct illumination at the expense of slow frame-rates, i.e. extensive computation. Advanced effects such as multiple scattering can be solved more efficiently using photon mapping. Zhang et al. [77] presented a technique that combines photon mapping with a spherical basis function to pre-compute and store the outgoing radiance in a grid. While their technique allows interactive exploration using

the camera, changing the transfer function or light source still require minutes of computation time.

The following sections detail the challenges in using the spherical harmonics and photon mapping techniques. A thorough overview of the techniques within the interactive volumetric illumination field and its challenges are presented in Paper A.

3.3.1 Spherical harmonics encoding

The spherical harmonic basis representation is a set of orthonormal basis functions defined on a sphere. The underlying idea resembles a Fourier series, which represent functions on a circle instead of a sphere using the parametrization $(x, y, z) = (\sin \theta \cos \varphi, \sin \theta \sin \varphi, \cos \theta)$. One of the benefits of the spherical harmonic basis functions is that they can be rotated efficiently due to the sparse connections created by the orthonormal basis [10]. The basis functions $Y_l^m(\theta, \varphi)$ are ordered from low to high angular frequency, broken into bands where $l \geq 0$ is the band index and $m \in [-l, l]$ is the integer range of the band. The representation can in theory describe all square-integrable functions ($\int_{-\infty}^{\infty} |f(x)|^2 dx < \infty$) and it is therefore possible to approximate a function $f(\theta, \varphi)$ on the sphere using the representation. The real-valued approximation $\tilde{f}(\theta, \varphi)$ can be formed by expanding the orthonormal basis into a linear combination of the basis functions:

$$\tilde{f}(\theta, \varphi) = \sum_{l=0}^{n_l-1} \sum_{m=-l}^l f_{l,m} Y_l^m(\theta, \varphi), \quad (3.7)$$

where $n_l \geq 0$ is the degree of the basis functions and $f_{l,m}$ are the projected spherical harmonic coefficients. The projected spherical harmonic coefficients are found by integrating the product of the function and the spherical harmonic basis:

$$f_{l,m} = \int_{S^2} f(\theta, \varphi) Y_l^m(\theta, \varphi). \quad (3.8)$$

The number of coefficients used, i.e. the number of bands, is one of the key aspects to consider when using the spherical harmonic representation. Higher frequency effects can be reproduced when using more coefficients, but it requires more memory and computation. A trade-off between performance, quality and memory needs to be made for practical purposes. One of the main challenges in using the spherical harmonic representation is therefore to reduce the amount of required memory and computation while retaining the quality of the visualization.

3.3.2 Photon mapping

The radiance L arriving at a differential direction $d\vec{\omega}$ is given by the flux $\Phi(\mathbf{x})$, i.e. the amount of energy (light), over a finite volume dV :

$$L(\mathbf{x}, \vec{\omega}) = \frac{d^2 \Phi(\mathbf{x})}{d\vec{\omega} dV}. \quad (3.9)$$

The photon mapping illumination technique, presented by Jensen [22], tries to approximate the radiance using photons representing the flux. The technique is performed in two steps. Photons are first traced from the light sources into the scene until they are absorbed or scattered. The position $\mathbf{x}_p$ at which an absorption or scattering event occurs is stored along with the direction $\vec{\omega}_p$ and energy of the photon $\Phi_p(\mathbf{x}_p, \vec{\omega}_p)$. Scattered photons may be traced again until a maximum number of scattering events has occurred or using a Russian roulette approach [22]. Together, all photons form a distribution of the flow of light energy, flux, which is referred to as a photon map. An estimate of the radiance in Equation 3.9 can be combined with Equation 3.2 to approximate the incoming radiance by gathering the flux of all n photons within a sphere of radius r :

$$L_i(\mathbf{x}, \vec{\omega}) \approx \sum_{p=1}^n s(\mathbf{x}, \vec{\omega}_p, \vec{\omega}) \frac{\Phi_p(\mathbf{x}_p, \vec{\omega}_p)}{\frac{4}{3}\pi r^3} W(\|\mathbf{x} - \mathbf{x}_p\|/r), \quad (3.10)$$

where dV has been substituted with the volume of the sphere and $W(\|\mathbf{x} - \mathbf{x}_p\|/r)$ is a normalized spatial smoothing kernel. The equation above allows complex illumination effects, such as multiple scattering, to be taken into account and solved efficiently due to the reuse of photon calculations for multiple view rays. Also, due to the two-step process of photon tracing and photon gathering it is not necessary to recompute the photons when changing the view direction. However, the photon mapping technique is still too computationally expensive for high quality illumination in interactive settings. The challenge therefore lies in finding ways of evaluating the two steps efficiently without reducing quality.

3.3.3 Illumination of time-varying volumetric data

Most of the techniques within the volumetric illumination field have been designed for static volume data. This has allowed them to take advantage of different aspects of the volume rendering equation under the assumption that the data is not changing. For example, high quality illumination can be achieved by applying iterative techniques that improve the quality over time [36]. However, this is not possible for time-varying data as the scene is constantly changing and therefore resets the computation each frame. Time-varying data exhibit a similar behavior to changing the camera, light source and transfer function at the same time. It is therefore particularly challenging for techniques trying to solve the rendering equation, since all of the parts in the rendering equation need to be solved at the same time.

3.3.4 Summary of challenges in volumetric illumination

Previous techniques for interactive volumetric illumination have used various simplifications of the volume rendering equation, cf. Equation 3.1 on page 17,

which has allowed them to yield interactive frame rates. Unfortunately, these simplifications put constraints on the number of light sources, types of light sources or supported scattering functions. The techniques which can produce complex illumination effects, such as multiple scattering, do so using crude approximations or the mentioned simplifications. A challenge within the volumetric illumination field is therefore *to enable high quality interactive illumination without large simplification of the volume rendering equation*. Time-varying volume data adds an additional challenge in which all parts of the volume rendering equation need to be solved for each frame.

Challenge: Enable high quality interactive illumination without large simplification of the volume rendering equation.

3.4 Data exploration using the transfer function

The transfer function is an essential tool to be able to explore volume data and there has been an extensive amount of work done on how to improve the ability to extract the desired features and on how to make it easier to use. The following sections discuss challenges in improving classification without increasing the complexity for the user and how to make the transfer function available to users who are not familiar with volumetric data.

3.4.1 Feature separation

Sensors often measure the signal by applying a filter on the source signal. The filtered value is stored and, in our case, forms a voxel value. Multiple materials may have been involved in the filtering which causes a voxel value to represent a mixture of materials. This effect is known as the *partial volume effect*. Classifying a feature based on a signal which does not have a one-to-one mapping between value and feature is difficult by nature. Drebin et al. [7] introduced the transfer function as a probabilistic classification of samples, where they combined visual contributions based on the mixture percentages in each voxel. In general, however, the mixture percentages are not known. There have been three main approaches to addressing the partial volume issue.

The first approach is to *visually classify* the material through the use of additional attributes such as gradient magnitude [33, 40]. This approach puts the power, or burden, on the user to separate materials from each other. Statistical attributes such as moment curves [54] can be also be used to improve the separation of boundaries between materials. Introducing additional attributes to data that already are multivariate, such as dual energy CT, is problematic. The method

needs to deal with the increasing memory requirement and an interface that allows the user to manipulate the transfer function in higher dimensions. Consequently, more complicated interactions and multivariate data prevent these methods from reaching a wider audience. There are also related techniques focusing on deriving more advanced attributes to help the user to separate features. Sato et al. [62] used local structures based on eigenvalues to identify sheets, lines or blobs. Sheets, lines and blobs can roughly correspond to the anatomical structures of skin, vessels and nodules, respectively. Attributes based on object level abstractions, such as size [6] and shape [56], can also be identified and used as additional attributes. Lindholm et al. [42] combined spatial conditions and semantic labels to separate tissues based on expressions such as "show gas close to fat".

The second approach is to use *statistical models* to compute the most probable mixture and location of materials within the voxel as summarized by Kaufman and Mueller [32]. In a sense, the statistical models try to reverse-engineer the partial volume effect and either produce a weight per material or the most likely material per voxel. Producing multiple weights is problematic from a memory consumption perspective while taking the most likely material discards all the information about less likely materials. A challenging scenario with the probabilistic models is that they require a-priori knowledge about all materials. This means that values corresponding to materials that are not known still must be forced to belong to a material included in the model, which may mislead the user.

When existing, a *segmentation* can be combined with the transfer function to constrain the result and thereby remove artifacts. The segmentation itself provides a binary decision on which material the voxel contains. Simply using the segmentation will therefore produce poor results in boundaries between materials. Tiede et al. [70] and Hadwiger et al. [16] have presented methods to combine the transfer function and segmentation to produce smooth boundaries with binary decisions. However, accurate segmentation masks are usually not available and often need to be manually created which is too time-consuming for practical use.

3.4.2 Feature reconstruction

The most common way of reconstructing data for visualization is to assume that the signal is continuous. All values between two neighboring sample points must exist under the continuity assumption. The implication is that all materials represented between those two values also must exist in-between. Even for the cases where all the materials do not exist between two neighboring sample points, i.e. during the scanning procedure, they will nevertheless be reconstructed and displayed to the user. As an example consider a CT scan of a human body. A ray that passes through air and bone, which are far apart in the HU scale, must pass through all the soft tissues defined between air and bone under the continuity assumption. As such, features currently visualized, blood, liver, kidney and so forth, will be

Figure 3.5: A histogram (blue surface) can be added to the transfer function editor to help the user understand the contents of the data set. However, it is still difficult to make a connection between the data values and the final rendering on the screen.

visualized along the ray. Clearly, a ray passing through the head will most likely not contain features such as kidney or liver. A major challenge is to reconstruct the signal such that it can be classified into features that are expected by the user, i.e. not a liver in the head.

3.4.3 Interaction complexity

Most work on transfer functions has been focused on improving feature separability and usability for expert users. The methods can therefore expose detailed or advanced features to allow the users to improve their classification. Novice users have been left with a set of predefined settings designed by experts [48].

One of the most common ways of exploring volume data is to use a transfer function that performs material mapping solely based on intensity values. A problem with this approach is that there is no information about the data to guide the user in the exploration process. It is possible to display additional information to help the user, such as the histogram depicted in Figure 3.5. The histogram allows the user to see how often the values occur within the data set. Other attributes can also be displayed such as local histograms [47] based on spatially varying regions of the data, or isosurface similarity [3]. Still, since an intensity value may occur at different locations in the data it can be hard to predict which feature will change in the visualization when altering the transfer function. Marks et al. [50] introduced design galleries, which allow the user to explore the parameter space using pictures generated based on an analysis of the data. A set of parameters is chosen by selecting the image containing the desired features, which may be more intuitive than moving widgets in the intensity-based transfer function editor. However, only a subset of the high dimensional parameter space can be represented by images so there is a need to navigate the parameter space to retrieve the desired visualization. A semi-automatic two-step approach to navigate this multidimensional space was suggested by Pinto and Freitas [55]. They generated similar images based on the

selected preview and then allowed the user to make detailed changes by manipulating the transfer function directly in a second step. The automatic and semi-automatic approaches require a metric that determines what a user thinks is important, which may be impossible to determine. Pinto and Freitas also change all the color and opacity at the same time, which may not be the intention of the user. König and Gröller [35] presented a tool for medical doctors where previews are correlated with the intensity domain to help the users establish a connection between the data values and the final rendering. They use a three-step approach with detailed parameter editing and previews available at each step. The method starts with initial peak specification, then color specification and finally opacity specification. However, there has not been any study on how well the method performs.

3.4.4 Summary of challenges in transfer function interaction

In order for volume data visualization to reach a wider audience it is necessary to develop techniques and tools that are intuitive and can delineate the features desired by the user. Enabling the user to *improve feature separation without introducing additional complexity* is one of the major challenges when developing transfer function methods. This involves dealing with the partial volume effect as well as signal discontinuity.

Most techniques presented for exploring volume data have been targeted to expert users. Expert users may spend a considerable amount of time learning the tool and have high demands on advanced features required to retrieve the desired result. Novice users, on the other hand, need an easy interface that does not require much explanation. Here, the challenge lies in *enabling novice users to explore volume data without prior knowledge on transfer function editors and little time to learn the interface*. This requires a technique that enables the user to understand the connection between exploration parameters and the final visualization, provides feedback on the interactions such that the user can understand the effect of changing a parameter and is responsive such that the feedback is immediate. All of the above requirements need to be met in order to give the user a feeling of being in control.

Challenge: Improve feature separation without introducing additional complexity and enable novice users to explore volume data without prior knowledge on transfer function editors and little time to learn the interface.

Enhancing and exploring salient features in volume data (contributions)

This chapter presents the contributions of the papers included in this thesis. An overview contextualizing the papers within the visualization pipeline, is first presented followed by a more detailed description of each contribution.

4.1 Overview

The contributions of each paper with respect to the visualization pipeline is shown in Figure 4.1. Paper A presents an overview of the state-of-the-art within the field of volumetric illumination and classifies the listed techniques with respect to their support for illumination effects, computation speed when changing the camera, light setup or transfer function as well as memory consumption. Papers B–D present new methods to improve volume illumination using spherical harmonics and photon mapping. The trade-off with respect to memory requirements, speed and quality are evaluated for each of the presented methods. Paper E combines the reconstruction and mapping stages within the visualization pipeline, enabling domain knowledge to be encoded and desired features to be displayed. Paper F introduces a dynamic gallery approach to make volume exploration more accessible to novice users. This dynamic gallery allows the user to interact with the filtering and mapping stages in a more intuitive way compared to the standard widget-based transfer function editor.

Figure 4.1: Overview of the contributions of the papers included in this thesis with respect to the visualization pipeline applied to volume visualization. Papers **A–D** utilize knowledge from the mapping stage, i.e. the transfer function, to improve the results of the rendering stage. Papers **E** and **F** enhance the filtering and mapping stages by improved data reconstruction and transfer function interaction. All the presented techniques leverage user knowledge to improve the quality or performance of the method.

4.2 Unconstrained volume illumination

The possibilities of enhancing perception in volume visualization is limited by the types of illumination effects supported by the illumination algorithm. This section presents three methods for interactive volumetric illumination, where the constraints are reduced compared to state-of-the-art methods in the field. The first method addresses the performance and memory impact of changing the light sources in the scene using spherical harmonics. The second method enables the photon mapping technique to be used for interactive exploration of the volume data using the transfer function. The third method further improves the performance of the photon mapping technique, thereby enabling interactive frame-rates when changing the camera or applying the technique to time-varying data.

4.2.1 Dynamic light sources

There are three main parts to consider when solving direct (single scattering) light transport in the rendering equation in terms of computational impact. These three parts can be seen in Figure 3.3 on page 18. The first part involves computing the attenuation from the camera to the point within the volume, i.e. the attenuation in the integral in Equation 3.1. The second part involves computing the integration of light from all directions on the sphere at the evaluated point, cf. Equation 3.2. The third part, connected to the integration of light, involves computing the attenuation from the evaluated point to all light sources. The first part is often solved by traversing the view-ray and incrementally computing the attenuation along the ray as explained in Section 3.2.2. The second and third parts need to be performed at

Figure 4.2: The spherical harmonics basis function weights, representing the visibility at each point, are stored in a flat blocked multiresolution datastructure. The multiresolution datastructure allows computational efforts and memory consumption to be focused on regions with high spatial detail and thereby substantially reduce the memory requirements.

each step along the ray, which is more computationally expensive compared to the first part. Paper B solves this by encoding attenuation and light from all incoming directions using spherical harmonics. As discussed in Section 3.3.1, the use of spherical harmonics is challenging from a memory and performance perspective. These challenges are addressed in Paper B using a multiresolution data structure as explained below.

Spherical harmonics visibility encoding

The visibility $V(\mathbf{x})$, i.e. the attenuation from all incoming directions at position $\mathbf{x}$, is approximated using spherical harmonics basis function of degree $n_l \leq 3$ such that $V(\mathbf{x}) \approx \tilde{V}(\mathbf{x})_{l,m}$. Similarly, the incoming light from all directions at this position is also approximated using spherical harmonics such that $L_i(\mathbf{x}) \approx \tilde{L}(\mathbf{x})_{l,m}$. By using the properties of spherical harmonics it is possible to solve the second part of the equation, the integration of light, using the dot product between the spherical harmonics representation of the visibility and the light sources in the scene:

$$\tilde{L}_i(\mathbf{x}, \vec{\omega}) \approx \frac{1}{4\pi} \sum_{l=0}^{n_l-1} \sum_{m=-l}^l \tilde{V}(\mathbf{x})_{l,m} \tilde{L}(\mathbf{x})_{l,m} = \sum_{i=0}^{n_l^2} \tilde{V}(\mathbf{x})_i \tilde{L}(\mathbf{x})_i. \quad (4.1)$$

Here, the constant $\frac{1}{4\pi}$ comes from the fact that an isotropic phase function is assumed and the index i is simply flattened from indices (l, m) to better show the applicability of the dot product. Note that it is possible to change the light source

Figure 4.3: Efficient visibility computation using spherical harmonics. (a) The local visibility is computed for each point in the multi-resolution data structure by adaptively sampling the spherical neighborhood. (b) The spherical harmonics representation of the local visibility is reused when computing the global visibility to improve the performance.

setup without recomputing the spherical harmonics representation of the visibility since the visibility is independent of the light source information.

Unfortunately, encoding the visibility using spherical harmonics has a substantial impact on memory consumption. Each location storing the visibility requires at least four basis function weights, which means that an overhead of at least four times the original data set is required for a spatial resolution equivalent to the volume data. Paper B addresses the memory challenge by using a flat blocked multiresolution datastructure [45] to store the spherical harmonics weights as illustrated in Figure 4.2. The use of a multiresolution datastructure allows the spatial resolution of the visibility information to vary, increasing where high detail is needed and decreasing where the data is constant or not visible, according to the given memory budget as described in the work by Ljung et al. [45]. The multiresolution datastructure is also used to decrease the computation time as described in the following subsection.

Visibility computation

The visibility computation was inspired by the two-step integration approach used in the work by Hernell et al. [20]. They computed ambient occlusion by first integrating the local occlusion in a direction and then use it to take larger steps when integrating the global occlusion. Our approach uses the same two steps but encode the visibility from all directions using spherical harmonics. The first step, illustrated in Figure 4.3(a), computes the local visibility around each point in the multiresolution grid. The local integration utilizes the level-of-detail given by a

Figure 4.4: Comparison between the use of shading (left) and attenuated illumination computed using with the spherical harmonics visibility encoding (right). The highlights emphasize that spatial structures become more apparent when taking the global visibility into account.

multiresolution datastructure to adaptively sample the data. The computation of the local visibility has two advantages. First, it may be used to incorporate local ambient occlusion in the visualization, which can be beneficial to use when visualizing noisy data [21]. Second, it can be used to speed up the computation of global visibility, which is the second step in the visibility computation. The global visibility is computed by taking steps with lengths equal to the extent of the local visibility as illustrated in Figure 4.3(b). The local visibility can thereby be reused instead of computing the visibility again.

Results and applicability

Spherical harmonics-based illumination can enhance the depth perception compared to local illumination as demonstrated in Figure 4.4. A thorough evaluation of the impact on image quality was performed for varying number of spherical harmonics coefficients, number of rays used to compute the visibility and local visibility radius. The evaluation showed that the visual quality, using the HDR-VDP norm [49], increases linearly with the number of spherical harmonics coefficients used. Roughly 30 different directions should be used for the visibility computation from a quality versus performance point of view. Using too few directions to evaluate the visibility will introduce noisy and distorted shadows. A local visibility radius of about twelve to sixteen voxels was shown to be optimal in terms of computation speed for the global visibility. While changing the camera can be done with rendering speeds of 6-30 frames per second, changing the transfer function require computation times in the order of seconds. The light sources can be changed interactively since it only involves updating their spherical harmonics coefficients. A wide variety of light source types and complex light setups are supported. However, the produced shadows are generally of low frequency since it is not feasible with interactive

performance to use the number of spherical harmonics coefficients necessary to reproduce high frequency changes.

4.2.2 Dynamic transfer function

Although the spherical harmonics method presented in Paper B can handle complex light setups, the generated shadows are of low frequency and it cannot produce illumination effects such as multiple scattering. The photon mapping technique, on the other hand, can produce a wide range of illumination effects with high quality. We would therefore like to use the photon mapping technique for interactive volume rendering. However, as pointed out in Section 3.3.2, the photon mapping technique has been limited to offline rendering due to the amount of computations required. Paper C address the challenge of enabling photon mapping to be used for interactive exploration of volume data using the transfer function. The underpinning idea is to reduce the amount of computation necessary when changing the transfer function by only recomputing the parts of the volume rendering equation affected by the change. Identifying and recomputing the affected parts must be faster than recomputing everything and the memory overhead should be minimized. Following the work of Jensen and Christensen [24], the rendering equation is separated into two parts. One considers the generation of the light distribution, i.e. photon tracing, and the other considers the estimation of incoming light along a view-ray, i.e. photon gathering. Information about both of these parts needs to be stored in order to know if they are affected by a change.

Historygram

A brute-force method to store information about the used parameters would store a list of all the material parameters (color, opacity, scattering properties and so on) occurring at each step in the photon tracing or view-ray calculation. When a parameter changes, for example the color or opacity, one would search for the photons and view-rays that contain the changed parameter and recompute them. However, the memory overhead would be substantial since many steps may be taken along each ray and searching through all used material parameters would not be efficient. Paper C utilizes the fact that all material parameters are dependent on the data values due to the mapping through the transfer function, which means that it is possible to reduce the storage to actual data values without losing any information. While this reduces the amount of data to a list of data values instead of a list of all material parameters, each photon or view-ray may use hundreds of data values, which is still too costly. It is possible to reduce the required information further by utilizing the fact that it is only necessary to know if a value has been used, not how many times, in order to determine if a change affected the photon or view-ray. Paper C propose to uniformly divide the data domain and use a binary

Figure 4.5: The data used by view-ray segments S1-S7 and photon interactions P1-P4 are encoded using a binary representation indicated by the colored boxes. This example uses two bits to represent the data while in practice 32-64 bits are used. The segments and photons to recompute when the user changes the transfer function can be efficiently determined by comparing the used data with the data range affected by the change.

flag to indicate if a data value within the range has been used. The representation of the history of a photon or view-ray is referred to as a historygram.

Each view-ray is divided into a set of segments in order to gain a more fine-grained detection of change. The view-ray segments can be combined during the rendering using the multiplicative property of the attenuation in Equation 2.2. Each photon and view-ray segment store a historygram as seen in Figure 4.5. The binary information is represented using hardware supported formats, typically with 32 or 64 bits, which means that evaluating if a photon or view-ray has changed can be performed using native hardware binary instructions. The historygram data structure is therefore efficient to create and evaluate while requiring a small memory storage overhead. A change of the transfer function is encoded using a query historygram. The query historygram is created by marking the bins representing the changed scalar values. Each photon and view-ray segment historygram is compared to the query historygram to see if they are affected by the change and therefore need to be recomputed. The changed photons are retraced and the changed view-ray segments perform the photon gathering step again. The final image is generated by combining the view-ray segments along each ray.

Photon gathering on the GPU

Photons need to be gathered within a given radius at each step along the view ray according to Equation 3.10. Using a brute force approach would be too computationally expensive to gain interactive performance since the operation needs to be performed at each step along the ray. Finding the nearby photons therefore requires an efficient data structure that is adapted to GPUs. A common

Figure 4.6: High quality illumination of a CT scan of a hand using the photon mapping technique with multiple scattering. About 70 % of the calculations can be reused when changing the material representing the bone.

approach is to use kd-tree for CPU-based photon mapping techniques [23]. However, these types of data structures require data access patterns that generally do not perform well on GPUs. Instead, Paper C uses a chaining hash table for this purpose. The chaining hash table allow photons to be found in $O(1)$ time. Each bin within the hash table represent a uniformly sized box with a size equal to the search radius. The photons within a bin are stored contiguously in memory to improve performance through memory-aligned access. Finding all photons within the radius at a location along the ray can be done by querying the twenty-seven neighboring bins. The performance of the photon mapping gathering process can thus be improved without sacrificing quality.

Results and applicability

The quantitative results show that speedups between 1.3 to 15.2 times can be made when changing the transfer function at the cost of an initialization overhead between 5 to 25 percent. The performance improvement means that interactive frame-rates can be achieved when exploring volume data using the transfer function. High quality visualization can be created using multiple light sources of different types placed anywhere in the scene with effects such as multiple scattering. An example of the visual quality possible with this method is given in Figure 4.6. Further work needs to be done to improve the photon gathering performance to allow interactive camera movements. The method is suitable in applications with static volume data where high quality illumination is desired.

4.2.3 Dynamic volume data

Solving the volume rendering equation for time-varying volume data is challenging since all parts of the equation need to be solved all the time as discussed in Section 3.3.2. Interactive photon mapping can therefore not be achieved with current hardware when applying the method in Paper C to time-varying volume data. Paper D extends the underpinning idea of updating the photons that change

Figure 4.7: The two regions A and B in the time-varying ultrasound data set have different characteristics over time. Region A looks almost the same in the two time steps, while region B has a large visual change. Paper D identifies the regions within the data between time steps that have highest visual change and focus the photon distribution computation to region B in this case.

in Paper C to also work for time-varying data. The challenge in Section 3.3.2 is addressed by identifying and utilizing the correlation between photon distributions to propagate the subset of the photons that change to the new photon distribution. Thus, the method in Paper D enables the photons invariant to change between time-steps in region A in Figure 4.7 to be reused, while prioritizing the photons changing in region B according to their visual importance.

Similar to Paper C, the overhead in terms of computation and memory for identifying the correlation needs to be low, such that the benefit of utilizing the correlation is not lost. The computation and memory overhead can be reduced by minimizing the amount of information necessary to compute the correlation as well as using a low resolution data structure for storage. It was identified that metadata containing the minimum and maximum values of a region can give sufficient information if it is combined with the changes of the region in time and detailed information about the transfer function. A uniform grid of lower resolution, where each cell stores these values, is used to reduce the memory and computation overhead. Reducing the resolution of the grid increases the computation speed but decreases the separability of small features. Each cell in the uniform grid encodes the mean value difference between two time steps $E[|V(t_i) - V(t_{i-1})|]$ as well as the minimum and maximum values, V_{min} and V_{max} , respectively, within the cell. The metadata of a region together with the transfer function are used to compute the visual importance of a change. The visual importance is collected by the photons in the scene and used to prioritize the photon tracing of the changed photons. The most important photons are retraced and sent to the gathering stage, which enable the computations to operate on a subset of the photon distribution instead of the whole one. The described pipeline is illustrated in Figure 4.8. The novel visual importance metric

Figure 4.8: Changes due to time-varying volume data or visual parameters are used to calculate an importance map. The importance map only consider changes that may have a visual impact. The photons are prioritized by collecting the visual importance along their paths. The most important photons are re-traced and then sent to the gathering step along with the corresponding old ones. The final rendering can be computed by removing the contributions from the old photons and adding the contributions from the retraced photons.

and new photon gathering approach in the pipeline are described in more detail below.

Visual importance

The correlation between time steps is found by applying an importance metric that considers visible changes. A change within the data is only visible if the transfer function is varying, i.e. not constant. Paper D introduces a visual importance metric, which considers changes in the data as well as the transfer function. The response of the metric is high in regions where the data changes a lot in combination with a varying transfer function. The change in data is measured as the mean difference between two time steps within a cell $E[|V(t_i) - V(t_{i-1})|]$. The change in the transfer function is evaluated between the minimum and maximum values within the cell, $v \in [V_{min}, V_{max}]$. Both the maximum change of color $\vec{C}(v)$ in the Lab color space and the opacity $\alpha(v)$ are considered when forming the visual importance ψ_V :

$$\psi_V = E[|V(t_i) - V(t_{i-1})|] \left(\max_{v \in [V_{min}, V_{max}]} \left(\left| \frac{d}{dv} \vec{C}(v) \right| \right) + \max_{v \in [V_{min}, V_{max}]} \left(\left| \frac{d}{dv} \alpha(v) \right| \right) \right). \quad (4.2)$$

Here, $C_0 = \|(100, 500, 400)\|$ is a normalization factor for the Lab color and $v \in [V_{min}, V_{max}]$ are the minimum and maximum values within the cell of both time steps. Paper D describes a similar method to compute the visual importance due to a transfer function change. These two metrics are added together to form the final visual importance stored in the grid. Photons passing cells with zero visual importance can be directly propagated to the next time step. The other photons are prioritized according to their accumulated visual importance, traced again and sent to the photon gathering stage.

Approximate photon gathering

The most computationally expensive step in the photon mapping method is the gathering of photons. A ray is cast from each pixel on the screen. At each step along the ray, typically two steps per voxel, the nearby photons need to be collected in order to estimate the incoming radiance in Equation 3.10. While Paper C used a hash data structure to improve the performance of accessing nearby photons, it is still too computationally expensive to perform interactively when changing the camera. Paper D presents an approximation to this step. Equation 3.10 is divided into two parts, one where the density estimation is computed and one where the shading is done. The density estimation is computed and stored at each position $\mathbf{x}_i$ in a grid co-located with the original data set:

$$\Phi(\mathbf{x}_i) \approx \sum_{p=1}^n \frac{\Phi_p(\mathbf{x}_p, \vec{\omega}_p)}{\frac{4}{3}\pi r^3} W(\|\mathbf{x}_i - \mathbf{x}_p\|/r). \quad (4.3)$$

The number of expensive density estimations is thereby reduced to the number of grid-locations instead of the number of steps along all rays. The density estimation along each ray can be reconstructed from the grid using hardware accelerated linear filtering, which is less costly than finding and taking nearby photons into account. Unfortunately, the directional component of the photons is lost using this approximation. The direction is instead approximated by performing the shading for each light source and weighing the shading by the relative power of the light sources.

The density estimation in Equation 4.3 is extended to operate on the photon difference distribution over time, i.e. the subset of photons changing between time steps t_{old} and t_{new} , instead of the whole distribution. The photon distribution $\Phi(\mathbf{x}_i, t_{new})$ at the new time step is computed by removing the changed photons $\Delta\Phi_p(\mathbf{x}_p, \vec{\omega}_p, t_{old})$ and adding the re-traced photons $\Delta\Phi_p(\mathbf{x}_p, \vec{\omega}_p, t_{new})$:

$$\begin{aligned} \Phi(\mathbf{x}_i, t_{new}) \approx & \Phi(\mathbf{x}_i, t_{old}) + \sum_{p=1}^n \frac{\Delta\Phi_p(\mathbf{x}_p, \vec{\omega}_p, t_{new})}{\frac{4}{3}\pi r^3} W(\|\mathbf{x}_i - \mathbf{x}_p\|/r) \\ & - \sum_{p=1}^n \frac{\Delta\Phi_p(\mathbf{x}_p, \vec{\omega}_p, t_{old})}{\frac{4}{3}\pi r^3} W(\|\mathbf{x}_i - \mathbf{x}_p\|/r). \end{aligned} \quad (4.4)$$

Here, $\Phi(\mathbf{x}_i, t_{old})$ is the density estimation computed at the previous time step. The subset of photons needs to contain at most half of the total photon distribution for Equation 4.4 to be beneficial to use.

Results and applicability

The method was evaluated with respect to image quality and performance when limiting the number of updated photons per frame. The perceptual impact of

(a) Visual importance (dotted line) versus equal importance (solid line) with a maximum of 40% of photons updated per frame.

Figure 4.9: Perceptual impact of the visual importance approach applied to an ultrasound data set. The image quality is improved when applying the visual importance based metric to determine which photons are updated (a). Reference image where all photons have been recomputed, equal importance and visual importance at frame 37 in the sequence (c)–(d). (f)–(g) Color mapped SSIM error of (c) and (d).

applying the method to an ultrasound data set is shown in Figure 4.9. A maximum of 40% of the total photons were permitted to be updated at each frame. The visual importance based approach is compared to using a uniform pattern for deciding which photons are updated. The mean structural similarity (MSSIM) [73] was used as error metric. The MSSIM metric ranges from $[-1, 1]$, where -1 is worst and 1 is the same as the reference image. The visual importance based method outperforms the uniform pattern in terms of image quality as seen in Figure 4.9(a). Similar results were shown in Paper D for a CT scan of a beating heart and a fused MRI/fMRI scan of a brain.

The correlated photon mapping method is 3 to 9 times faster without losing image quality compared to retracing and gathering the entire photon distribution at each frame. In addition, the approximate photon gathering approach is about eight times faster than the method presented in Paper C.

The correlated photon mapping method is suitable for applications with time-varying volumes or static volumes using progressive computation. The support for incremental computations based on visual importance implies that interactive performance with high quality illumination can be reached for static volumes with higher resolution. The presented visual importance metric cannot in its current form capture all aspects of visual quality, especially for noisy data. This can be remedied by extending the concept of visual importance with more advanced data models or including more information about the scene.

4.3 Reconstructing what the user wants

Paper E presents an approach that combines the reconstruction and mapping stages within the visualization pipeline to address the feature separation challenges mentioned in Section 3.4.1. The combination of these two stages allows the user's knowledge about the data to be incorporated while avoiding additional interaction complexity. The incorporated knowledge enables a more accurate representation of the studied object to be visualized. The core idea is that the object of study consists of a set of features that are represented by a continuous signal within each feature but may be discontinuous between features. In practice, this means that color and opacity can vary within a feature but will not be affected by other features. Given prior knowledge about the features within the object of study, it is possible to reconstruct a signal for each visualized feature that is better than using a single reconstruction for multiple features.

4.3.1 Feature-based reconstruction

The feature-based reconstruction idea employs a probabilistic model for the features together with a reconstruction per feature as illustrated in Figure 4.10. A data value y_i at a grid point belongs to a certain feature f_j with the probability $P(f_j | y_i) \in [0, 1]$

Figure 4.10: The users' knowledge is incorporated using a classification per feature, which allows a feature-specific reconstruction to be performed. The features can thus be separated and combined into a visual response.

A principal idea of Paper E is that the probability can be derived directly from the existing transfer function editor, which is the key to reduce transfer function interaction complexity. The kernel regression framework by Takeda et al. [69] is extended to allow optimal continuous signal reconstruction within features using the constraints given by the feature classification. The estimated reconstructed signal $\hat{z}(\mathbf{x})$ at location $\mathbf{x}$ for a normalized spatial kernel is given by

$$\hat{z}(\mathbf{x}) = \sum_{y_i \in \Omega_K} w_i y_i, \quad (4.5)$$

where w_i is the spatial weight and Ω_K the neighboring samples within the kernel of size K . The standard formulation for a spatial kernel in Equation 4.5 is extended to support features using an additional feature weight $w_f = P(f_j | y_i)$:

$$\hat{z}_f(\mathbf{x}) = \frac{\sum_{y_i \in \Omega_K} w_i w_f y_i}{\sum_{y_i \in \Omega_K} w_i w_f}. \quad (4.6)$$

Here, the sum in the denominator is for kernel normalization. The reconstructed feature signals are mapped through the transfer function and combined to form the final visual contribution at position $\mathbf{x}$.

4.3.2 Dealing with the partial volume effect

The difficult problem of partial volume is addressed by adaptively selecting the kernel size in Equation 4.6 such that high frequency features are kept in areas with solid features, i.e. corresponding to high probabilities. The maximum kernel size $\sigma_{\max}$ is set to be two times the point spread function (PSF) σ_{PSF} , which is the root

Figure 4.11: A contrast agent (red) and bone (white) are visualized using continuous data reconstruction (left) and using feature-based reconstruction (right). The continuous reconstruction incorrectly produces contrast agent around the skull. The feature-based reconstruction preserves the continuity within the contrast agent and bone features while at the same time preventing interpolation between them. The data was scanned using a dual energy scanner and is therefore multi-variate, which is difficult for other methods to handle.

cause of the partial volume effect. The support of a feature is first determined using a kernel neighborhood $\Omega_{\max}$ of a size equal to the observed PSF:

$$\mathcal{F}_f(\mathbf{x}, \mathbf{f}, \Omega_{\max}) = \sum_{y_i \in \Omega_{\max}} w_i w_f. \quad (4.7)$$

Higher support will be given to sample points where features of high probability are close to the sample point. The adaptive kernel size for a feature is computed using a linear combination of the minimum ($\sigma_{\min} = 2^3$ voxels) and maximum $\sigma_{\max}$ kernel sizes:

$$\sigma_f = \mathcal{F}_f(\mathbf{x}, \mathbf{f}, \Omega_{\max}) \sigma_{\min} + (1 - \mathcal{F}_f(\mathbf{x}, \mathbf{f}, \Omega_{\max})) \sigma_{\max}. \quad (4.8)$$

The adaptive kernel size allows solid features to be reconstructed using a small kernel size and retain a sharp appearance. Note that kernels with zero support will not generate a visual contribution, which means that numerical issues in the denominator Equation 4.6 are avoided.

Results and applicability

Figure 4.11 shows the results of applying the method to a dual energy CT scan of a human head. Using a continuous reconstruction method will visualize the injected contrast agent around the skull due to the transition between air and bone. The feature-based reconstruction approach is capable of preserving the continuous signal within the bone and contrast agent while preventing reconstruction between them. This type of misclassification problem has been addressed previously by

Figure 4.12: Dynamic galleries for volume data exploration. The top view displays the visualization, while the bottom part is reserved for volume data exploration. On the left, the bottom shows a gallery of images generated based on subsets of the intensity domain. The user can get an overview of the data set or drill down to detail within the gallery by zooming using touch interactions. Visual properties can be edited by selecting a gallery image. On the right, the bottom shows how the visualization will look when changing the opacity of the edited data range.

introducing additional attributes during the transfer function mapping, such as gradient magnitude. As discussed in Section 3.4.1, the use of additional attributes adds interaction complexity to the transfer function which is already a limiting factor when exploring volume data.

The applicability of this work is limited by its performance. The method has not been optimized for GPUs and competes with built-in hardware linear interpolation, with next to zero overhead on GPUs.

4.4 Volume data exploration for novice users

Paper F presents a gallery based technique for exploring volume data. The most commonly used 1D transfer function editor is conceptually simple but hard to master due to the lack of spatial connection to the data as discussed in Section 3.4.3. The strengths and weaknesses of the 1D transfer function editor come from the fact that the optical properties are changed based on a projection of the data to the intensity domain. The intensity domain lacks information about where the data exist in the volume and it therefore becomes difficult to predict where the optical property changes will occur in the visualization.

4.4.1 Spatial connection for intuitive understanding

The core idea in Paper F is to add a spatial connection and combine it with interaction to allow the user to predict where a change will have effect while exploring the data set in the intensity domain. The intensity domain is subdivided into equally spaced regions and a preview is generated for each region, forming a gallery that is dynamically updated upon interaction. A zoom operation will

Figure 4.13: Improving intuitiveness by using preview when changing the opacity from low to high. Linearly varying the opacity (top). Varying the opacity using the presented mapping function (bottom).

change the data range used for each preview region. A zoomed out view will thus generate a gallery covering the whole intensity domain, while a zoomed in view will display a subset of the intensity domain. A panning operation will change the displayed data range in the gallery and, correspondingly, filter away the data outside of the displayed range. The user can edit an intensity range by selecting the preview representing the given range. The effect of changing the opacity for the intensity range will be displayed using the dynamic gallery as well as a simplified color selector. An overview of these operations is given in Figure 4.12.

4.4.2 Improving intuitiveness

The method was developed using an iterative design process, which involved improving the design and method in several aspects. Generating a gallery based on linearly varying the opacity from zero to one will produce a non-linear visual appearance due to the exponential behavior of the attenuation, cf. Equation 2.2 and Figure 4.13, top. A mapping function which reverts the non-linearity was therefore introduced. The mapping function alters the opacity in a non-linear fashion such that the visual change is linear under the assumption that the opacity is constant for the data range being edited:

$$\tilde{\alpha} = 1 - (1 - \alpha)^{\frac{1}{\tilde{d}}}, \quad (4.9)$$

where $\tilde{d}$ is a normalization factor that takes the distance a ray passes through the constant opacity into account. The effects of applying the mapping function can be seen in Figure 4.13, bottom.

Most color pickers display two dimensions of the color space and use a slider to control the third dimension. Changing the color in this case involves two interactions, one for selection in the two-dimensional space and one for the slider. Paper F instead proposes to use a color picker where all three dimensions are projected onto a spherical surface using the HSV color space as illustrated in Figure 4.14. The radial angle on the spherical surface represents the hue while the radius is divided into two segments, one for the saturation and one for the

Figure 4.14: Simplified color picker in HSV color space. The user interaction for selecting color is minimized by projecting the three-dimensional color space onto a two-dimensional color sphere.

value. While desaturated colors with varying values are not possible to select, it constitutes a good tradeoff between interaction complexity and flexibility since only a single interaction is necessary.

Results and applicability

A user study was performed in a science center with sixteen participants aged between six and forty years old. The main goal of the study was to find out if the added spatial coupling made the volume data exploration more intuitive, while the secondary goals were to find out how easy, engaging, inviting and fun to use the method was compared to the commonly used widget-based transfer function with a histogram in the background. The study took place around Easter, so the task of the participants was to find an Easter egg hidden in two randomly selected data sets, one for each method. The task involved changing the opacity and color for the intensity values to make the egg visible as well as filtering away features hiding the egg. The initial setup shown to the participants for one of the data sets is depicted in Figure 4.15.

Out of the sixteen participants, three favored the standard transfer function while thirteen favored the dynamic gallery approach. In general, the participants made more conscious decisions with the dynamic gallery method when deciding what to edit. It was observed that many participants pressed the visualization area to see if they could edit the object, a feature missing in both methods. The quantitative results showed that most people thought it was entertaining to explore the volume data using both approaches and would try it in an exhibition, but that the dynamic gallery approach was easier to learn, more fun to use and had a more inviting design.

The presented method is suitable in applications where ease of use is more important than detailed control of advanced transfer function parameters. The user study

Figure 4.15: Initial setup of the user study for the piggy bank data set in Paper F. The standard transfer function editor has been simplified and adapted for touch interaction to allow a better comparison. The participants were assigned the task of finding an Easter egg, which in this case was hidden inside the piggy bank.

indicates that the intuitiveness can be improved by combining the approach further with methods that allow the user to pick and edit features in the final visualization, such as the one presented by Guo et al. [14].

Concluding remarks

In this thesis, methods have been presented to enable high quality illumination for interactive volume rendering as well as tools for exploring data that are better at separating features, dealing with discontinuities and are easier to use for novice users. The applicability of the presented methods has been demonstrated on a range of data sets predominantly from the medical domain. These data sets stem from modalities with different characteristics from low noise CT data to time-varying noisy ultrasound data. This chapter provides a summary of the contributions as well as a reflection on future work.

5.1 Interactive volumetric illumination

Achieving high quality visualizations at interactive frame rates is a challenging task that requires tradeoffs with respect to feasible illumination effects, memory footprint and speed. These aspects have been evaluated with respect to the use of spherical harmonics and photon mapping. A central theme of the presented illumination algorithms is the use of encoded knowledge of the visible content to improve the efficiency of the methods.

A multi-resolution data structure is used for the spherical harmonics representation where the transfer function determines the level of detail within the data structure. This data structure allows the memory overhead and computation time to be reduced while retaining the illumination quality. An evaluation of the trade-off between performance and memory overhead shows that the illumination quality improves linearly with the number of spherical harmonics coefficients. About 30 directions should be used to determine the visibility for each point in the grid and

the best performance is reached when the local visibility radius is between twelve to sixteen voxels. The spherical harmonics visibility approach is best suitable for scenarios with dynamic complex light setups.

The photon mapping technique produces high quality illumination and support a wide range of illumination effects. The presented methods improve the existing photon mapping technique by detecting and utilizing changes within the data that will have a visual effect on the visualization. Computations are focused on changing areas in order to reduce the computation time for the photon tracing and photon gathering. Two efficient methods for detecting changes were presented. The first method encodes the history of the used data for each photon and yields superior performance when the transfer function changes. The second method uses a low resolution grid in object space to detect changes and determine the visual impact of the change. The object space detection enables changing areas to be recomputed for time-varying data in addition to transfer function changes. A priority is assigned based on the visual impact, which is used to retain quality while reducing the amount of required computations. The presented photon mapping methods allow high quality volumetric illumination to be used in interactive applications with time-varying data while changing the transfer function or the camera.

5.2 Volumetric data exploration

The transfer function is one of the core tools to apply when exploring volumetric data. The fundamental concept behind the transfer function is to classify and map a measured data value to optical properties. Each measured value is commonly created using a spatial filter, i.e. the value represents a mixture of objects within the spatial extent of the filter kernel. Reconstructing the original objects from measured values is difficult since the material mixture is generally not known during reconstruction. A method was presented which encodes the user's knowledge about the objects in probabilities in order to improve the reconstruction. Boundaries between objects become sharp in areas where the probability is high and fuzzy in uncertain areas. The user's knowledge can be derived directly from the transfer function and does not add additional interaction complexity. The performance of the presented reconstruction needs to be improved before it can be used in practice.

A transfer function tool that can be immediately understood and used by a user having no previous experience of volumetric data is yet to be presented. A major obstacle in making a transfer function immediately understood is that it is difficult to convey that data from the same point of view can be visualized in different ways. Changing the point of view is familiar from everyday life activities. However, being able to bring out new features that are not visible from the current viewpoint is not part of everyday life and can therefore be a hard concept to grasp. A first take on an intuitive exploration method employs a gallery of images that are dynamically generated based on user interactions. Each gallery image depicts a subset of the full

data range, which enables the user to see the features contained within the data set even though they are not present in the final rendering. A change of opacity can result in a larger or smaller change in the final rendering depending on the value of the opacity due to the non-linear behavior of the underlying function. A novel approach to perform a visually linear change independent of the opacity value was presented and used together with the dynamic gallery to improve the intuitiveness of the method. In a study with sixteen participants, the presented dynamic gallery approach was compared to the most commonly used transfer function editor at a science center. While the participants made more conscious decisions using the novel approach, it was observed that they also wanted to be able to edit features directly in the final visualization. The quantitative results shows that visitors find the dynamic gallery approach to be easier to learn and use.

5.3 Outlook

Volume rendering has come a long way since it was introduced in the late 20th century. The emission-absorption model can be used to visualize static petabyte sized data sets on a desktop computer interactively by streaming the necessary data [17]. However, when it comes to applying illumination, which requires a global access pattern, we still have difficulties handling gigabyte sized data sets interactively. Illumination algorithms utilizing multi-resolution data structures or sparse layouts will need to be designed in order to be applicable to these larger data sets. During the exploration process, they need to be able to take a continuous flow of data into account. Progressive approaches often fail for these types of large data sets when it comes to time-varying data. Most of the data varies and therefore needs to be replaced over a short period of time, which interrupts the progressive rendering employed for generating high quality results. Time-varying data also adds additional constraints on the illumination algorithms since less time can be spent on the rendering and auxiliary data needs to be streamed for all but the smallest time series. In this thesis, focusing the computation on certain parts of the data enabled high quality illumination to be used for time-varying data. However, the presented metrics used for determining which parts to focus on are basic and need to be improved for those scenarios. A performance gain, without losing visual quality, could be made by designing a perceptual metric that determines exactly what is important in a given scenario. A better quality metric could be achieved by further investigating the perceptual effects of illumination and taking domain specific knowledge into account.

Linear interpolation is ubiquitously used when it comes to reconstruction of volume data on the GPU. The built-in functionality produces next to zero overhead compared to fetching the nearest neighbors. Re-implementing linear interpolation by fetching the nearest neighbors or using any other type of reconstruction has a substantial performance penalty, which prevents better reconstruction methods

from being used in real-world applications. New opportunities will open up if the hardware manufacturers would allow developers and researchers to alter the built-in interpolation scheme.

Transfer functions have thus far mostly been developed for general use-cases such that they can be applied to any type of data. While the tools are good for many users, they are not perfect for everyone. The medical field is a good example. While the tools for exploring volume data exist in the medical workstations, they are rarely used due to the time effort it takes to learn and use them. Radiologists could potentially start to explore the data if they had a tool specifically designed for their examination. Time-varying volumetric data exploration requires an entirely different approach on the transfer function side. A single time-step is likely not the focus of the data anymore. Tuning the transfer function to a single time-step will cause features to appear and disappear over time in the presence of noise. The tools therefore need to adapt to the variability of the data such that features are visualized over time consistently. The data needs to be aggregated in a way that is intuitive for the user to understand and presented such that an overview of the complete data over time is given. Intuitive interaction needs to be added as well to allow the user to drill down into specific regions within a time step.

In a broader perspective, I believe that the combination of medical visualization and augmented reality will have a major impact on healthcare. Vital organs and important information can, for example, be augmented on the body during surgery. A surgeon will thereby be able to see the salient features within the body, thus increasing the precision and making it easier to both plan and perform an operation. There will also be an increase in the amount and variety of information about a patient during daily activities due to the use of more sensors measuring for example heart rate and insulin levels. Both the amount and the variety is a challenge for this type of data. Visualization will have a key role in making sense of the data to prevent illness and diagnose patients.

We have come a long way since the introduction of volume visualization and visualization in general, but there is still an abundance of challenges to address!

Bibliography

- [1] F. Attneave. Some informational aspects of visual perception. *Psychological review*, 61(3):183, 1954. [page 1]
- [2] U. Behrens and R. Ratering. Adding Shadows to a Texture-Based Volume Renderer. In *IEEE Symposium on Volume Visualization*, pages 39–46, 1998. [page 21]
- [3] S. Bruckner and T. Möller. Isosurface Similarity Maps. *Computer Graphics Forum*, 29(3):773–782, 2010. [page 26]
- [4] N. Bruno and J. E. Cutting. Minimodularity and the Perception of Layout. *Journal of Experimental Psychology: General*, 117(2):161–170, 1988. [page 3]
- [5] B. Cabral, N. Cam, and J. Foran. Accelerated Volume Rendering and Tomographic Reconstruction using Texture Mapping Hardware. In *Symposium on Volume Visualization*, pages 91–98. ACM, 1994. [pages 20 and 21]
- [6] C. D. Correa and K.-L. Ma. Size-based Transfer Functions: A New Volume Exploration Technique. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1380–1387, 2008. [page 25]
- [7] R. Drebin, L. Carpenter, and P. Hanrahan. Volume rendering. *Computer Graphics (Proceedings of SIGGRAPH 1988)*, 22(4):65–74, 1988. [page 24]
- [8] R. A. Drebin, L. Carpenter, and P. Hanrahan. Volume rendering. In *Computer Graphics (Proceedings of SIGGRAPH 1988)*, pages 65–74, 1988. [page 11]
- [9] R. R. Edelman and S. Warach. Magnetic Resonance Imaging. *New England Journal of Medicine*, 328(10):708–716, 1993. [page 8]
- [10] A. R. Edmonds. *Angular momentum in quantum mechanics*. Princeton University Press, 1996. [page 22]
- [11] K. Engel, M. Hadwinger, J. M. Kniss, C. Rezk-Salama, and D. Weiskopf. *Real-Time Volume Graphics*. A K Peters/CRC Press, Boca Raton, FL, USA, 2006. [page 5]

- [12] T. Etienne, D. Jönsson, T. Ropinski, C. Scheidegger, J. L. D. Comba, L. G. Nonato, R. M. Kirby, A. Ynnerman, and C. T. Silva. Verifying Volume Rendering using Discretization Error Analysis. *IEEE Transactions on Visualization and Computer Graphics*, 20(1):140–154, 2014.
- [13] J. J. Gibson. The perception of visual surfaces. *The American journal of psychology*, 63(3):367–384, 1950. [page 3]
- [14] H. Guo, N. Mao, and X. Yuan. WYSIWYG (What You See is What You Get) Volume Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2106–2114, 2011. [page 47]
- [15] R. B. Haber and D. A. McNabb. Visualization Idioms: A Conceptual Model for Scientific Visualization Systems. *Visualization in scientific computing*, pages 74–93, 1990. [page 6]
- [16] M. Hadwiger, C. Berger, and H. Hauser. High-Quality Two-Level Volume Rendering of Segmented Data Sets on Consumer Graphics Hardware. In *IEEE Visualization*, pages 301–308, 2003. [page 25]
- [17] M. Hadwiger, J. Beyer, W. K. Jeong, and H. Pfister. Interactive Volume Exploration of Petascale Microscopy Data Streams Using a Visualization-Driven Virtual Memory Approach. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2285–2294, 2012. [page 51]
- [18] L. G. Henyey and J. L. Greenstein. Diffuse Radiation in The Galaxy. *The Astrophysical Journal*, 93:70–83, 1941. [page 19]
- [19] F. Hernell, P. Ljung, and A. Ynnerman. Efficient Ambient and Emissive Tissue Illumination using Local Occlusion in Multiresolution Volume Rendering. In *Eurographics/IEEE VGTC Symposium on Volume Graphics*, pages 1–8, 2007. [page 21]
- [20] F. Hernell, P. Ljung, and A. Ynnerman. Interactive Global Light Propagation in Direct Volume Rendering using Local Piecewise Integration. In *IEEE/EG Symposium on Volume and Point-Based Graphics*, pages 105–112, 2008. [page 32]
- [21] F. Hernell, P. Ljung, and A. Ynnerman. Local Ambient Occlusion in Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 16(4):548–559, 2010. [page 33]
- [22] H. W. Jensen. Global Illumination using Photon Maps. In *Rendering Techniques*, pages 21–30. Springer-Verlag, 1996. [page 23]
- [23] H. W. Jensen. *Realistic Image Synthesis Using Photon Mapping*. AK Peters, Ltd., 2001. [page 36]

- [24] H. W. Jensen and P. H. Christensen. Efficient Simulation of Light Transport in Scenes with Participating Media Using Photon Maps. In *Computer Graphics (Proceedings of SIGGRAPH 1998)*, pages 311–320, 1998. [page 34]
- [25] D. Jönsson and A. Ynnerman. Correlated Photon Mapping for Interactive Global Illumination of Time-Varying Volumetric Data. *IEEE Transactions on Visualization and Computer Graphics*, 23, 2017, in press.
- [26] D. Jönsson, P. Ganestam, M. Doggett, A. Ynnerman, and T. Ropinski. Explicit Cache Management for Volume Ray-Casting on Parallel Architectures. In *Eurographics Symposium on Parallel Graphics and Visualization*, pages 31–40, 2012.
- [27] D. Jönsson, J. Kronander, T. Ropinski, and A. Ynnerman. Historygrams: Enabling Interactive Global Illumination in Direct Volume Rendering Using Photon Mapping. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2364–2371, 2012.
- [28] D. Jönsson, E. Sundén, A. Ynnerman, and T. Ropinski. State of The Art Report on Interactive Volume Rendering with Volumetric Illumination. In *EG - State of the Art Reports*, volume 1, pages 53–74, 2012.
- [29] D. Jönsson, E. Sundén, A. Ynnerman, and T. Ropinski. A Survey of Volumetric Illumination Techniques for Interactive Volume Rendering. *Computer Graphics Forum*, 33(1):27–51, 2014.
- [30] D. Jönsson, M. Falk, and A. Ynnerman. Intuitive Exploration of Volumetric Data Using Dynamic Galleries. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):896–905, 2016.
- [31] D. Jönsson, E. Sundén, G. Låthén, and I. W. Hachette. Method and System for Volume Rendering of Medical Images, patent pending 2016. [Online]. Available: <http://patentscope.wipo.int/search/en/detail.jsf?docId=US173535325>, Accessed: 2016-09-05.
- [32] A. Kaufman and K. Mueller. *The Visualization Handbook*, chapter Overview of Volume Rendering, pages 127–174. Elsevier, 2005. [page 25]
- [33] J. Kniss, G. Kindlmann, and C. Hansen. Multi-Dimensional Transfer Functions for Interactive Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 8(3):270–285, 2002. [pages 12 and 24]
- [34] J. Kniss, S. Premoze, C. Hansen, and D. Ebert. Interactive Translucent Volume Rendering and Procedural Modeling. In *IEEE Visualization*, pages 109–116, 2002. [page 21]

- [35] A. König and E. Gröller. Mastering Transfer Function Specification by using VolumePro Technology. In *Spring Conference on Computer Graphics*, number 17, pages 279–286, 2001. [page 27]
- [36] T. Kroes, F. H. Post, and C. P. Botha. Exposure Render: An Interactive Photo-Realistic Volume Rendering Framework. *PLoS ONE*, 7(7):1–10, 2012. [pages 21 and 23]
- [37] J. Kronander, D. Jönsson, J. Löw, P. Ljung, A. Ynnerman, and J. Unger. Efficient Visibility Encoding for Dynamic Illumination in Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 18(3):447–462, 2012.
- [38] J. Kronander, J. Dahlin, D. Jönsson, M. Kok, T. Schön, and J. Unger. Real-Time Video Based Lighting Using GPU Raytracing. In *European Signal Processing Conference (EUSIPCO 2014)*, pages 1627–1631, 2014.
- [39] J. Krüger and R. Westermann. Acceleration Techniques for GPU-based Volume Rendering. In *IEEE Visualization*, pages 287–292, 2003. [page 20]
- [40] M. Levoy. Display of Surfaces from Volume Data. *IEEE Computer Graphics and Applications*, 8(3):29–37, 1988. [page 24]
- [41] F. Lindemann and T. Ropinski. Advanced Light Material Interaction for Direct Volume Rendering. In *Eurographics/IEEE Symposium on Volume Graphics*, pages 101–108, 2010. [page 21]
- [42] S. Lindholm, P. Ljung, C. Lundström, A. Persson, and A. Ynnerman. Spatial Conditioning of Transfer Functions Using Local Material Distributions. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1301–1310, 2010. [page 25]
- [43] S. Lindholm, D. Jönsson, H. Knutsson, and A. Ynnerman. Towards Data Centric Sampling for Volume Rendering. In *SIGRAD*, pages 55–60, 2013.
- [44] S. Lindholm, D. Jönsson, C. Hansen, and A. Ynnerman. Boundary Aware Reconstruction of Scalar Fields. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2447–2455, 2014.
- [45] P. Ljung, C. Lundström, A. Ynnerman, and K. Museth. Transfer Function Based Adaptive Decompression for Volume Rendering of Large Medical Data Sets. In *IEEE Symposium on Volume Visualization and Graphics*, pages 25–32, 2004. [page 32]
- [46] P. Ljung, J. Krüger, E. Gröller, M. Hadwiger, C. D. Hansen, and A. Ynnerman. State of the Art in Transfer Functions for Direct Volume Rendering. 35(3): 669–691, 2016. [page 11]

- [47] C. Lundström, P. Ljung, and A. Ynnerman. Local Histograms for Design of Transfer Functions in Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1570–1579, 2006. [page 26]
- [48] C. Lundström, T. Rydell, C. Forsell, A. Persson, and A. Ynnerman. Multi-Touch Table System for Medical Visualization: Application to Orthopedic Surgery Planning. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1775–1784, 2011. [page 26]
- [49] R. Mantiuk, S. Daly, K. Myszkowski, and H.-P. Seidel. Predicting Visible Differences in High Dynamic Range Images: Model and its Calibration. In *IST/SPIE Symposium on Electronic Imaging*, volume 5666, pages 204–214, 2005. [page 33]
- [50] J. Marks, B. Andalman, P. A. Beardsley, W. Freeman, S. Gibson, J. Hodgins, T. Kang, B. Mirtich, H. Pfister, W. Ruml, K. Ryall, J. Seims, and S. Shieber. Design Galleries: A General Approach to Setting Parameters for Computer Graphics and Animation. In *Computer Graphics (Proceedings of SIGGRAPH 1997)*, pages 389–400, 1997. [page 26]
- [51] N. Max. Optical Models for Direct Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995. [page 20]
- [52] K. T. Nguyen, A. Eklund, H. Ohlsson, F. Hernell, P. Ljung, C. Forsell, M. T. Andersson, H. Knutsson, and A. Ynnerman. Concurrent Volume Visualization of Real-Time fMRI. In *IEEE/EG Volume Graphics*, pages 53–60, 2010. [page 8]
- [53] J. Parulek, D. Jönsson, T. Ropinski, S. Bruckner, A. Ynnerman, and I. Viola. Continuous Levels-of-Detail and Visual Abstraction for Seamless Molecular Visualization. *Computer Graphics Forum*, 33(6):276–287, 2014.
- [54] D. Patel, M. Haidacher, J.-P. Balabanian, and E. M. Gröller. Moment Curves. In *IEEE Pacific Visualization Symposium*, pages 201–208, 2009. [page 24]
- [55] F. d. M. Pinto and C. M. D. S. Freitas. Two-Level Interaction Transfer Function Design Combining Boundary Emphasis, Manual Specification and Evolutive Generation. In *Brazilian Symposium on Computer Graphics and Image Processing*, pages 281–288. IEEE, 2006. [page 26]
- [56] J.-S. Praßni, T. Ropinski, J. Mensmann, and K. Hinrichs. Shape-based Transfer Functions for Volume Visualization. In *IEEE Pacific Visualization Symposium*, pages 9–16, 2010. [page 25]
- [57] B. Preim and C. Botha. *Visual Computing for Medicine*, chapter Part II: Visualization and Exploration of Medical Volume Data, pages 227–368. Morgan Kaufmann, Boston, USA, second edition, 2014. [page 5]

- [58] F. Qiu, F. Xu, Z. Fan, N. Neophytos, A. Kaufman, and K. Mueller. Lattice-Based Volumetric Global Illumination. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1576–1583, 2007. [page 21]
- [59] T. Ropinski, J. Meyer-Spradow, S. Diepenbrock, J. Mensmann, and K. H. Hinrichs. Interactive Volume Rendering with Dynamic Ambient Occlusion and Color Bleeding. *Computer Graphics Forum*, 27(2):567–576, 2008. [page 21]
- [60] T. Ropinski, C. Döring, and C. Rezk-Salama. Interactive Volumetric Lighting Simulating Scattering and Shadowing. In *IEEE Pacific Visualization Symposium*, pages 169–176, 2010. [page 21]
- [61] N. Rostamzadeh, D. Jönsson, and T. Ropinski. Comparison of Volumetric Illumination Methods by Considering the Underlying Optical Models. In *SIGRAD*, pages 35–40, 2013.
- [62] Y. Sato, C.-F. Westin, A. Bhalerao, S. Nakajima, N. Shiraga, S. Tamura, and R. Kikinis. Tissue Classification Based on 3D Local Intensity Structures for Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 6(2):160–180, 2000. [page 25]
- [63] P. Schlegel, M. Makhinya, and R. Pajarola. Extinction-based Shading and Illumination in GPU Volume Ray-Casting. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):1795–1802, 2011. [page 21]
- [64] P. Sloan, J. Kautz, and J. Snyder. Precomputed Radiance Transfer for Real-Time Rendering in Dynamic, Low-Frequency Lighting Environments. In *ACM Trans. on Graphics (Proceedings of SIGGRAPH)*, pages 527–536, 2002. [page 21]
- [65] M. Smelyanskiy, D. Holmes, J. Chhugani, A. Larson, D. M. Carmean, D. Hanson, P. Dubey, K. Augustine, D. Kim, A. Kyker, V. W. Lee, A. D. Nguyen, L. Seiler, and R. Robb. Mapping High-Fidelity Volume Rendering for Medical Imaging to CPU, GPU and Many-Core Architectures. *IEEE Transactions on Visualization and Computer Graphics*, 15:1563–1570, 2009. [page 21]
- [66] W. A. Stokes, J. A. Ferwerda, B. Walter, and D. P. Greenberg. Perceptual Illumination Components: A New Approach to Efficient, High Quality Global Illumination Rendering. *ACM Trans. on Graphics*, 23(3):742–749, 2004. [page 16]
- [67] E. Sundén, A. Bock, D. Jönsson, A. Ynnerman, and T. Ropinski. Interaction Techniques as a Communication Channel when Presenting 3D Visualizations. In *IEEE VIS International Workshop on 3DVis*, pages 2–6, 2014.

- [68] N.-C. Tai and M. Inanici. Space Perception and Luminance Contrast: Investigation and Design Applications Through Perceptually Based Computer Simulations. In *Spring Simulation Multiconference*, pages 185:1–185:8, 2010. [page 16]
- [69] H. Takeda, S. Farsiu, and P. Milanfar. Kernel Regression for Image Processing and Reconstruction. *IEEE Trans. on Image Processing*, 16(2):349–366, 2007. [page 42]
- [70] U. Tiede, T. Schiemann, and K. H. Höhne. High Quality Rendering of Attributed Volume Data. In *Proceedings of Visualization '98*, pages 255–262. IEEE, 1998. [page 25]
- [71] D. W. Townsend. Positron Emission Tomography/Computed Tomography. *Seminars in Nuclear Medicine*, 38(3):152 – 166, 2008. [page 8]
- [72] R. van Lier and J. Wagemans. Perceptual Grouping Measured by Color Assimilation: Regularity Versus Proximity. *Acta Psychologica*, 97(1):37 – 70, 1997. [page 3]
- [73] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. [page 41]
- [74] M. Wertheimer. Untersuchungen zur Lehre von der Gestalt. II. *Psychological Research*, 4(1):301–350, 1923. [page 1]
- [75] Y. Zhang and K.-L. Ma. Fast Global Illumination for Interactive Volume Visualization. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D)*, pages 55–62, 2013. [page 21]
- [76] Y. Zhang and K.-L. Ma. Lighting Design for Globally Illuminated Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2946–2955, 2013. [page 16]
- [77] Y. Zhang, Z. Dong, and K.-L. Ma. Real-Time Volume Rendering in Dynamic Lighting Environments Using Precomputed Photon Mapping. *IEEE Transactions on Visualization and Computer Graphics*, 19(8):1317–1330, 2013. [page 21]

Publications

Publications

The articles associated with this thesis have been removed for copyright reasons. For more details about these see:

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-131023>