Efficient Methods for Volumetric Illumination

Frida Gyllensvärd

Department of Science and Technology
Linköping University
Norrköping 2011
Efficient Methods for Volumetric Illumination

Frida Gyllensvård

Cover Image:

Copyright © 2011 Frida Gyllensvård, unless otherwise noted.
Printed by LiU-Tryck, Linköping 2011
Linköping Studies in Science and Technology Dissertations No. 1406
ISBN 978-91-7393-041-3 ISSN 0345-7524
Abstract

Modern imaging modalities can generate three-dimensional datasets with a very high detail level. To transfer all the information to the user in an efficient way there is a need for three-dimensional visualization. In order to enhance the diagnostic capabilities the utilized methods must supply the user with fast renderings that are easy to interpret correctly.

It can thus be a challenge to visualize a three-dimensional dataset in a way that allows the user to perceive depth and shapes. A number of stereoscopic solutions are available on the market but it is in many situations more practical and less expensive to use ordinary two-dimensional displays. Incorporation of advanced illumination can, however, improve the perception of depth in a rendering of a volume. Cast shadows provide the user with clues of distances and object hierarchy. Simulating realistic light conditions is, however, complex and it can be difficult to reach interactive frame rates. Approximations and clever implementations are consequently required.

This thesis presents efficient methods for calculation of illumination with the objective of providing the user with high spatial and shape perception. Two main types of light conditions, a single point light source and omni-directional illumination, are considered. Global transport of light is efficiently estimated using local piecewise integration which allows a graceful speed up compared to brute force techniques. Ambient light conditions are calculated by integrating the incident light along rays within a local neighborhood around each point in the volume.

Furthermore, an approach that allows the user to highlight different tissues, using luminous materials, is also available in this thesis. A multiresolution data structure is employed in all the presented methods in order to support evaluation of illumination for large scale data at interactive frame rates.
Acknowledgements

First of all, I would like to thank my supervisors Anders Ynnerman and Anders Persson for being supportive and inspiring. I would also like to show my appreciation to Claes Lundström for valuable assistance during my last year as a PhD student.

A special thanks to my co-author Patrik Ljung for fruitful collaboration and helpful discussions. Thanks also to my co-author Tan Khoa Nguyen. I would also like to acknowledge Matthew Cooper for assistance in proof-reading of manuscripts. Furthermore, I wish to express appreciation to Center for Medical Image Science and Visualization (CMIV) for supplying me with interesting data to visualize. Thanks also to former and present colleagues at Media and Information Technology (MIT) and CMIV.

I would also like to thank my family for all their love and encouragement. I am very grateful to my parents and sisters for being so supportive.

Most of all I wish to thank my husband Björn and our little sunshine Elsa. You two truly fill my life with light!

This work has been supported by the Swedish Research Council, grant 621-2004-3829 and 621-2008-1257, the Linnaeus Center CADICS and the Strategic Research Center MOVIII, founded by the Swedish Foundation for Strategic Research, SSF.
# Contents

1 Introduction ................................................. 3
   1.1 Medical Visualization .................................. 4
   1.2 From Data Acquisition to User Interaction ............. 5
      1.2.1 Generation of Medical Data ......................... 5
      1.2.2 Voxel Classification ................................. 7
      1.2.3 Direct Volume Rendering ............................. 8
   1.3 Challenges in Volumetric Illumination ................. 9
   1.4 Contributions ........................................... 11

2 Aspects of Volumetric Lighting ................................. 13
   2.1 Light Interactions .................................... 14
   2.2 The Volume Rendering Integral ........................ 16
      2.2.1 Local Estimation of \( g(s) \) ....................... 17
      2.2.2 Numerical Approximation ............................ 19
      2.2.3 GPU Raycasting ..................................... 19
   2.3 Global Illumination Approximations .................... 20
      2.3.1 Volumetric Shadows and Scattering Effects .......... 21
      2.3.2 Ambient Occlusion ................................ 21

3 Improving Volumetric Illumination ............................. 25
   3.1 Ambient Occlusion for Direct Volume Rendering .......... 25
   3.2 Local Ambient Occlusion ................................ 26
      3.2.1 Light Contribution ................................ 27
      3.2.2 Absorption TF ...................................... 29
   3.3 Luminous Illumination Effects .......................... 29
      3.3.1 Using Light as an Information Carrier ................ 31
   3.4 Global Light Transport ................................ 32
      3.4.1 Local Piecewise Integration ......................... 33
      3.4.2 First Order Scattering Effects ..................... 37
   3.5 Pipeline Processing ..................................... 37
      3.5.1 Flat Multiresolution Blocking ....................... 38
      3.5.2 Multiresolution Illumination Estimations .......... 38
      3.5.3 LAO Pipeline ......................................... 40
List of Publications

This thesis is based on the following papers, which will be referred to in the text by their Roman numerals. The papers are appended at the end of the thesis.

I Efficient Ambient and Emissive Tissue Illumination using Local Occlusion in Multiresolution Volume Rendering  
Frida Hernell, Patric Ljung and Anders Ynnerman  
In Proceedings Eurographics/IEEE VGTC Symposium on Volume Graphics 2007, Prague, Czech Republic

II Interactive Global Light Propagation in Direct Volume Rendering using Local Piecewise Integration  
Frida Hernell, Patric Ljung and Anders Ynnerman  
In Proceedings Eurographics/IEEE VGTC on Volume and Point-Based Graphics 2008, Los Angeles, California, USA

III Local Ambient Occlusion in Direct Volume Rendering  
Frida Hernell, Patric Ljung and Anders Ynnerman  

IV Concurrent Volume Visualization of Real-Time fMRI  
Tan Khoa Nguyen, A. Eklund, Henrik Ohlsson, Frida Hernell, Patric Ljung, Camilla Forsell, Mats Andersson, Hans Knutsson and Anders Ynnerman  
Introduction

Vision is one of the body’s most important senses in order to create an awareness of the surrounding world. The ability to see is made possible by allowing the retina to intercept visible light and convert it to impulses that are sent to the brain for interpretation. The three-dimensional world can easily be misinterpreted since the retina only captures a two-dimensional image. Different cues are therefore necessary in order to perceive depth correctly. The most obvious cue is stereopsis which uses the binocular disparity from the eyes’ horizontal separation supplying the brain with clues of distances.

In art it is desirable to trick the brain to believe that a painting has depth. In this case it is not possible to use stereopsis, so other cues are needed. A number of cues can be used such as, linear perspective, interposition and relative size. However, a very effective cue is caused by lighting [Lip82]. Cast shadows caused by a light source can help us to comprehend the spatial relationships between objects. Figure 1.1 demonstrates the importance of shadowing for determination of distance. In the left image the cone seems to be located behind the box and the cylinder, which is an illusion. As the shadows are shown, in the right image, it is suddenly more obvious that the cone is hovering above the ground and located along the same line as the other objects.

![Figure 1.1](image.png)

Figure 1.1: The cast shadows (right image) are important in order to perceive the spatial relations between the three objects. When shadows are absent (left image) it becomes difficult to comprehend that the cone is hovering above the line that the box and the cylinder are located along.
1. INTRODUCTION

Figure 1.2: Shading is very important in order to perceive shapes correctly. Details become exposed in the shaded teapot (right) compared with the unshaded teapot (left).

How light is reflected on an object can furthermore reveal information about the object’s solidity and curvature. An illustration of how simple shading can enhance the perception of shape is provided in figure 1.2.

Images on a computer screen face the same problems as paintings concerning perception and the same types of cues can be employed. 3D solutions with goggles and 3D screens are nowadays available which evidently improves the perception of depth. However, traditional visualization on a 2D screen is still desirable in many situations.

A number of perceptual studies, for example [Wan92], [WFG92], [HWSB99] and [LB99] has shown that illumination and shadows can affect the interpretation of spatial relations and shapes positively in computer generated images. One application where it is particularly important with correct depth perception is medical visualization.

1.1 Medical Visualization

The use of volume visualization is increasing rapidly, especially in the medical field. For instance, when looking at vessels using Computed Tomography Angiography (CTA) and complex fractures, where the depth perception plays an important role in the understanding of the information. An additional example is postmortem imaging (virtual autopsies) [LWP+06] which can serve as a great complement to standard autopsies.

Since the visualization is examined by medical experts, and the examination may lead to a decision about a patient’s treatment, it is important that the volume is perceived correctly. A wide range of approaches that aim at providing the user with informative visual representations of medical data have been presented over the years. Both illustrative techniques [BGK05, BG06] and attempts to simulate more photo-realistic images have been provided (see examples in chapter 2).

Introducing realistic illumination implies advanced computations which can be troublesome to evaluate interactively, especially for medical data that is often large. This thesis focuses on how to improve illumination in volume visualization at interactive speed. The techniques described in this thesis can be used for any volumetric dataset but the main focus is on medical applications. A short presentation of the steps in image synthesis from three-dimensional medical data is provided next.
1.2 From Data Acquisition to User Interaction

To accomplish an improved medical visualization it is important to understand the properties of the data to visualize. A brief review of the whole chain (see figure 1.3), from acquisition of data to a final rendered image, is presented in this section.

1.2.1 Generation of Medical Data

There are a number of different imaging modalities available in hospitals today. However, Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) are the most commonly used for visualization in 3D. The techniques behind these two modalities differ greatly and the resulting images highlight different aspects of the patient’s anatomy (see figure 1.4). The obtained data is often stored as a stack of two-dimensional images, also called slices, which together build up a volume. A data element in the volume is referred to as a voxel, which is analogous to a pixel in two-dimensions. A brief presentation of CT and MRI is provided next.

Computed Tomography

A problem with conventional x-ray techniques is that only a two-dimensional image is created. This means that some organs are overlapped and it can be difficult to find important information. Computed Tomography (CT), on the other hand, is an x-ray technique that stores the result in a rectilinear grid which allows for visualization in 3D. With this technique, small x-ray sources and corresponding detectors spin around the patient in a donut-shaped device. This makes it possible to collect data from multiple angles. Since different tissues absorb the x-ray radiation differently, it is possible to classify the different inner organs of a human body. An advantage with CT is the ability to separate bones from soft tissues. It can, however, be difficult to identify structures within soft tissues.
1. INTRODUCTION

a) CT  b) MRI  c) fMRI

Figure 1.4: A comparison of axial brain scans generated with CT (a) and MRI (b). For instance, the skull is apparent in the CT scan while the structures in the soft matter is more evident in the MRI scan. The MR scanner can also be used to derive functional images (c) in which activity for a specific task, for instance motor control, is imaged.

An injection of a "contrast agent" can be used in order to highlight, for example, vascular lumen, in order to enhance the possibility of finding abnormalities.

Magnetic Resonance Imaging

Nuclear Magnetic Resonance Imaging is the full name of this technique but MRI is the most commonly used abbreviation. MRI is based on the magnetic property of hydrogen atoms. The nuclei of these atoms acts as small compass needles when exposed to a strong magnetic field. If radio waves, at a certain frequency, are applied then the nucleus starts spinning. The hydrogen atoms emit energy when returning to their normal state. These emissions are detected and different properties are used as a basis for deriving an image.

MRI is superior to CT when it comes to distinguishing structures in soft tissues. However, it is difficult to detect bones since only tissues containing substantial amounts of water are found. Other information than anatomical structures can additionally be obtained with the MR scanner, for instance blood flow and functional activity. A short description of functional MRI is provided next.

Functional MRI

The aim of functional MRI (fMRI) is to localize different functions, for example language and motor control, instead of anatomical structures as in CT and MRI. This information can be of great aid when removing a brain tumor since awareness of where important regions are located is helpful in order to preserve as much functionality as possible.
1.2. FROM DATA ACQUISITION TO USER INTERACTION

Figure 1.5: A transfer function is used to map the original scalar values to colors and opacities. The volume can thereby be visualized in many different ways.

fMRI is based on the magnetic property of hemoglobin molecules. If a region in the brain is active then there is a higher demand for oxygenated blood to that region. Since the magnetic property varies depending on whether the hemoglobin molecules are oxygenated or not, it is possible to determine the level of activity in a region. During an fMRI examination the patient performs different tasks, that activate an area of interest, in short intervals interleaved with rests. If the task sequence correlates with a Blood Oxygen Level Dependent (BOLD) signal then activity is considered to be found. A strong signal correlation implies that there is a high certainty that activity has been found.

1.2.2 Voxel Classification

The data acquired by the different imaging modalities often contain one scalar value for each voxel. To visually separate different tissue types it is possible to map these scalar values to colors. Furthermore, it is often helpful to make some tissues transparent or semi-transparent in order to reveal other structures inside the body. An opacity value must therefore also be mapped to each scalar value. These mappings are often performed with a transfer function (TF) as illustrated in figure 1.5. The original scalar values, $s$, are usually mapped to four-component color vectors, $c$, consisting of color (Red,Green,Blue) and opacity (Alpha), as in equation 1.1.

$$c = T(s), \quad T : \mathbb{R} \rightarrow \mathbb{R}^4$$ (1.1)

In medical applications it is common that the user adjusts with the TF in order to find appropriate settings that correspond to the current aim of the visualization process. To facilitate fine-tuning of a TF in medical applications it is important that this process is
fast and that the resulting rendering can be seen interactively. Figure 1.6 illustrates how a TF can be used to reveal different aspects of a volume.

It can sometimes be difficult to find a suitable TF depending on the scalar value ranges. Care must be taken since neighboring tissues can have similar scalar values and coloring of the volume can thereby easily lead to misinterpretation of the volume content. Using domain knowledge, for instance with local histograms [LLY06b], can enhance the likelihood of classifying voxels correctly.

1.2.3 Direct Volume Rendering

With the small amount of light that reaches the retina it is possible for the brain to create an image of the surrounding environment, as mentioned in the introduction. A photo can be created in a similar way with image sensors that capture light. Actually, the origin of the word photography is Greek and means “drawing with light”. When rendering an image of a 3D volume each pixel is a sensor that captures the tiny fraction of light that reaches the pixel.

In direct volume rendering (DVR) the volume is considered to be a set of light emitting particles with varying densities. The transparency of the particles affects the ability of
1.3 Challenges in Volumetric Illumination

The complexity of the computations used in raycasting affects the feeling of realism in a rendered image. For instance, inclusion of an external light source that casts shadows can
increase the visual quality greatly. Evaluation of realistic illumination requires advanced computations since light can be absorbed, emitted or scattered in different directions at arbitrary points when traveling through a participating medium. Furthermore, the path between a voxel and the light source must be examined in order to find out if it is shadowed or not. The content in other voxels along the path might occlude the light and reduce the radiance to the voxel. For a large volume these evaluations can easily become very intense. Simplifications are therefore needed in order to perform the calculations at interactive frame rates which is important in medical applications. Further descriptions of why the calculations are so complex are provided in chapter 2.

A volume is most commonly illuminated by a single light source. However, according to Langer and Bülthoff [LB99], omni-directional illumination, with light arriving from all directions, is especially advantageous in order to increase the human’s ability to distinguish objects and their properties. With omni-directional illumination the volume appears as if it was illuminated on an overcast day. However, it is a challenge to consider incident light from all directions simultaneously while reaching interactive frame rates.

Incorporation of illumination in applications used for medical diagnosis is associated with additional challenges. For example, tissues that shall be examined by medical experts are not allowed to be hidden due to absence of light. If important regions are fully shadowed then it is impossible to make an accurate diagnosis. The quality of the acquired medical data can also be a challenge. Volumes generated with MRI can sometimes be very noisy and it can be difficult to preserve the perception of shapes without intensifying the noise. An additional challenge, related to medical datasets, is that some of the modalities can nowadays generate volumes with a very high level of detail which results in large datasets. These datasets can be difficult to handle due to limitations of storage space and computational cost. Large datasets are often approached with some kind of multiresolution data structure. It is thus also important to implement the illumination calculations in a way that allows fast calculations for large datasets. For instance, by utilizing the possibility of running parallel computations on the Graphics Processing Unit (GPU).

To summarize, the challenges to be approached in volumetric illumination in order to enhance diagnostic capabilities are:

1. to find suitable approximations that can be evaluated at interactive speed
2. to support large scale data
3. to handle noisy data
4. to avoid light conditions where some regions are fully shadowed
5. to handle single light sources
6. to support omni-directional illumination
1.4 Contributions

This thesis contributes with methods that incorporate advanced illumination in DVR. The objective is to enhance the depth and shape perception in medical visualization at interactive frame rates. The published papers, which are appended to this thesis, provide detailed descriptions of the individual contributions.

- **Paper I** presents an efficient method for local ambient occlusion and emission in DVR that supports interactive frame rates.

- **Paper II** provides a method based on piecewise integration for efficient approximation of volumetric light transport from a point light source at an arbitrary position.

- **Paper III** extends the techniques from paper I with an adaptive sampling scheme and an additional TF for absorbed light.

- **Paper IV** investigates the possibility of using emission as an information carrier to improve the visual clarity of fMRI active regions.
Aspects of Volumetric Lighting

Medical volumes consist of different tissue types with different densities that influence the transport of light in various ways. The interaction of light in this type of volume is comparable with light interactions in a simple cloud in the sky, which is a volume with varying densities of water particles (see figure 2.1). In some regions the particles are distributed very sparsely so that a large amount of light can penetrate. Other regions attenuate the light quickly, creating large shadows both within the volume and on the ground. Light can also be refracted and reflected by the water particles creating scattering effects. Simulating these phenomena is computationally demanding and several methods, that balance the image quality against a desired frame rate, have been presented over the years.

Figure 2.1: A cloud is a volume that consists of varying densities of water particles. Light beams from the sun are either attenuated within the volume or are scattered in various directions before leaving the cloud. In a densely packed cloud the majority of beams are attenuated causing shadows on the ground. This is a photo of clouds taken on a vacation in Norway.
To conquer the challenge of reaching interactive frame rates while simulating realistic illumination in large scale medical volumes it is necessary to find suitable approximations. It is thus important to understand how light travels through a participating medium. This chapter provides a short review of how light interacts in a volume and an overview of previous research efforts in finding appropriate approximations for computationally demanding illumination estimations.

2.1 Light Interactions

If light does not intersect with any participating medium it carries on in the original direction. The possible influences of light when arriving at a particle in a volume are illustrated in figure 2.2. It can either be scattered away in another direction or be absorbed by the particle and turned into heat, which reduces the radiance in the medium. Heat can also be used to generate light which makes the particle emissive \cite{HKRs+06}. Different materials have different abilities to absorb, scatter and emit light so the content of a volume determines precisely how light travels through it.

The most simple approach when illuminating a volume is to define a background intensity and restrict the computations to only consider emission and absorption, as illustrated in figure 2.3a. The particles with which the ray intersects can, with this method, only increase or decrease the radiance depending on their ability to absorb and emit light. The radiance that remains, when the ray exits the volume, is assigned to a pixel in the rendered image. Illuminating a volume with a background intensity exclusively results in a quite flat impression and another light source is often desired.

The amount of light that reaches each particle from a light source can be estimated

\begin{figure}[h]
\centering
\begin{tabular}{cccc}
  a) Absorption & b) Out-scattering & c) Emission & d) In-scattering \\
\end{tabular}
\caption{Illustration of how light can interact with a particle. Light arriving at a particle can either be absorbed or scattered in any direction. Light can also be emitted by the particle or collected from other directions.}
\end{figure}
2.1. LIGHT INTERACTIONS

with various levels of complexity. A simple approach is to completely ignore the fact that light can be attenuated between the light source and each particle (figure 2.3b), which means that the intensity not is influenced by other particles in the volume. However, if attenuation of light is considered (figure 2.3c) then the light source can cast shadows, which can improve the visual quality greatly.

Figure 2.3: Illustration of different levels of complexity for approximating the illumination in a volume.

a) Volume illuminated with a background intensity. Only emission and absorption is considered.

b) Local single scattering. The radiance from a point light source that reaches a particle along the view-aligned ray is not affected by other particles in the volume.

c) Single scattering with attenuation. In contrast to (b) the intensities can be affected by other particles that the beams intersect with.

d) Multiple scattering. All the light interactions in figure 2.2 are considered, which means that all the particles in the volume can affect the intensity that reaches a particle along the view-dependant ray.

Figure 2.3: Illustration of different levels of complexity for approximating the illumination in a volume.
The most complicated light interaction to simulate is multiple scattering (figure 2.3d). The incident light to each point in the volume can then arrive from many directions. These calculations are very computationally demanding and so constrained illumination models, excluding multiple scattering effects, are often used.

2.2 The Volume Rendering Integral

The light transport from one point to another can be estimated using the volume rendering integral. Figure 2.4 illustrates a ray passing through two points, \( s_0 \) and \( s_1 \). If no light is absorbed then the intensity, \( I \), is constant along the path (left image), while the initial intensity, observed at \( s_0 \), decreases if absorption is considered (right image). The optical depth, \( \tau \), can be used to measure the amount of radiance that is absorbed along the path between the points (equation 2.1).

\[
\tau(s_0, s_1) = \int_{s_0}^{s_1} \kappa(s) ds
\]  

(2.1)

\( \kappa \) is the absorption coefficient which indicates how quickly the intensity decreases in the medium. A large optical depth corresponds to an opaque content. The actual transparency, \( T \), of the medium between the two points can be computed using the optical depth

\[
T(s_0, s_1) = e^{-\tau(s_0, s_1)} = e^{-\int_{s_0}^{s_1} \kappa(s) ds}
\]  

(2.2)

Consequently, multiplying the initial intensity, \( I(s_0) \), with the transparency, \( T(s_0, s_1) \), results in the reduced intensity, \( I(s_1) \), at point \( s_1 \) (see equation 2.3).

\[
I(s_1) = I(s_0) \cdot T(s_0, s_1)
\]  

(2.3)

Only the absorbed radiance is considered in this equation. However, some of the points between \( s_0 \) and \( s_1 \) might also emit light, which must be considered in order to simulate realistic illumination. Figure 2.5 provides an example of how intermediate emission affects

Figure 2.4: Illustration of how the intensity is affected along a ray in a participating medium if attenuation of light is considered (right) or not (left).
2.2. THE VOLUME RENDERING INTEGRAL

Figure 2.5: The attenuated intensity is increased if light is emitted anywhere along the ray. The source term \( g(s) \) defines how light is contributed at a point \( s \).

the radiance along a ray. \( g(s) \) is the source term that describes the emissive contribution at a point \( s \). Equation 2.3 must therefore be extended:

\[
I(s_1) = I(s_0) \cdot T(s_0, s_1) + \int_{s_0}^{s_1} g(s) \cdot T(s, s_1)
\] (2.4)

This equation is the classical volume rendering integral \([\text{Max95}]\) which is used in ray-casting (illustrated in figure 1.7) to estimate the intensity of a pixel in a rendered image. It is possible to define the source term, \( g(s) \), in various ways. The realism increases when using a complex \( g(s) \), however, it is common to only use a local approximation in order to evaluate the illumination at high speed. This type of shading is described in section 2.2.1.

2.2.1 Local Estimation of \( g(s) \)

A simple approach to reach interactive frame rates when simulating illumination is to approximate the source term, \( g(s) \), in the volume rendering integral using local estimates. The main difference between local and global illumination is the awareness of other objects in the volume. The information available at each point, for instance the position, surface normal, viewing direction and direction to the light source is sufficient for evaluating the local light condition. Consequently, the limitation of such illumination is that shadowing effects are not possible since there is no awareness of the existence of other voxels in the volume that might occlude the light.

In local approximations, like the Blinn-Phong shading model \([\text{Bli77}]\), the source term \( g(s) \) is extended with three additional components: diffuse and specular reflections, and ambient light, as in equation 2.5.

\[
g(s) = A(s) + D(s) + S(s)
\] (2.5)
Ambient light simulates both direct and indirect illumination, which means light arriving directly from a light source and light reflected by all the other surfaces in the scene. An ambient contribution is vital in medical visualization where the aim is to reveal structures. Objects that are totally occluded from the light source would otherwise appear black to the viewer and important information could be hidden. Thus, it is very complex to compute and a rough approximation, using an ambient reflection coefficient $k_a$, simulating a uniform light distribution is often employed. The ambient intensity is thereby simply $A(s) = k_a c(s)$ where $c(s)$ is the color at sample $s$. The left teapot in figure 1.2 is illuminated using only an ambient reflection coefficient.

If a surface is glossy then incident light rays will be reflected just as by a mirror. Accordingly, the direction of the incoming and outgoing light has the same angle to the surface normal. This light contribution is defined in the $S(s)$ intensity and can be written as

$$S(s) = k_s(N \cdot H) p c(s)$$

(2.6)

where $H$ is the half-vector between the viewer and the light source vectors, $V$ and $L$ respectively. $N$ is a surface normal which is given by the normalized gradient vector. The specular coefficient $k_s$ specifies the amount of specular reflection at sample point $s$ and $p$ is a measure of shininess.

Matte surfaces do not reflect all the intensity from an incident light ray in one direction, unlike specular reflection. Instead, the intensity is spread over a number of outgoing light rays in various directions. A completely matte surface reflects the incident light equally in all directions. Diffuse reflections contribute to the source term with $D(s)$ and can be written as

$$D(s) = k_d \max(L \cdot N, 0) c(s)$$

(2.7)

and an example of the resulting shading that diffuse illumination can provide is given in the right image in figure 1.2. The diffuse coefficient $k_d$ defines the amount of diffuse reflection at sample point $s$.

When the normal, $N$, is used to describe the light condition at a point it is assumed that the point is associated with a surface. However, medical volumes often contain large homogeneous regions and have soft transitions between different tissue types. Since gradients are only well-defined at sharp transitions between different scalar values, gradient-based shading only gives a good approximation of light on, for example, bones and the skin.

Another problem is that medical volumes sometimes can be very noisy. Noisy data results in unreliable gradients which lead to misleading rather than helpful illumination since the noise probably is amplified. More realistic lighting models are required in order to enhance the shape perception of fuzzy volumes. Nevertheless, gradient-based shading is one of the most commonly used methods for illumination estimation in medical applications today.
2.2. THE VOLUME RENDERING INTEGRAL

2.2.2 Numerical Approximation

In order to solve the volume rendering integral by sampling along a cast ray, a numerical approximation is needed. The change in intensity along a ray is continuous but must be discretized in order to be solved numerically, and the ray is therefore divided into \( n \) equidistant segments. The color, \( c_i \), with the corresponding transparency, \( T_i \), for a segment, \( i \), can be written as

\[
c_i = \int_{s_{i-1}}^{s_i} g(s) \cdot T(s, s_i), \quad T_i = T(s_{i-1}, s_i)
\]

(2.8)

With this notation the final intensity, \( I(s_n) \), can be computed as follows

\[
I(s_n) = I(s_{n-1})T_n + c_n
\]

\[
= (I(s_{n-2})T_{n-1} + c_{n-1})T_n + c_n
\]

\[
= ...
\]

and \( I(s_n) \) can, hence, be written as

\[
I(s_n) = \sum_{i=0}^{n} c_i \prod_{j=i+1}^{n} T_j, \quad \text{with} \quad c_0 = I(s_0)
\]

(2.9)

The opposite property of transparency is opacity, \( \alpha \), which means that \( T = (1 - \alpha) \). Color and opacity along a ray can be evaluated iteratively, in a front-to-back order, according to equation 2.10. \( c'_i \) and \( \alpha'_i \) are the accumulated color and opacity, respectively. Computations in the reverse order, back-to-front, are also possible but are not considered in this thesis.

\[
c'_i = c'_{i-1} + (1 - \alpha'_{i-1}) \cdot c_i \cdot \alpha_i
\]

\[
\alpha'_i = \alpha'_{i-1} + (1 - \alpha'_{i-1}) \cdot \alpha_i
\]

(2.10)

An efficient approach to the solution of equations 2.10 is to perform the computations directly on the GPU, as described next.

2.2.3 GPU Raycasting

One of the challenges in medical visualization, is to reach interactive frame rates. An efficient approach to the solution of the volume rendering integral is therefore needed which can be achieved by utilizing the Graphics Processing Unit (GPU). Different shader programs can be used to program the GPU. A vertex shader can be used to perform operations on each vertex, for example setting a color or changing the position in space, while a fragment shader can be used to perform per-pixel effects.
2. ASPECTS OF VOLUMETRIC LIGHTING

Krüger and Westerman proposed a technique for implementing raycasting on the GPU \cite{KW03}. Their method is a multi-pass approach which casts rays for each fragment through the volume until a termination condition is fulfilled. An improvement of this technique, provided by Stegmaier et al. \cite{SSKE05}, exploits the support for loops within a fragment program. In their approach the entire volume rendering integral can be solved for a fragment in a single pass. Performing raycasting on the GPU is beneficial for a number of reasons. Due to the GPUs highly parallel structure it is possible to compute several rays simultaneously and thereby reach high performance. Additionally it is possible to improve performance by introducing empty space skipping and early ray termination \cite{KW03}.

A basic DVR pipeline, from data acquisition to a rendered volume, was illustrated in figure 1.3. Estimation of the light transport in a participating medium is dependent on the TF settings and must therefore be performed when the TF is defined. Re-estimation of the illumination is hence required if the TF is modified. A common approach is to compute the illumination in a pre-processing step, as shown in the expanded pipeline in figure 2.6. Some methods proposed in previous research, for example Kniss et al. \cite{KPH03}, incorporate the estimation of illumination in the final rendering step. An overview of methods that approximates global illumination is provided next.

2.3 Global Illumination Approximations

In global illumination it is possible to include very complex estimations of light interactions between particles in the participating medium. Emission, absorption and scattering are parameters that can influence the contributed radiance to a point. This allows for self shadowing, which improves the perception of spatial relations greatly and the resulting illumination becomes much more realistic compared with local illumination. A vast amount of research has been conducted for efficient estimation of global illumination, especially for polygonal data. However, the main focus of this thesis is methods for illumination in participating media. Short reviews of previously presented methods are provided in this section. Hadwiger et al. \cite{HLSR08} provide a survey of advanced illumination techniques for GPU volume raycasting for further reading.
2.3. **GLOBAL ILLUMINATION APPROXIMATIONS**

2.3.1 **Volumetric Shadows and Scattering Effects**

The most straightforward approach to the evaluation of volumetric shadows caused by one light source, excluding scattering effects, is to estimate the attenuated radiance along rays cast from each voxel to the light source. However, this approach is slow and it can be difficult to reach interactive frame rates. One approach to improve the rendering speed is to pre-compute shadows and store the values on a regular 3D grid that corresponds to the structure of the volume, as proposed by Behrens and Ratering [BR98]. A similar method, deep shadow maps [LV00, HKS06], store a representation of the volumetric occlusion in light space. As long as the light condition is static these methods are efficient. However, if the light source is moved or an additional light source is added then a time consuming recomputation is needed.

Kniss et al. [KPH+03] have proposed a method to speed up the estimations using a technique called half-angle texture slicing. With this approach shadows are evaluated in parallel with the rendering in image space using 2D buffers. This method handles color bleeding and forward scattering. A similar approach presented by Desgranges et al. [DEP05] reduces hard shadow edges and enhances the appearance of translucency by integrating dilation of light. Schott [SPH+09] extends [KPH+03] with directional occlusion shading effects using a backward-peaked cone phase function. The resulting illumination that these methods can simulate is visually appealing but has two main limitations. Slice based volume rendering is required and only one light source is supported.

Ropinski et al. [RDRS10] evaluate light propagation slice by slice, similar to Kniss et al. [KPH+03], but along the major volume axis. The illumination values are stored in an additional volume, similar to Behrens and Ratering [BR98]. With this approach it is possible to use any volume rendering technique. To avoid popping artifacts when changing the light position, the estimated illumination must be blended with a light propagation evaluated for an additional volume axis.

Simulating scattering effects in global illumination is computational demanding and Qiu et al. [QXF+07] have presented a method using Face Centered Cubic lattice for improved sampling efficiency. General scattering effects can also be achieved using Monte Carlo raycasting [Sal07]. However, it is difficult to reach interactive frame rates with these methods.

Promising approaches using spherical harmonics have recently been presented for simulation of global illumination. For instance, Lindemann and Ropinski [LR10] have introduced a technique that simulates reflectance and scattering by incorporating complex material functions. Kronander et al. [KJL+11] provide an efficient spherical harmonics approach that supports dynamic lighting environments. A multiresolution grid is considered in their local visibility estimations but not for global visibility.

2.3.2 **Ambient Occlusion**

The approximation of ambient light, described in section 2.2.1, is very gross since it assumes uniform light. A better estimate of the ambient contribution is given with ambient
Figure 2.7: Rays are cast from a point $p$ to a surrounding hemisphere $\Omega$ to evaluate ambient occlusion. Rays that intersect with other geometry do not contribute with any ambient light to point $p$.

Ambient occlusion (AO) which was first introduced by Zhukov et al. [ZIK98]. The aim of their work was to illuminate polygonal geometry and achieve more accurate ambient light than provided by Blinn-Phong shading while avoiding expensive methods like radiosity.

The basic concept of AO, illustrated in figure 2.7, is to estimate the incident light to a point, $p$, by casting rays in all directions of a hemisphere to that surrounds the surface to which $p$ belongs. If the cast rays intersect with any occluding geometry then the radiance is reduced. An object illuminated with AO obtains a matte surface and appears as if it was illuminated on an overcast day.

Mathematically formulated, the ambient light, $A_L$, at point $p$ can be computed by integrating the visibility function, $V_{p,\omega}$, over the hemisphere $\Omega$, as in equation 2.11, where $N$ is the surface normal.

$$A_L = \frac{1}{\pi} \int_{\Omega} V_{p,\omega}(N \cdot \omega) d\omega$$  \hspace{1cm} \text{(2.11)}$$

This approach is referred to as an “all-or-nothing method” since the visibility function is set to zero if the the point is occluded by any geometry in the direction of $\omega$. Otherwise, if light reaches point $p$, in the direction of $\omega$, it is defined to be one.

Vicinity shading, presented by Stewart [Ste03], is the first approach to using AO in volume rendering. He extends the all-or-nothing method by only blocking incident light that is occluded by a surface of a higher density than the density of point $p$. This approach only consider occlusions in the vicinity of each point, restricted by a defined radius of the surrounding hemisphere. Vicinity shading has been followed by a number of models for approximating ambient occlusion for iso-surfaces. Ruiz et al. [RBV+08] extended Stewart’s...
approach using the distance to the first occluder as an obscurrence factor. Desgranges and Engel [DE07] have proposed a less expensive method using a linear combination of opacity volumes. The occlusion is approximated by blurring the opacity in the surrounding of each voxel using different filter sizes. Penner and Mitchell [PM08] compute AO for iso-surfaces with a representation of statistical information about neighboring voxels. A disadvantage with this method is that transparent materials are not supported. Beason [BGB+06] presented a method that supports translucency of iso-surfaces by pre-computing illumination using a path-tracer. However, this method only supports static lighting.

AO results in a major improvement in the simulation of realistic ambient light. However, the methods presented above are only applicable for iso-surfaces. Only a few approaches have been presented for approximations of AO in DVR. Ropinski et al. [RMSD+08] proposed a method using pre-processed local histograms which describe the distribution of intensities that surrounds each voxel. This information, which is independent of rendering parameters, is used together with the user-defined transfer function to find the color of each voxel during rendering. Global illumination effects are not obtained since only local histograms are used. A disadvantage with this method is that the pre-processing step is time-consuming. AO effects can also be simulated using spherical harmonics [KJL+11].
Improving Volumetric Illumination

Several approaches for the simulation of volumetric illumination have been presented in previous research, as mentioned in section 2. However, one aspect that is often avoided is that medical volumes are often large datasets. Consequently, an illumination method must be designed so that the computations can be performed on a large set of data and still reach interactive frame rates. The requirement of perceiving medical volumes correctly and the increasing resolution of the acquired data imply very high demands on the utilized techniques. The main objective of the research contributions in the appended papers is to simultaneously fulfill these two demands by using advanced illumination estimations in combination with a multiresolution data structure. This chapter gives an overview of the methods that are the contributions of this thesis. A list of the four main aspects in volumetric illumination that the approaches deal with is provided below:

- Ambient lighting with local occlusion that simulates diffuse illumination
- Global illumination with cast shadows from a point light source
- Local in-scattering effects
- Luminous materials in the interior of a volume

The idea behind the methods in papers I-IV originate from ambient occlusion which was described in section 2.3.2. How ambient occlusion can be extended to be utilized in DVR is presented next.

3.1 Ambient Occlusion for Direct Volume Rendering

When estimating ambient occlusion for DVR the surrounding hemisphere, which is sufficient for a surface, must be extended to a whole sphere in order to surround each voxel. Light can arrive from all directions, even from the interior of an object. Furthermore, consider a voxel located inside an object, as shown in figure 3.1. If equation 2.11 is employed then the voxel will always turn black since all the incoming light is terminated due to occlusion (left image). For instance, if the volume consists of translucent objects then
When estimating ambient occlusion, with an all-or-nothing visibility function, for a voxel inside a volume (left image) all the rays will be terminated, resulting in an unlit, black voxel. Instead, if semi-transparency is considered and the attenuation of light is estimated along cast rays (right image), then the voxel will be partially illuminated.

 Ambient occlusion, with an all-or-nothing visibility function, causes deep dark shadows even though the obstacles are semi-transparent. A model for realistic illumination should instead estimate the attenuated light integrated along each ray, similar to the approach in raycasting (section 1.2.3 and 2.2). The amount of intensity that reaches point \( p \) would then be in the range \([0,1]\), instead of either zero or one.

Solving the volume rendering integral for multiple rays originating at each voxel in a volume results in cumbersome computations. Medical volumes are often large and the rays must be sampled quite densely in order to cover all occluding obstacles. Approximations are therefore needed in order to reach interactive frame rates. A simplified model that yields visually appealing results, called Local Ambient Occlusion (LAO), is presented next. This method is explained in detail in paper I and III.

### 3.2 Local Ambient Occlusion

The concept of Local Ambient Occlusion (LAO) is to reduce the complexity of AO by constraining the radius of the surrounding sphere, which results in a local diffuse shadowing effect. The aim of the visualization process influences the need for different illumination properties. In some medical visualizations it can be a drawback to include shadows cast from distant objects since important regions can become too dark. Instead, it can be...
3.2. LOCAL AMBIENT OCCLUSION

beneficial to only consider shadows from the vicinity of each voxel. These shadows are often adequate in order to comprehend spatial relations among anatomical structures. The idea of restricting the neighborhood is similar to the approach of Stewart [Ste03] in which vicinity shadows is approximated for iso-surfaces.

In LAO the incident light at a point, \( p \), is approximated by summing the incident light from a number of rays that build up a surrounding sphere. This is formulated in equation 3.1 for one ray direction, \( k \). An initial offset, \( a \), is introduced in order to prevent the voxel, \( x \), from occluding itself. The radius, \( R_\Omega \), of the surrounding sphere, \( \Omega \), is constrained by the user. Since the rays are shorter with a decreased radius less sample points are needed. A great speed-up can thereby be gained, while yielding high quality shadows from obstacles in the vicinity.

\[
A^L_k(x) = \int_a^{R_\Omega} g_{AL}(s) \cdot e^{-\int_t^s \tau(t)dt} ds
\]  

(3.1)

The volume rendering integral (equation 2.4) includes a background intensity. This intensity is excluded when estimating LAO. Instead, the contributed light is restricted to the emittance of each sample point, defined in the source term \( g_{AL} \), which is described next.

3.2.1 Light Contribution

The distribution of light within the spherical neighborhood, \( \Omega \), must be known when integrating the radiance incident to a voxel. In ambient occlusion, formulated as in equation 2.11, the radiance at the hemisphere is equal to one. A comparable approach, when extending ambient occlusion for volumetric estimations, would be to set the source term, \( g_{AL} \), to a Dirac delta, \( \delta \). The source term can then be written as in equation 3.2, where \( s \) is each location along the ray. With this approach light will only be contributed at the boundary of the sphere.

\[
g_{AL}(s) = \delta(s - R_\Omega)
\]  

(3.2)

This method implies that no radiance is emitted anywhere along the ray. On top of that, the resulting shadows becomes very sharp (see figure 3.2c) and hence a significant number of rays are required, to create a smooth appearance.

A more realistic result is achieved if the emitted light is evenly distributed along the ray, as proposed in paper I and III. The resulting shadows then become softer and less sensitive to the number of rays used (see figure 3.2d). A uniform contribution of light is incorporated in the source term, \( g_{AL} \), expressed as

\[
g_{AL}(s) = \frac{1}{R_\Omega - a}
\]  

(3.3)
3. IMPROVING VOLUMETRIC ILLUMINATION

a) Light is only contributed at the sphere boundary

b) Light is contributed at each sample point

c) LAO evaluated with settings in (a).

d) LAO evaluated with settings in (b).

Figure 3.2: When LAO is estimated with a light contribution constrained to the sphere boundary, ((a) and (c)), the shadowing effect becomes rather sharp. The estimations performed for (d) with a continuous light contribution as illustrated in (b), the shadows become softer and thereby also less sensitive to the number of rays used when evaluating the LAO. © Eurographics Association 2007 applies to (c) and (d).

A numerical solution for the LAO intensity can be formulated as in equation 3.4. The ray is divided into M segments and each sample point, \( m \), has an emitted radiance of \( \frac{1}{M} \).

However, to simplify the equation the light contribution, which is only a scaling factor, is moved outside the sum. The source term, \( g_m \), which is a discretization of \( g_{AL}(s) \), is therefore equal to one.

\[
A_{AL}^L(x) = \frac{1}{M} \sum_{m=1}^{M} g_m \prod_{i=1}^{m-1} T_i \quad (3.4)
\]

The ambient term, \( A_{AL}^L(x) \), only includes estimations of intensity integrated from one ray. To obtain the final intensity of a voxel the estimated intensities, from all the incoming rays, must be summarized. This leads to equation 3.5, where \( K \) is the number of rays cast from the voxel to the sphere. A weight, \( w_k \), is introduced in order to influence the amount of initial light arriving from different directions. For uniform ambient light the weight is equal to one.

\[
A^L(x) = \frac{1}{K} \sum_{k=1}^{K} w_k A_{AL}^L(x) \quad (3.5)
\]
### 3.3. LUMINOUS ILLUMINATION EFFECTS

To further enhance the perception of details in a volume and increase the ability of separating different tissue types it can be beneficial to allow some scalar values to be luminous. If important tissue types are highlighted it might be easier to detect them quickly. Appealing luminous effects can be achieved with a very limited addition to LAO. A description of this approach is provided next.
A colored light emission, \( c_E \), has been introduced in paper I in order to define the luminosity of each scalar value. Similar to the approach described in section 3.2.2, an additional TF is used to steer the mapping. To include colored emission in DVR the ambient part of the source term, \( g(s) \), can be extended according to equation 3.6.

\[
A(s) = A^I(s) \cdot c(s) + c_E(s)  
\]  

(3.6)

However, the luminous effect caused by equation 3.6 is limited since an emissive voxel cannot affect its surroundings. This can, instead, be achieved by extending LAO to also consider the emissive contribution defined by \( c_E(s) \). The source term used in LAO thus becomes:

\[
g_A(x) = \frac{1}{R_\text{a} - a} + c_E(s)  
\]  

(3.7)

This results in single light scattering which allows the luminosity of neighboring voxels to affect both the intensity and the color of voxel x. Figure 3.4 shows a comparison of the different effects that are achieved by restricting the luminosity to be incorporated in either the raycasting pass or the LAO estimations. An image using \( c_E(s) \) in both passes is also provided, which results in an additional brightness of the emissive effect.
3.3. LUMINOUS ILLUMINATION EFFECTS

3.3.1 Using Light as an Information Carrier

An fMRI volume contains scalar values that describes the probability of brain activity in a voxel, as described in section 1.2.1. The information in this type of volume can be a great aid when planning surgical removal of brain tumors. With the knowledge of where important regions are located it is possible to remove as much of a tumor as possible and at the same time avoid causing deteriorated functions in the patient’s brain. However, it is complicated to navigate in fMRI volumes and it is necessary to combine the visualization with anatomical information.

Paper IV proposes a method that combines fMRI volumes with MRI anatomy volumes using LAO and luminous effects. With this approach the fMRI signal regulates the luminosity of a voxel and the fMRI regions appear as light sources that illuminate the anatomy. In the setup used in paper IV, the anatomy volume is generated in an initial scan while the fMRI signals are acquired in real-time. The registration between the two volumes is performed by maximizing the mutual information $VI_{95}$.

The illumination has to be re-estimated each time the fMRI signal is updated. However, it is not necessary to recompute the ambient lighting since the anatomy is static, and this light condition changes only if the TF for $c(s)$ is modified. Since the source term, $g_A$, in equation 3.7, is a sum of both ambient light and emission ($g_A + g_AE$), it is possible to separate the computations for $A_L$ and $A_E$. The results can then be stored in different textures for easy access during raycasting, when equation 3.8 is solved. Only recomputation of $A_E$ is thereby needed when the fMRI signal changes (see figure 3.5).

$$A(s) = (A^L(s) + A^E(s)) \cdot c(s) + c_E(s)$$ (3.8)

In some situations it can be sufficient to only include $c_E(s)$ in the raycasting pass, which means that $A^E$ is ignored, as illustrated in figure 3.6. This improves the speed of the computations. Another possible approach for speeding up the computations is to mask possible fMRI regions and ignore empty spaces when estimating the illumination on the GPU.
If the density of the anatomy is too high then it can be difficult to see important activity regions in the interior of the brain. An additional variant of combining fMRI and anatomy is to reduce the opacity of samples, along the raycasting rays, that do not contain any luminosity. A detailed description of this variant is provided in paper IV.

### 3.4 Global Light Transport

The use of distant shadows can be beneficial in some medical applications since interaction with a single light source that causes cast shadows from distant objects can be an aid when figuring out the object hierarchy. The possibility of perceiving depth in a rendered image might thereby be improved.
3.4. GLOBAL LIGHT TRANSPORT

Figure 3.7: A two-dimensional example of four voxels (orange) casting rays towards a light source. The four rays intersects with almost the same voxels and it is desirable to reuse some parts of the light integrations in order to speed up the calculations.

Figure 3.8: If the ray in the illustrations is sampled too sparsely then the occluding green voxels might be missed. The incident intensity to point $s_0$ differs greatly when the step length $\Delta s$ (left) is used compared to $\Delta s'$ (right), to integrate the light contribution. The opacities at the sample points are interpolated from the neighboring voxels.

If distant shadows are evaluated by integrating the light attenuation along cast rays, from each voxel to the light source, then a large number of sample points are needed. In order to find all occluding objects each ray must be sampled quite densely. However, the occlusion along large parts of these rays are computed several times since multiple rays are overlapping, as illustrated in figure 3.7. One of the methods that this thesis contributes is an approach that reuses light integrations in order to speed up the estimation of the global light transport. The theory behind this method, called Local Piecewise Integration, is presented next. This approach is described in detail in paper II.

3.4.1 Local Piecewise Integration

A ray divided into $k$ segments, with a distance of $\Delta s$, is illustrated in figure 3.8. The absorbed light along the ray can be computed with the following approximation

$$I(s_0) = I_0 \cdot \prod_{n=0}^{k} T(s_n, s_{n+1})$$

(3.9)

where the light source has the intensity $I_0$. Normally, $T$ is computed with $T = (1 - \alpha)$ where the opacity, $\alpha$, is derived by tri-linear interpolation using the opacities of the neighboring voxels. If $\Delta s$ is large then an interpolated $\alpha$ gives a poor estimation of the occlusion.
since obstacles can easily be missed, as illustrated in the left image in figure 3.8. If $T$ of each segment is integrated separately, as an initial step, and these piecewise integrations are used instead of the interpolated opacities then more obstacles can be found. Figure 3.9 illustrates how pieces of the integration along a one-dimensional ray can be combined. The opacity of a segment can be obtained with equation 3.10, where $N$ is the number of samples taken along the segment. $\alpha(n_i)$ is the opacity at a sample point $n_i$.

$$\alpha_{lp}(s_n) = 1 - \prod_{i=0}^{N}(1 - \alpha(n_i))$$ (3.10)

Computing the opacity along one ray with sparse sampling in combination with pre-integrated opacities seems unnecessary compared to using dense sampling directly. An obvious drawback is that extra storage space is needed for the local piecewise opacities. However, if the pre-integrated opacities can be reused for multiple integrals then a great speed-up can be obtained. The complete occlusion from $s_0$ to $s_n$ thus becomes:

$$\alpha_g(s_0) = 1 - \prod_{i=0}^{k}(1 - \alpha_{lp}(s_i))$$ (3.11)

Using local piecewise integration in the one-dimensional case (figure 3.9) is simple and does not introduce any errors. However, when computing $\alpha_g$ in three-dimensions then trilinear interpolation is needed to find the best approximation of $\alpha_{lp}$ for each sample point. The stored values for the eight neighboring voxels are thus weighted depending on their distance to the sample point. Small errors can potentially arise if a neighboring voxel is occluded by an object that does not occlude the current sample point.

The intermediate results for both $\alpha_{lp}$ and $\alpha_g$ needs to be stored in 3D textures. However, it is often desirable to minimize the amount of storage space. By combining the intensities in $\alpha_{lp}$ and $\alpha_g$, when evaluating the direct light contribution, $I_d$, to each voxel, it is possible to reduce the size of the $\alpha_g$-texture and still remain a nice shadowing effect. As illustrated in figure 3.7 neighboring rays coincide closer to the light source. The difference between two neighboring rays are therefore mainly in the vicinity of the voxel. Consequently, if $\alpha_{lp}$ is used for the first part of the integration then $\alpha_g$ can be used for the remaining part. This is illustrated in figure 3.10 and mathematically formulated in equation 3.12. Using this combination of $\alpha_{lp}$ and $\alpha_g$ results in visually appealing shadows even though a reduced size of the $\alpha_g$-texture is used.

$$I_d(s_0) = I_0 \cdot (1 - \alpha_g(s_1)) \cdot (1 - \alpha_{lp}(s_0))$$ (3.12)

Figure 3.11 provide with examples of representing each block of $16^3$, $8^3$, $4^3$ and $2^3$ voxels, in the $\alpha_g$-texture, with only one voxel. The quality of the light integrations, stored in $\alpha_g$, are not influenced by this texture reduction.

The described method only takes the absorbed light into account. How to expand this method to include first order scattering effects is presented in 3.4.2.
3.4. GLOBAL LIGHT TRANSPORT

Figure 3.9: The opacities along the ray in (a) are shown in the chart in (b). The ray is divided into equidistant segments and for each segment the opacity is integrated (c). The resulting opacities for each segment are shown in (d). The opacity along the whole ray is estimated in (e). Only a sparse sampling frequency, red dots, is needed when using the piecewise opacities in (d). In a one-dimensional case the result is just as accurate as a traditional integration of the opacity, which is shown with a dashed blue line in (e). The axes in the charts (b)-(e) represents intensity and ray points for the y-axis and x-axis, respectively.
3. IMPROVING VOLUMETRIC ILLUMINATION

Figure 3.10: An illustration of how the direct light contribution, $I_d$, to a voxel can be estimated using a combination of $\alpha_{lp}$ and $\alpha_g$. The blue grid represents the reduced size of the $\alpha_g$-texture. The first part of the integration is taken from the stored $\alpha_{lp}$-texture and the remaining part is a tri-linear interpolation of the neighboring values in the $\alpha_g$-texture.

Figure 3.11: If the size of the $\alpha_g$ texture is reduced too much then jagged edges and band artifacts appear, as in (a) and (b). The shadows obtained when representing each block of $4^3$ voxels by a single voxel, as in (c), result in softer edges. However, these shadows are probably sufficient for most applications. © Eurographics Association 2008.
3.4.2 First Order Scattering Effects

Approximating first order scattering to a voxel, presented in paper II, is similar to the approach of estimating LAO. The only difference is that the emission at a sample point is obtained from the resulting intensities of equation 3.12. Only scattering effects from the immediate vicinity are considered, since the incident rays are restricted by a surrounding sphere $\Omega$. The in-scattering contribution thus becomes

$$I_s(s_0) = \int_{\Omega} \varphi(\omega) \int_{s_0}^{R_0} I_d(r) \cdot e^{-\int_{s_0}^{R_0} \tau(t) dt} dr d\omega$$

(3.13)

A numerical evaluation of this integral is given in equation 3.14, where $M$ is the number of samples taken along each ray, $j$, and $J$ is the number of ray directions. An optional phase function, $\varphi$, is introduced in paper II to allow directional weighting. An isotropic phase function is used to weight the incident light from all directions equally. The resulting $I_s$ is the intensity that illuminates each sample point during raycasting.

$$I_s = \sum_{j=0}^{J} \varphi_j \sum_{m=0}^{M} I_d(s_m) \prod_{i=0}^{m-1}(1 - \alpha(s_i))$$

(3.14)

3.5 Pipeline Processing

The development of medical imaging modalities is rapidly improving, resulting in higher volume resolution. This leads to memory problems and costly rendering computations. A number of multiresolution techniques have been developed to deal with these large datasets. Flat Multiresolution Blocking [LLYM04] is an efficient multiresolution technique which plays an important role in the methods presented in this thesis. A short introduction to flat multiresolution blocking is provided in section 3.5.1 followed by a description of how this technique can be utilized to improve the performance of illumination calculations. Sections 3.5.3 and 3.5.4 describes the pipelines used for LAO and illumination computed with piecewise integrations.

All the methods presented, in papers I-IV, require that the illumination condition at each voxel in the volume is evaluated and stored in intensity maps. These intensities are then easily accessed during the image synthesis when solving the volume rendering integral. A fast approach to perform computations for all voxels in a volume is to map all the slices, one by one, as a rendering target of a frame buffer object (FBO). A quad is rendered over the entire frame buffer and each pixel, together with the current slice z-offset, maps to one voxel in the 3D texture. For each pixel in the FBO, a fragment program is initiated that performs illumination estimations.
3. IMPROVING VOLUMETRIC ILLUMINATION

3.5.1 Flat Multiresolution Blocking

This section describes the technique of flat multiresolution blocking, which has been developed by Ljung et al. [LLYM04] to handle large scale data.

Consider an axial slice of a brain, illustrated in figure 1.4. Some parts of the slice contain large regions of air (black) while other regions are highly detailed, for example the blood vessels inside the skull. Using storage space for all the voxels containing air is unnecessary. The approach in flat multiresolution blocking is to divide the volume into small blocks of $16^3$ voxels. Each block is examined and a level-of-detail, LOD, is chosen depending on the block content. The choice is based on minimizing the color error in the TF domain in combination with a memory limit constraint. The blocks are packed in a new volume according to their LOD, not according to their spatial location. Figure 3.12 shows an illustration of how the blocks can be packed, in a two-dimensional case. A block that only contains air does not contribute to the volume visually and is therefore not represented in the packed volume.

3.5.2 Multiresolution Illumination Estimations

If a volume employs the flat multiresolution data structure then fewer passes are needed when traversing the volume by mapping slices to the FBO. A lot of computations are saved with this approach since empty voxels are not represented in this data structure and large regions of low detail level are represented by only a few voxels. However, there is one drawback: blocks of the volume are packed in the data structure according to their LOD and not arranged according to their spatial locations. Consequently, two additional textures that contain forward and reverse indexing are needed in order to move between the packed coordinates and volume coordinates (see figure 3.13). This approach requires some additional texture look-ups but it is, nevertheless, in most cases more efficient to perform the computations directly in the packed data structure since the number of voxels for which the light condition must be evaluated can be greatly reduced.

How to create a forward indexing texture is described in Ljung et al. [LLY06a]. The reverse indexing texture is more complicated since a voxel either can represent a whole block or just one voxel. If the reverse indexing would allow one block to be represented by one voxel then the resulting texture would be of the same size as the packed texture. Blocks with the resolution levels $1^3$ and $2^3$ are, in the methods presented in this thesis, therefore ignored, which allows for a texture reduction by a factor of 64. Consequently, empty blocks are discarded and other blocks are restricted to a minimum LOD resolution of $4^3$.

Another disadvantage of packing volume blocks according to their LOD is that linear interpolation at points near block boundaries can lead to artifacts. A method for interblock interpolation has been presented by Ljung et al. [LLY06a] but using this approach would decrease the interactive speed of the illumination estimations, since a large number of texture look ups are required. Instead, sample points can be clamped to a distance, $\vartheta$, from the block boundary so that interpolation is only performed within each block, as described in [LLY06a].
3.5. PIPELINE PROCESSING

Figure 3.12: Illustration of how an image (a) can be packed (b) when using the data structure in flat multiresolution blocking. Blocks of $16^2$ pixels are examined and a LOD is chosen, marked in (c), (d) and (e), depending on the block content and are packed in (b). Empty blocks (f) are discarded. The image in (a) can be stored at almost half of it’s original size with this method with negligible loss in image quality.

Figure 3.13: Scalar data in a slice of the volume (V) and the packed texture (P) is shown in (a) and (b), respectively. One indexing texture is needed to jump from volume coordinates to packed texture coordinates and an other indexing texture is needed to jump in the reverse order.
3. IMPROVING VOLUMETRIC ILLUMINATION

3.5.3 LAO Pipeline

Estimation of incident light from multiple directions is performed in both LAO (paper I, III and IV) and the in-scattering part of the global light transport (paper II). Integrating the complete contribution to a voxel from multiple directions can be time consuming depending on the number of rays used to build up the surrounding sphere $\Omega$. In order to maintain interactive speed it is possible to perform the computations incrementally, with one ray per rendered frame, until all the rays are processed, resulting in a progressively refined illumination effect. The pipeline is illustrated in figure 3.14. The final intensity texture, used when rendering the volume, contains RGBA quadruplets with the emissive color and the ambient occlusion of each voxel.

Blending of the new intensity, $A_k$, and the previously stored intensity, $A^{k-1}$ of a voxel is performed with the OpenGL blend operation. The accumulated intensity, $A^k$, after $k$ directions can be expressed as

$$A^k = \frac{1}{k} \cdot A_k + \left(1 - \frac{1}{k}\right) \cdot A^{k-1} \quad (3.15)$$

where the initial intensity, $A^0$, is zero. It is possible to dynamically configure the number of rays, $K$, to use in LAO.

It is important that the rays used to collect the incident light contribution to a voxel from multiple directions, are uniformly distributed. Otherwise the incoming radiance from a particular direction will be over-represented. This is achieved by utilizing platonic solids which have equal lines, angles and surfaces. A tetrahedron, icosahedron or octahedron are subdivided to different levels depending on the desired number of rays, as in Tarini et al. [TCM06]. The resulting direction vectors are stored in a texture together with a directional weight, which is 1 for a uniform light distribution.

Figure 3.14: The pipeline for evaluation of the incoming intensities from one direction is iterated until all the rays are processed.
3.5. PIPELINE PROCESSING

3.5.4 Pipeline for Illumination Estimations with Piecewise Integration

The pipeline for the global illumination with piecewise integration, in paper II, can be divided into three parts, resulting in the 3D textures $\alpha_{lp}$, $\alpha_g$ and $I_s$ (figure 3.15-3.17). To fill these textures each slice must be rendered to an FBO, as described in section 3.5. The process of finding $\alpha_{lp}$ is similar to the pipeline in LAO. A ray is cast in the direction of the light source and sample points are taken along the ray (illustrated in figure 3.15).

Figure 3.15: A short ray, in the direction of the light source, is sampled for each voxel in the volume. The resulting values, $\alpha_{lp}$, are stored in a 3D texture.

Figure 3.16: The opacities, $\alpha_g$, along the paths from each voxel to the light source are computed by sampling in the $\alpha_{lp}$ texture and are stored in an additional 3D texture.

Figure 3.17: The intensity of one incident ray is evaluated using the resulting $\alpha_g$ and $\alpha_{lp}$ at sample points along the ray. The final intensity from one direction is blended together with previously estimated intensities. The volume is rendered after each ray update which gives a progressively refined impression of the illumination.
3. IMPROVING VOLUMETRIC ILLUMINATION

Figure 3.18: Pre-integration of segments originating in empty blocks is not performed since those not are represented in the flat blocking multiresolution data structure. If sample points are taken in empty blocks (left image) when evaluating the global opacity $\alpha_g$, then obstacles can easily be missed. A jump to the next non-empty block is therefore performed when a sample is taken in an empty block (right image). The grid of the blocks is marked with red dots.

A longer ray is cast from each voxel towards the light source when estimating $\alpha_g$. The samples taken along the ray are obtained from the previously created $\alpha_{lp}$ texture, as shown in figure 3.16. Obstacles can easily be missed when computing $\alpha_g$ if samples are taken in empty blocks. These blocks are not represented in the data structure used and $\alpha_{lp}$ is thereby not defined in those regions. The solution for this problem is to jump to the next non-empty block, as illustrated in figure 3.18.

Having both $\alpha_g$ and $\alpha_{lp}$ it is possible to perform the last step, figure 3.17, which estimates $I_s$. The final intensity is incrementally improved until all the in-scattered rays are computed, which is performed progressively, interleaved with the raycasting pass. Blending of the new intensity and the stored intensity of a voxel is performed using equation 3.15, just as for the accumulated intensity in LAO. The direction of the light source is computed first, since it gives the best approximation. The number of directions can be modified by the user and the directions are taken from a texture, as described in 3.5.3.

As mentioned in section 3.5.2, clamping is used when interpolation is performed in the data structure of flat multiresolution blocking. However, clamping is not possible for $\alpha_g$ if it is reduced by a large factor. $\alpha_g$ is therefore estimated in a regular linear volume instead, which makes it possible to employ linear interpolation without causing artifacts.

3.6 Sampling

Additional speed-ups can be achieved by modifying the sampling density. Regions stored with a low LOD do not need to be sampled at the same frequency as regions with a high LOD. An illustration of how the step lengths along LAO rays can be modified depending on the LOD of the center voxel is provided in figure 3.19a. This method uses a fixed sampling rate for the whole sphere and so might lose accuracy if the rays intersect with blocks of higher LOD. An additional sampling approach, presented in paper III, adapts the step lengths depending on the LOD of the underlying volume data, as illustrated in figure 3.19b. This method gives a more accurate result compared to the fixed sampling
3.6. SAMPLING

a) Fixed sampling  

b) Adaptive sampling

Figure 3.19: The fixed step length in the left image is based on the LOD at the center of each sphere. In the right image the step length is adapted to the LOD at every sample position. © IEEE 2010.

rate but can be slower to evaluate. One reason for this is that the parallelism is decreased since the integrations for neighboring voxels use different numbers of sample points.

When changing the step length it is also important to change the amount of emitted light along the rays in LAO in order to maintain a consistent light contribution. The intensity must thereby be based on the step size, $\Delta s$, as in equation 3.16, which is illustrated in figure 3.20. $l_i$ is the light contribution for sample point $i$ and $\Delta s$ is the sample’s extent.

$$l_i = \frac{\Delta s_i}{R_\Omega - a} \quad \Delta s_i = \begin{cases} \frac{\Delta s_i}{2} & \text{if } i = 1 \\ \frac{\Delta s_{i-1} + \Delta s_i}{2} & \text{if } 1 < i < m \\ \frac{\Delta s_{i-1}}{2} + \Delta s_m & \text{if } i = m \end{cases} \quad (3.16)$$

The extent of the samples also affects the emissive light defined in $c_E(s)$, which can be adjusted with

$$c'_E(s) = \frac{\Delta s}{\Delta B} c_E(s) \quad (3.17)$$

where $\Delta B$ is the base sampling density. Also the opacities must be modified when the step length is changed. This is performed with an opacity correction formula

$$\alpha' = 1.0 - (1.0 - \alpha) \frac{\Delta s}{\Delta B} \quad (3.18)$$
3. IMPROVING VOLUMETRIC ILLUMINATION

Figure 3.20: Illustration of the varying step lengths, $\Delta i$, along a ray which are adapted according to the LOD at each sample position, as in figure 3.19b. $\Delta s$ refers to a sample’s extent. © IEEE 2010.

3.7 Illumination Quality

The quality of illumination estimates can be affected if approximations are used in the light calculations. One way to evaluate the quality of a rendering is to compare the image with an “accurate” image. It is, however, difficult to obtain accurate images. The comparisons in this section are instead performed against images that are considered to be good enough. A perceptually adapted color error in the CIE 1976 L*$u^*v^*$ color space (CIELUV) [LLYM04, Fa98] is used to measure the pixel-wise difference between the two images. This measure has a Just Noticeable Difference (JND) at 1.0. An overall error, which is the mean square error, is denoted with $\Delta E_{RMS}$. This measure does not describe the distribution of errors and individual pixels might differ significantly. An additional error measure $\Delta E_6$ is therefore also provided in order to illustrate the fraction of the pixels that have a difference that is greater than 6.0, as in [LLYM04].

3.7.1 LAO Accuracy

If too few rays are used when estimating LAO there is a risk of missing important obstacles. In figure 3.21 illumination estimations with 64 and 8 directions are compared with an estimation performed with 128 directions, which is considered to be accurate enough. Corresponding error images for 8 and 64 rays are provided in the lower row. $\Delta E_{RMS} = 1.7$ and $\Delta E_{RMS} = 0.4$ for 8 and 64 rays, respectively. The errors in some pixels in figure 3.21b and 3.21c are rather high but the LAO appearance is maintained and it is still easy to comprehend structures in the volume.

The same type of error measurement has been performed for an image computed with different levels of data reduction. The images are compared to an image with a data reduction of 8:1. Error images that illustrate the pixel-wise differences when using 16:1 and 93:1 are provided in figure 3.22. The errors that appear in these images depend, not only on the illumination estimations, but some errors arise during rendering since the same data reduction is used in the raycasting pass. A data reduction of 93:1 is very large and errors of $\Delta E_{RMS} = 6.0$ and $\Delta E_6 = 18\%$ are thereby quite small. When using a data reduction of 16:1 the errors are reduced to $\Delta E_{RMS} = 3.8$ and $\Delta E_6 = 4.7\%$. 

\[ \Delta \]
3.7. **ILLUMINATION QUALITY**

Figure 3.21: The LAO computations used for the images in the upper row have been estimated using 128, 64 and 8 rays, respectively. Color mapped error images (lower row) show the pixel-wise errors with respect to the 128 ray image. © IEEE 2010

Figure 3.22: The pixel-wise errors that arise when increasing the data reduction from 8:1 to 16:1 and 93:1 are illustrated with color-mapped error images in the lower row. © IEEE 2010
3. IMPROVING VOLUMETRIC ILLUMINATION

3.7.2 Local Piecewise Integration Accuracy

Errors can be introduced when computing global opacity with piecewise integrations since interpolation is employed. The errors caused by using local piecewise integration instead of traditionally sampled rays are illustrated in figure 3.23. It took almost ten times longer to estimate the illumination for a volume containing the scan of a lion without piecewise integration. A pixel-wise comparison of the two images results in the error measures $\Delta E_{RMS} = 0.19$ and $\Delta E_6 = 6\%$. The greatest errors appear at sharp edges and thin structures. However, it is difficult for a human eye to detect these errors when comparing the two images.

3.8 Performance

Performance has been measured using a standard PC equipped with an Nvidia GeForce 8800 Ultra graphics board with 768 MB of graphics texture memory for LAO (section 3.8.1) and piecewise integration (section 3.8.3) and an Nvidia GTX 285 graphics card with 1GB of graphics texture memory for the concurrent volume visualization (section 3.8.2).

3.8.1 LAO Performance

Only estimations for one direction is needed for an incremental update of LAO illumination. Obviously, the time needed for updating one ray is independent of the number of rays into which the surrounding sphere $\Omega$ is discretized. Measurements of performance for one LAO direction are provided in the chart in figure 3.24. Two lines are shown in the chart, one for fixed sampling rate (red) and one for adaptive sampling rate (blue). As can be seen, the LAO estimations are slightly super-linear in the level of data reduction. There are two
3.8. PERFORMANCE

Figure 3.24: Chart of measured performance for LAO versus data reduction for a dataset of $512^3$ voxels. Both fixed sampling (red dotted line) and adaptive sampling (blue solid line) is shown. 15 samples are taken along each ray for the highest resolution in the fixed sampling rate and $R_Ω = 16$. © IEEE 2010.

main reasons for this. First, a higher data reduction implies that there are a decreased number of voxels for which to compute occlusion. Second, the sample rate is dependent on the LOD of the underlying volume data and since a high data reduction increases the number of low resolution blocks longer steps can be taken which leads to an improved speed of the calculations.

The performance with adaptive sampling is rarely much faster than using fixed sampling. In many cases, as in figure 3.24, it is noticeably worse. The reason for this is the mixture of LOD blocks, which depends on the current TF settings. If a voxel need to process a very different sampling rate than its neighboring voxels then the parallelism of the computations is decreased, leading to reduced performance. However, an adaptive sampling rate does result in fewer artifacts in the LAO estimations compared to the fixed sampling rate.

3.8.2 Performance for Concurrent Volume Visualization

Table 3.1 provides measurements for updating the ambient light and emission when using fMRI to illuminate an anatomy volume. To compute ambient light for the whole anatomy volume ($512 \times 512 \times 256$ voxels) took between 5.38 and 22.02 seconds using 32 to 128 rays,
respectively. These computations can be performed for one ray per frame (approximately 0.17 seconds/ray) interleaved with raycasting of the volume, for an incrementally improved illumination. However, since the anatomy volume is static and the illumination is invariant with respect to the light and view direction, the ambient lighting need only be estimated once, unless the TF is changed. The time for this update is therefore not so crucial. Fast updates of the emissive contribution (see table 3.1) is, on the other hand, more important since the fMRI volume (64x64x22 voxels) is acquired in real-time. An optimized calculation, using masks on the GPU as mentioned in section 3.3.1, are in these measurements only 0.21 for 32 rays, which is 4 times faster than a full estimation.

<table>
<thead>
<tr>
<th>LAO calculation</th>
<th>anatomy(s)</th>
<th>fMRI signal(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>full calculation</td>
</tr>
<tr>
<td>Initialization</td>
<td>32 rays</td>
<td>5.38</td>
</tr>
<tr>
<td></td>
<td>64 rays</td>
<td>10.98</td>
</tr>
<tr>
<td></td>
<td>128 rays</td>
<td>22.02</td>
</tr>
<tr>
<td>fMRI signal change</td>
<td>32 rays</td>
<td>0.82</td>
</tr>
<tr>
<td></td>
<td>64 rays</td>
<td>1.54</td>
</tr>
<tr>
<td></td>
<td>128 rays</td>
<td>3.02</td>
</tr>
</tbody>
</table>

Table 3.1: Performance measures for updating the ambient light and emission, for an anatomy and fMRI volume, using LAO.

### 3.8.3 Performance for Illumination Estimated with Piecewise Integration

There are three main factors that affect the performance of illumination computed with piecewise integration. These factors are the size of the $\alpha_g$-texture, the length of the piecewise segments, and the data reduction in the multiresolution data management. Performance changes due to these factors are provided in table 3.2. The left table shows the effect of increasing the data reduction and the right table provides measurements using varying sizes of the $\alpha_g$-texture versus different lengths of the piecewise segments.

With these settings, the global light integration is updated in 284-552 ms, depending on the $\alpha_g$ size, for a piecewise segment length of 8 voxels. An additional 62-233 ms are needed for first order scattering incremental refinements, depending on the applied data reduction. When the estimations for the illumination are complete then the volume can be rendered at high frame rates of 15-22 frames per second (fps). Only one texture look-up is required to determine the intensity at each sample point during raycasting, unless any light settings are modified.
3.8. PERFORMANCE

Table 3.2: These tables provide performance measures for different levels of illumination updates versus data reduction (left) and piecewise segment length (right). (A) includes estimations for producing the textures $\alpha_p$ and $\alpha_g$. (B) corresponds to the estimations of $I_s$ and (C) to the volume raycasting step. All measures are provided in milliseconds (ms). The piecewise segments in the left table are 8 voxels long and the data reduction in the right table is 8.9:1. The size of the volume, being rendered in a 1024x1024 window, is 512³.

<table>
<thead>
<tr>
<th>Data reduction</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.9:1</td>
<td>284</td>
<td>552</td>
<td>233</td>
</tr>
<tr>
<td>14.8:1</td>
<td>178</td>
<td>515</td>
<td>145</td>
</tr>
<tr>
<td>22.1:1</td>
<td>121</td>
<td>403</td>
<td>96</td>
</tr>
<tr>
<td>35.2:1</td>
<td>81</td>
<td>365</td>
<td>62</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Segment length (voxels)</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>32³</td>
</tr>
<tr>
<td>4</td>
<td>261</td>
</tr>
<tr>
<td>8</td>
<td>284</td>
</tr>
<tr>
<td>16</td>
<td>331</td>
</tr>
<tr>
<td>32</td>
<td>436</td>
</tr>
</tbody>
</table>
3. IMPROVING VOLUMETRIC ILLUMINATION
Conclusions

Illumination is a very important part of medical visualization in order to avoid a flat impression of the anatomy in an acquired volume. As shown in chapter 2, realistic illumination is associated with intense computation and approximations are currently necessary in order to reach interactive frame rates.

This thesis contributes efficient methods for volumetric illumination which are described in detail in the appended papers, with an overview of the methods provided in chapter 3. This chapter gives a short summary of the contributions followed by an evaluation of how the challenges, listed in section 1.3, have been approached. Finally, a short discussion of future research in this area is provided in the end of the chapter.

4.1 Summary of Contributions

The three main illumination models that this thesis contributes with are summarized below.

- A method for local effects of omni-directional illumination which simulates the light condition that appears on a cloudy day. This has been achieved by integrating the incident light from a discretized number of rays in a bounding sphere defined by the user.
- A method for simulation of global light transport and local in-scattering effects which are approximated using piecewise integrations. This technique is up to ten times faster than a brute force estimation that integrates the incident light along cast rays from each voxel to the light source.
- A method that highlights important features of a volume, for instance fMRI, with luminous materials. The user can steer the color and intensity of emissive tissues with an additional TF.

How these methods can handle the challenges presented in section 1.3 is described next.

4.2 Approached Challenges

To develop illumination methods for medical applications is associated with a number of challenges. This section provide a short discussion of how the challenges listed in section 1.3 have been addressed in the methods that this thesis contributes with.
Figure 4.1: Different levels of noise have been applied to a volume consisting of the letter T. The upper row, illuminated with diffuse illumination, is much more sensitive to noise in the data compared with the lower row, illuminated with LAO. © IEEE 2010.

The main challenge in volumetric illumination is to find suitable approximations that can be evaluated interactively while revealing important information in the volume. The methods that are presented in this thesis reach promising frame rates, as noted in section 3.8, both for support of a single light source and for omni-directional illumination.

The challenge of estimating these illumination models for large scale data has been approached in two ways. First, by using a multiresolution data structure that allows a graceful data reduction, both for the illumination calculations and the rendering pass. Second, by exploiting the possibility of running parallel computations on the GPU.

A concern, mentioned in section 1.3, about using shadows in medical visualization is that important tissues might be hidden due to absence of light. When looking for abnormalities in a volume it is important that no regions are fully shadowed. The feature of allowing all sample points in LAO to emit light minimizes the problem of fully shadowed tissues in the body, which otherwise could be difficult to examine.

Other challenges, described in section 1.3, concerns the quality of medical data. For instance, MRI volumes can be rather noisy. The lower row in figure 4.1 illustrates how LAO can handle volumes with varying levels of noise. As a comparison, the upper row in figure 4.1 shows renderings of the same volumes using gradient based shading. The shape of the T is better preserved in the lower row, while the T in the upper row almost disappears in the noise.
4.2. APPROACHED CHALLENGES

Figure 4.2: It is easier to perceive the density of the skull in the left image, which is illuminated with LAO, compared with the right image which is illuminated with diffuse shading. © IEEE 2010.

Another positive effect with LAO is that it can be easier to perceive the density of different tissues. An example of this is given in figure 4.2. It is almost possible to see through some parts of the skull when illuminated with LAO, compared to the gradient based rendering. However, some high frequency structures are more apparent on the skull in the right image and a combination of the two methods might be beneficial in some situations, unless the volume is noisy. An example of how LAO and diffuse shading can be combined is provided in figure 4.3. This figure also shows the positive effect of allowing a foreign material to be emissive, where it is easier to observe the emissive bullet fragments in figure 4.3b and c compared to the non-emissive fragments in figure 4.3a.

Figures 4.1 and 4.2 only compare LAO with diffuse illumination. Improvements have been proposed for volumetric illumination in previous research, as mentioned in chapter 2, that result in more realistic illumination than gradient based shading. These methods are, of course, more comparable to the methods presented in this thesis. However, a comparison with diffuse illumination is still interesting since it is one of the most frequently used illumination methods in medical applications today.

A comparison of gradient based shading and illumination estimated with piecewise integrations is provided in figure 4.4. A clip plane is used in this visualization, which is frequently used in medical applications in order to remove unnecessary structures that hide important tissues. When using gradient based shading the gradients are often pre-
4. CONCLUSIONS

a) Diffuse shading b) LAO c) Diffuse shading + LAO

Figure 4.3: The effect of very high frequency details can be reduced when using LAO compared to diffuse shading. If the volume is not noisy then a combination of the two illumination methods can be favorable. This figure also illustrate that the bullet fragments become more apparent when being luminous, as in (b), compared to (a). Close-ups are provided in the lower row. © Eurographics Association 2008.

computed and stored in a texture for fast access during rendering. If a part of the volume is removed with a clip plane then the gradients suddenly are inaccurate which can lead to misleading illumination at the clip surface. Incorrect illumination at clip surfaces are not an issue in the methods presented in the appended papers since illumination is updated according to the current TF settings and invisible voxels are not represented in the multiresolution data structure.

4.3 Future Research

This thesis has contributed efficient methods for volumetric illumination that reveal different aspects of a volume. Further research can certainly be carried out to find even more efficient approximations and future hardware development can help to speed up the illumination calculations further. Nevertheless, the most important part in future research is to perform further evaluations.
4.3. FUTURE RESEARCH

Figure 4.4: The shape and depth of the blood vessels are more prominent in (a) compared to in (b). In this volume a stent has been inserted into the aorta. A clip plane is used to reveal additional structures inside the heart.

Approaches for gradient-based illumination, like the Blinn-Phong shading model [Bli77] (described in section 2.2.1) have been available since the seventies. Since this type of illumination has been used for such a long time it is not questioned to the same extent as more recently presented techniques. Even though gradient-based shading has clear drawbacks, as illustrated in figures 4.1, 4.2 and 4.4, it is still one of the most commonly used method in medical visualizations today.

Some perceptual studies have been performed for volumetric illumination methods in previous research, for instance [LR11]. However, it is necessary to perform deeper clinical evaluations, with medical experts that utilizes 3D representations with different illumination models as diagnostic tool. The new generation of illumination methods, compared to gradient based methods, must be used more extensively at hospitals in the near future.

Finally, it would be interesting to evaluate the use of the presented illumination methods for additional types of medical data, for example ultra sound data. LAO is advantageous for illumination of noisy datasets, as shown in figure 4.1 and a promising noise test with error measurements is provided in paper III. If noisy ultra sound data is illuminated with LAO then a great improvement in the perception of shapes might be possible.
Bibliography


[SSKE05] Simon Stegmaier, Magnus Strengert, Thomas Klein, and Thomas Ertl. A simple and flexible volume rendering framework for graphics-hardware-based


