

Live Fetoscopic Visualization of 4D Ultrasound Data

DISSERTATION

zur Erlangung des akademischen Grades

Doktor der technischen Wissenschaften

eingereicht von

Andrej Varchola

Matrikelnummer 0728357

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung: Ao.Univ.-Prof. Dipl.-Ing. Dr.techn. Eduard Gröller

Diese Dissertation haben begutachtet:

(Ao.Univ.-Prof. Dipl.-Ing.
Dr.techn. Eduard Gröller)

(Univ.-Doz. Dipl.-Ing. Dr.techn.
Miloš Šrámek)

Wien, 27.09.2012

(Andrej Varchola)

Live Fetoscopic Visualization of 4D Ultrasound Data

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der technischen Wissenschaften

by

Andrej Varchola

Registration Number 0728357

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Ao.Univ.-Prof. Dipl.-Ing. Dr.techn. Eduard Gröller

The dissertation has been reviewed by:

(Ao.Univ.-Prof. Dipl.-Ing.
Dr.techn. Eduard Gröller)

(Univ.-Doz. Dipl.-Ing. Dr.techn.
Miloš Šrámek)

Wien, 27.09.2012

(Andrej Varchola)

Erklärung zur Verfassung der Arbeit

Andrej Varchola
Franzensbrückenstr. 13/22, 1020 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum)

(Unterschrift Verfasser)

Acknowledgements

This thesis was shaped and influenced by many people. I thank everyone for their support and help. I am especially grateful to my supervisor Meister Eduard Gröller for his guidance during last three years of my doctoral studies. I would also like to express my gratitude to Stefan Bruckner, for all discussions that were essential for making decisions during the development of the methods which are discussed in this thesis. This work is a result of a collaboration with GE Healthcare (Kretztechnik, Zipf, Austria). It could not have been completed without support of domain experts that were actively involved in the development of all presented ideas and achievements. I especially thank to Gerald Schröcker and Daniel Buckton. I also like thank all clinical experts from GE Healthcare, especially Marcello Tassinari who helped me with the evaluation of results presented in this work. Many appreciation goes to all of my colleagues at the Institute of Computer Graphics and Algorithms at the Vienna University of Technology for a productive environment. I am also thankful to students that I was supervising, in particular Johannes Novotny, Michael Seydl, Daniel Fischl. Furthermore, I thank my former colleagues from the Comission for Scientific Visualization of the Austrian Academy of Sciences. Special thanks to Miloš Šrámek, who introduced me to the exciting field of medical visualization. Special thanks also to Leonid Dimitrov, who engaged me in many scientific discussions and helped me also with the careful proofreading of this thesis. My gratitude goes also to many friends and family members who supported me during the past years of my studies.

Abstract

Ultrasound (US) imaging is due to its real-time character, low cost, non-invasive nature, high availability, and many other factors, considered a standard diagnostic procedure during pregnancy. The quality of diagnostics depends on many factors, including scanning protocol, data characteristics and visualization algorithms. In this work, several problems of ultrasound data visualization for obstetric ultrasound imaging are discussed and addressed.

The capability of ultrasound scanners is growing and modern ultrasound devices produce large amounts of data that have to be processed in real-time. An ultrasound imaging system is in a broad sense a pipeline of several operations and visualization algorithms. Individual algorithms are usually organized in modules that separately process the data. In order to achieve the required level of detail and high quality images with the visualization pipeline, we had to address the flow of large amounts of data on modern computer hardware with limited capacity. We developed a novel architecture of visualization pipeline for ultrasound imaging. This visualization pipeline combines several algorithms, which are described in this work, into the integrated system. In the context of this pipeline, we advocate *slice-based streaming* as a possible approach for the large data flow problem.

Live examination of the moving fetus from ultrasound data is a challenging task which requires extensive knowledge of the fetal anatomy and a proficient operation of the ultrasound machine. The fetus is typically occluded by structures which hamper the view in 3D rendered images. We developed a novel method of visualizing the human fetus for prenatal sonography from 3D/4D ultrasound data. It is a fully automatic method that can recognize and render the fetus without occlusion, where the highest priority is to achieve an unobstructed view of the fetal face. Our *smart visibility method for prenatal ultrasound* is based on a ray-analysis performed within image-based direct volume rendering (DVR). It automatically calculates a clipping surface that removes the uninteresting structures and uncovers the interesting structures of the fetal anatomy behind. The method is able to work with the data streamed on-the-fly from the ultrasound transducer and to visualize a temporal sequence of reconstructed ultrasound data in real time. It has the potential to minimize the interaction of the operator and to improve the comfort of patients by decreasing the investigation time. This can lead to an increased confidence in the prenatal diagnosis with 3D ultrasound and eventually decrease the costs of the investigation.

Ultrasound scanning is very popular among parents who are interested in the health condition of their fetus during pregnancy. Parents usually want to keep the ultrasound images as a memory for the future. Furthermore, convincing images are important for the confident communication of findings between clinicians and parents. Current ultrasound devices offer advanced imaging capabilities, but common visualization methods for volumetric data only provide limited visual

fidelity. The standard methods render only images with a plastic-like appearance which do not correspond to naturally looking fetuses. This is partly due to the dynamic and noisy nature of the data which limits the applicability of standard volume visualization techniques. In this thesis, we present a *fetoscopic rendering* method which aims to reproduce the quality of fetoscopic examinations (i.e., physical endoscopy of the uterus) from 4D sonography data. Based on the requirements of domain experts and the constraints of live ultrasound imaging, we developed a method for high-quality rendering of prenatal examinations. We employ a realistic illumination model which supports shadows, movable light sources, and realistic rendering of the human skin to provide an immersive experience for physicians and parents alike. Beyond aesthetic aspects, the resulting visualizations have also promising diagnostic applications. The presented fetoscopic rendering method has been successfully integrated in the state-of-the-art ultrasound imaging systems of GE Healthcare as *HDlive* imaging tool. It is daily used in many prenatal imaging centers around the world.

Kurzfassung

Die Ultraschallbildgebung ist aufgrund ihrer Echtzeit-Charakteristik, der niedrigen Kosten, dem nicht-invasiven Naturell, der hohen Verfügbarkeit und vieler weiterer Faktoren ein Standarddiagnoseverfahren während der Schwangerschaft. Die Qualität der Diagnose hängt von vielen Elementen ab, wie dem Abtastprotokoll, den Datenmerkmalen und den Visualisierungsalgorithmen. In dieser Arbeit werden verschiedene Probleme der Ultraschall-Datenvisualisierung in der geburtshilflichen Ultraschallbildgebung diskutiert und angesprochen.

Ultraschall-Scanner werden immer leistungsfähiger und moderne Ultraschallgeräte erzeugen große Datenmengen, die in Echtzeit bearbeitet werden müssen. Das bildgebende Verfahren durch Ultraschall ist im weiteren Sinne eine Pipeline von mehreren Operationen und Visualisierungsalgorithmen. Individuelle Algorithmen werden üblicherweise in Modulen geordnet, die separat Daten verarbeiten. Um das erforderliche Maß an Detail und qualitativ hochwertige Bilder mit der Visualisierungspipeline zu erreichen, befassen wir uns mit großen Datenflussmengen auf moderner Computer-Hardware mit begrenzter Kapazität. Wir entwickelten eine neuartige Architektur der Visualisierungspipeline für die Ultraschallbildgebung. Diese Visualisierungspipeline kombiniert mehrere Algorithmen, die in dieser Arbeit beschrieben werden, in das integrierte System. Als möglichen Ansatz für das große Datenflussproblem im Zusammenhang mit dieser Pipeline befürworten wir *schnittbasiertes Streaming*.

Die Ultraschall-Echtzeituntersuchung des beweglichen Fötus ist eine anspruchsvolle Aufgabe, die umfangreiche Kenntnisse in der fetalen Anatomie voraussetzt und eine kompetente Beherrschung des Ultraschallgeräts erfordert. Der Fötus wird typischerweise durch Strukturen verdeckt, die den Blick auf die generierten 3D Bilder behindern. Wir entwickelten ein neues Verfahren zur Visualisierung des menschlichen Fötus für die pränatale Sonographie aus 3D/4D Ultraschalldaten. Es ist ein vollautomatisches Verfahren, das den Fötus ohne Okklusion erkennen und wiedergeben kann, dabei ist die höchste Priorität, ein ungehinderten Blick auf das fetale Gesicht zu erzielen. Unsere *Smart-Visibility-Methode zum pränatalen Ultraschall* basiert auf einer Strahlenanalyse in der bildbasierten direkten Volumengrafik (DVR). Es berechnet automatisch eine Clipping-Oberfläche, die uninteressante Strukturen entfernt und dahinter interessante Strukturen der fetalen Anatomie aufdeckt. Die Methode kann mit den übertragenen Daten aus dem Ultraschallwandler arbeiten und eine zeitliche Abfolge von rekonstruierten Ultraschalldaten in Echtzeit visualisieren. Es hat das Potential, die Interaktion des Anwenders zu minimieren und den Komfort des Patienten durch Verringerung der Untersuchungszeit zu verbessern. Dies kann zu einem höheren Vertrauen in die pränatale Diagnostik mit 3D-Ultraschall führen und schließlich zu einer Verringerung der Untersuchungskosten.

Die Sonografie (Ultraschalluntersuchung) ist bei Eltern sehr beliebt, die an dem Gesundheitszustand ihres Fötus während der Schwangerschaft interessiert sind. Eltern wollen in der Regel die Ultraschallbilder als Erinnerung für die Zukunft behalten. Darüber hinaus sind überzeugende Bilder für die vertrauensvolle Kommunikation von Erkenntnissen zwischen Krankenhausärzten und Eltern wichtig. Aktuelle Ultraschallgeräte bieten erweiterte Bildgebungsfunktionen, jedoch verschaffen gewöhnliche Visualisierungsmethoden für volumetrische Daten nur begrenzt visuelle Glaubwürdigkeit. Die Standardmethoden erzeugen nur Bilder mit künstlichen Aussehen, die nicht den Föten in Natura entsprechen. Zum Teil ist dies bedingt durch die dynamische und rauschende Datenbeschaffenheit, die die Anwendbarkeit der Standardvolumen-Visualisierungstechniken begrenzt. In dieser Arbeit präsentieren wir eine *fetoskopische Rendering-Methode*, die die Qualität der fetoskopischen Untersuchungen (das heißt die physische Endoskopie der Gebärmutter) von 4D-Sonografie-Daten wiedergeben soll. Basierend auf den Anforderungen der Fachexperten und den Grenzen der Live-Ultraschallbildgebung, entwickelten wir ein Verfahren für die hochwertige Wiedergabe von pränatalen Untersuchungen. Wir verwenden ein realistisches Beleuchtungsmodell, das Schatten, bewegliche Lichtquellen und realistische Darstellung der menschlichen Haut unterstützt, um eine immersive Erfahrung für Ärzte und Eltern gleichermaßen zu bieten. Neben den ästhetischen Aspekten haben die resultierenden Visualisierungen auch vielversprechende diagnostische Anwendungen. Die vorgestellte fetoskopische Rendering-Methode wurde erfolgreich in den hochmodernen Ultraschallbildgebungssystemen von GE Healthcare als *HDlive* Bildbearbeitungswerkzeug integriert. Es wird täglich in vielen pränatalen Diagnosezentren auf der ganzen Welt verwendet.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Structure of the Thesis	5
1.3	Problem Statement	6
1.4	Aim of This Work	8
1.5	Methodological Approach	9
2	State of the Art	11
2.1	Medical Ultrasound Imaging and History	11
2.2	Ultrasound Visualization Pipeline	12
2.3	Basic Principles of Ultrasound Imaging	13
2.4	Ultrasound Data Acquisition and Reconstruction	15
2.5	Ultrasound Imaging Data Characteristics	18
2.6	Noise Reduction Methods	21
2.7	Classification Methods	22
2.8	Direct Volume Rendering	24
2.9	3D Ultrasound Imaging and Human Prenatal Anatomy	31
3	Streaming of Ultrasound Volume Data	35
3.1	Introduction	35
3.2	Related Work	37
3.3	Stream 3D Operations	38
3.4	Streaming Architecture of the Ultrasound Visualization Pipeline	41
3.5	Conclusion	44
4	Smart Visibility for Prenatal Ultrasound	45
4.1	Introduction	45
4.2	Related Work	48
4.3	The Algorithm of Smart Visibility for Prenatal Ultrasound	50
4.4	Implementation	63
4.5	Results	66
4.6	Conclusion	68

5 Fetoscopic Rendering	73
5.1 Introduction	73
5.2 Related Work	74
5.3 Goals of Live Fetoscopic Rendering	78
5.4 Fetoscopic Illumination Model	79
5.5 Implementation	102
5.6 Results and Discussion	106
5.7 Conclusion	115
6 Summary	119
Bibliography	121
Curriculum Vitae	133
Contact Information	133
Personal Details	133
Education	134
Employment History	134
Publications	135

Introduction

1.1 Motivation

The human body consists of many complex organs and it varies in many aspects among individuals. It has been an object of extensive studies in our history for a very long time. Human prenatal development is essential in the human reproduction process and the initial phase of the human life cycle. Prenatal development starts after the fertilization of the ovum by a sperm cell and it occurs inside the female's uterus. At this stage, a zygote is formed and divided to become an embryo. The embryo develops for eight weeks and then it becomes a fetus. The fetal period lasts thirty-eight weeks and it ends with the birth. After birth of the fully grown fetus, the infant is recognized as a person.

Insight into the prenatal development of the human body plays an important role in the history of life sciences. Leonardo da Vinci's drawings of the human fetus inside the womb from the 16th century can be considered as the foundations of modern anatomical illustrations (see Figure 1.1). His anatomical studies of the fetus originated from the post-mortem dissections of the pregnant uterus. The drawings in his notebook correctly depict the position of the fetus inside the womb. Together with the other pioneers from the Renaissance period, he helped to initiate a new scientific field which is nowadays called embryology.

The growing knowledge about human prenatal anatomy and development is progressively integrated by scholars into the recorded history with every new discovery. It is described and illustrated in standard textbooks and atlases of biology and medicine. Gray's Anatomy can be considered as an example of a classical textbook on the subject (see Figure 1.2). The work was initially published in 1858 and has continued to be revised and republished for more than 150 years [138].

A possibility to see real pictures of the human prenatal development *in vivo* became desirable also among the public. In 1965, a photographic book *A Child is Born* was issued [108]. It illustrates the development of the human embryo and fetus from conception to birth. The book was written in order to describe prenatal development and offer an advice on prenatal care. The high quality pictures in the book were acquired with conventional cameras with macro



Figure 1.1: 'Views of a Fetus in the Womb', a drawing by Leonardo da Vinci. Image is credited to the Royal Collection (c) 2012, Her Majesty Queen Elizabeth II.

lenses, endoscopes and scanning electron microscope technology (see Figure 1.3). Progress in the technology and science allowed to *look inside* the human body with various methods, e.g. radiography, endoscopy. However, to observe the human prenatal development *live*, without doing anything dangerous or invasive to the patients, became possible only with using ultrasound to image the human body.

Ultrasound imaging is based on the principle of echolocation. Several animals in the nature, such as dolphins and bats, use echolocation, also called biosonar, for navigation and hunting.

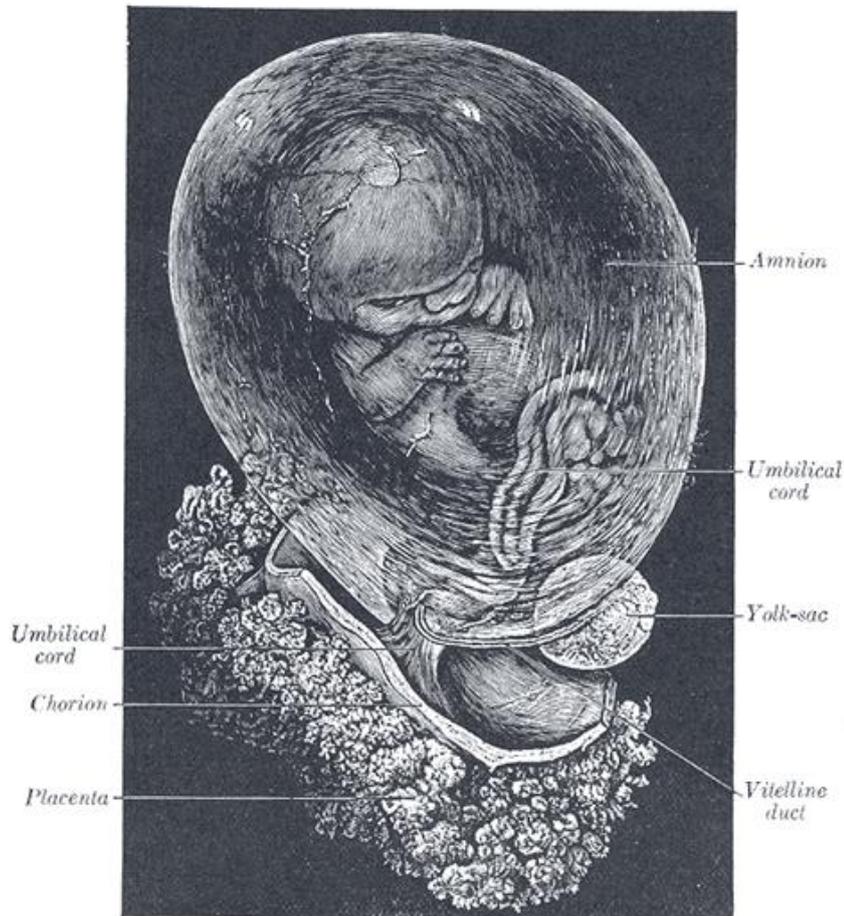


Figure 1.2: Illustration from Gray's Anatomy of the Human Body. Fetus of about eight weeks, enclosed in the amnion [153].

Echolocation abilities allow them to detect the location and to recognize the type of prey and other objects in their environment even in complete darkness. Their perceptual system emits ultrasound waves which are reflected from the objects in the environment. The returning echos are processed by the auditory and nervous system into a detailed image of their surroundings.

In ultrasound imaging, an acoustic hand-held probe, called transducer, is used to send ultra-high-frequency sound waves into the human body. Echoes of the sound waves, coming from reflections at internal structures, are acquired by the transducer and sent to the machine for reconstruction. The ultrasound machine reconstructs the acquired signal into images of internal structures of the human body. Medical ultrasound has developed into a medical imaging method that is applied on a daily basis in obstetrics and is well recognized by the public. It allows to visualize the embryo or fetus and to acquire various information about the health of a pregnant

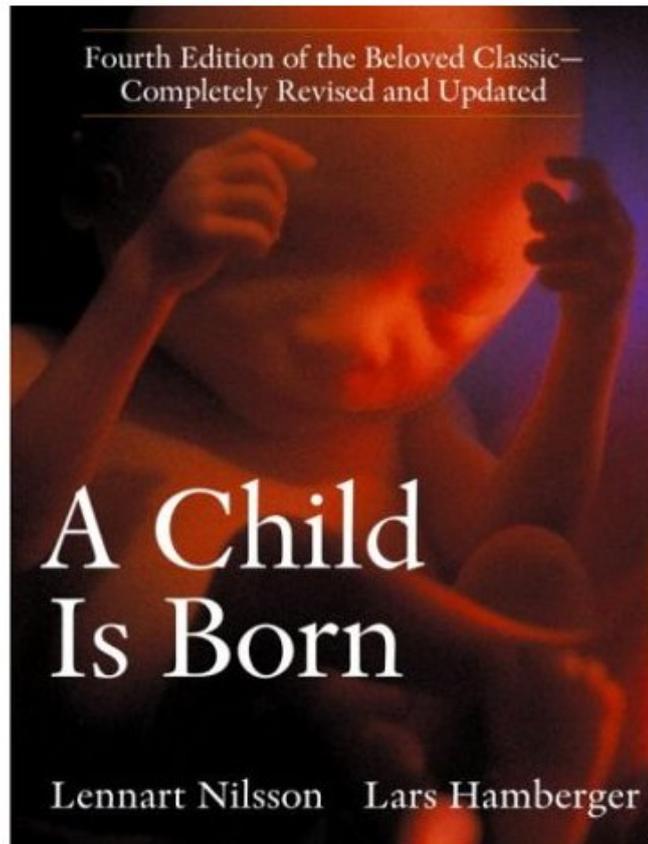


Figure 1.3: The cover of the book 'A Child is Born' with the photographic image of the fetus [108]. The book became the all-time best-selling illustrated book published.

woman. Ultrasound is due to its real-time character, low cost, non-invasive nature, high availability, and many other factors, considered a standard diagnostic procedure during pregnancy.

The quality of examination with ultrasound depends on many factors, including scanning protocol, data characteristics and visualization algorithm. Improvements in electronics and signal processing have a strong influence on the development of ultrasound imaging. Modern scanning devices allow to capture high-resolution data of the moving fetus in real-time (see Figure 1.4). The amount and the character of the scanned data opens new challenges for processing and visualization. Live examination of the moving fetus from ultrasound data is a difficult task which requires extensive knowledge of fetal anatomy and proficient operation of the ultrasound machine. The visual quality of images and clinical confidence have an important impact on the communication between clinicians and patients. Better visualization methods can lead to a better discussion after the examination results are available and have a potential to simplify the communication of findings.



Figure 1.4: Voluson E8 Expert. Modern ultrasound machine for female healthcare, including obstetrics, gynecology, maternal fetal medicine and assisted reproductive medicine. The image displayed on the monitor was rendered with the method described in this thesis [46].

1.2 Structure of the Thesis

In this thesis, we will focus on several visualization challenges with the objective to improve or to supplement the current methods of ultrasound imaging for prenatal development. The achievements presented in this thesis come from the cooperation between ultrasound domain experts, clinical experts and visualization experts from GE Healthcare (Kretztechnik, Zipf, Austria) and the Institute of Computer Graphics and Algorithms at the Vienna University of Technology.

The remaining part of this chapter provides the problem statement that was defined together with our collaboration partner. We also define our goals, which were the driving force during the work on the thesis, and provide a short outline of the methodology that was used in order to achieve the goals.

Chapter 2 provides the state-of-the-art in the field of ultrasound visualization. In the beginning, we shortly discuss the history of medical ultrasound technology. We introduce the visualization pipeline and in detail explain the function of each module that allows to transform

the sound signal into images. We discuss principles of acoustic signal acquisition and volume data reconstruction. Further steps of the pipeline, including noise reduction, classification and rendering are explained in more detail. The final section of the chapter covers the current possibilities for imaging prenatal anatomy with the ultrasound modality.

Chapters 3, 4 and 5 explain in detail novel visualization methods for ultrasound imaging. The results of each method were evaluated during the development cycle, in order to make them applicable in current ultrasound machines.

In Chapter 3 we discuss the streaming of volumetric data as a data flow concept which is used in our visualization pipeline. We present the main modules of our pipeline and show the data flow between them. Design decisions for individual modules are presented with respect to our streaming architecture.

Chapter 4 describes the novel method for smart visibility from prenatal ultrasound data. With this method it is easier to perform a live scan and to visualize the human fetus.

Chapter 5 covers the fetoscopic rendering from ultrasound data. It was developed in order to improve the image quality of current images.

Finally, the concluding chapter gives a summary of our research and achievements. The unique contributions are outlined together with limitations and possibilities of future work.

1.3 Problem Statement

In this work we address the problem of live visualization of 3D/4D ultrasound (US) data for obstetric ultrasound imaging. Examination of human prenatal development with US uses various rendering modalities. One of the methods which is most frequently utilized is called surface imaging. The method displays soft tissue information of the internal structures of the human body. Therefore, it is used for the evaluation of external surface anatomy, mostly the fetal face. In this mode, the embryo or fetus are displayed with a plastic-like appearance. The general goal of the thesis is to improve the US visualization of the human prenatal anatomy and other inner-body soft-tissue structures. Our intensive collaboration with ultrasound domain experts and clinicians during the research helped us to identify the following challenges and requirements more specifically:

- Large amount of ultrasound data generated on-the-fly:

Modern ultrasound probes produce 3D data in real-time with high resolution. The ultrasound machine reconstructs the acquired signal into images of internal structures and of the embryo or fetus. Ultrasound data has specific characteristics. It exhibits low contrast and low signal-to-noise ratio. Advanced visualization algorithms require a robust framework that can process and render the ultrasound data live, during the examination. Furthermore, the specification of the ultrasound machine is designed for reliable long-term application in a medical environment. It is challenging to design a visualization pipeline that can handle large amounts of data on-the-fly and support advanced visualization algorithms.

- Occlusion of the fetus by surrounding tissues:



Figure 1.5: The image illustrates a typical way of scanning the patient with the hand-held transducer. The sonographer has to operate the machine in order to acquire good images of the fetus [47].

Ultrasonography is a real-time medical imaging technique that is typically performed by a sonographer. A hand-held probe is positioned directly on the body part, covered with a water-based gel, and moved over the scanned area. The intermediate result of the examination is interactively displayed on the ultrasound screen in order to support navigation and to allow direct interpretation.

In prenatal ultrasound scanning of fetuses, the fetus is typically embedded in amniotic fluid. Ultrasound data contains interesting structures, i.e., the fetus and especially the face of the fetus. The fetus is embedded and surrounded by uninteresting tissues, i.e., amniotic fluid and other occluding structures (womb, placenta,...). In a 3D rendering these surrounding structures generate undesired occlusions. In practice it is difficult to locate the fetus occluded by surrounding tissue which can become even more complicated by the movement of the fetus during the scanning period. Currently the region of interest (ROI) has to be manually specified by the sonographer in order to visualize the fetus in the 3D rendering without any occluders. Figure 1.5 illustrates the typical way of scanning with an ultrasound transducer. The sonographer has to adjust controls on the ultrasound machine while scanning the tissues and trying to get a good view on the studied anatomy of the fetus. Therefore, developing a method that can automatically recognize and render the fetus without occlusion has the potential to minimize interactions of the clinical personnel during investigation.

- Plastic-like ultrasound images of the embryo and fetus:

There are several different rendering modes in 3D/4D obstetric ultrasound imaging. Nevertheless, the visual quality of the current ultrasound images of the embryo and the fetus is limited and rather plastic-like and does not produce enough realism. Especially when the generated images are compared with the images coming from fetoscopy. Fetoscopy can currently provide the most realistic images of the human prenatal development *in vivo*. It is an invasive endoscopic procedure in obstetrics that is performed during pregnancy to optically examine the fetus. A small camera with a light source is inserted through the abdominal wall and uterus into the amniotic cavity in order to directly screen the embryo or fetus. However, the procedure is, due to its invasive character, usually performed only when fetal surgery is necessary.

Prenatal images have a remarkable psychological value for the parents that undergo the ultrasound scan during pregnancy. Also clinicians, who perform an examination with ultrasound, need a special training for a confident evaluation of the current ultrasound images. They study the images acquired with endoscopy alongside with state-of-the-art ultrasound images to gain a clear understanding of the fetal anatomy. Developing a more realistic rendering method can lead to a better understanding of examination findings and increase the comfort for the patients. The new rendering mode has to be integrated into the latest generation of GE Healthcare imaging systems.

1.4 Aim of This Work

The purpose and the main goal of this work is to summarize our research in 3D/4D visualization of human prenatal development from ultrasound data. It started in order to develop effective novel approaches that are targeted to the current generation of ultrasound machines. The applicability of our work in clinical practice was a strong driving force and one of the goals of our work. In the text, we address the stated problems and requirements with focus on the following aims:

- Robust visualization pipeline for real-time rendering:

The visualization pipeline for 3D/4D ultrasound imaging has to be designed. The pipeline should be able to handle the required amount of data in real time. Visualization methods should be developed with respect to the architecture of the proposed pipeline. They should provide real-time performance on current hardware and require only minimal interaction of clinicians.

- Ultrasound smart-visibility algorithm for the fetus:

A novel method of visualizing the human fetus for prenatal sonography from 4D ultrasound data should be developed and tested. It should be a fully automatic method that can recognize and render the fetus without occlusion, where the highest priority is to achieve an unobstructed view of the fetal face. The method should be able to work interactively with the data streamed on-the-fly from the ultrasound probe and to visualize a temporal sequence of reconstructed ultrasound data in real-time. Real test cases with different categories (easy, medium, and difficult) are provided by GE Healthcare for analysis and

testing of the fetus visibility problem. The sophisticated algorithm should be able to detect and visualize the face of the fetus from ultrasound data where the face is covered by occluders in most of the test cases. The algorithm should minimize the interaction of the sonographer with the ultrasound machine.

- Fetoscopic rendering of the human prenatal anatomy:

It is required that fetal images from live rendering of ultrasound data have similar visual properties as photographic images coming from a real fetoscopy. Sophisticated algorithms of computer graphics and visualization can generate very convincing images of the human body for computer games, movies and illustration. Therefore, existing approaches should be analyzed and a feasible model should be proposed and implemented. One of the main goals of the thesis is to design a shading model that can achieve a convincing rendering of the human skin of the embryo or fetus from ultrasound data. The visual properties and realism of the images need to be further improved by additional perceptual cues like shadows or other advanced illumination effects. It is also important to customize the illumination model according to the requirements of domain experts. The novel fetoscopic rendering mode for ultrasound has to be integrated into the commercial US machine of GE Healthcare.

1.5 Methodological Approach

In accordance with the stated goals of our work, it is important to understand the way how the human prenatal development is examined with ultrasound. The understanding of the human prenatal anatomy itself is also an important aspect for the approaches used in this work and it will be shortly discussed in this thesis.

In general, the visualization approaches strongly depend on the character of the data which have to be transformed to images. Our case can be considered as a scientific visualization approach, because of the focus on data from a natural phenomenon, i.e., the human body. As the context of our application scenario is ultrasound in clinical environments, it can be classified also as medical visualization. Ultrasound scanning devices considered in our work generate 3D data in real-time. The data are reconstructed by the US machine into volumetric representations. Therefore our approach also belongs to the category of volume visualization.

Volume data from the human body can be produced by different imaging modalities. Each of them has its own specific characteristics and requires special considerations for visualization. Medical image data acquired with US scanning devices differ from data produced by modalities like CT or MRI in many ways. The main difference is caused by the character of the scanning process. One of the main advantages of US imaging and its importance in clinical environments is the real-time availability of the visualized body structures. Our methodological approach has to consider the character of the data produced by the US modality and today's ultrasound imaging quality standards in order to achieve the desired goals of this work.

There are several possible ways of how to address the stated problems and how to achieve satisfying solutions to our goals. In this work, we propose the following approaches:

- Streaming of ultrasound volume data

The ultrasound data is constantly acquired by the 4D ultrasound transducer. The data is processed in a streamed fashion that optimally utilizes resources of the ultrasound machine. The acquired raw ultrasound data is converted according to the scanning geometry of the transducer into a volumetric representation of a regular grid. In the next stage the data is filtered in order to improve the signal-to-noise ratio and to reduce artifacts before further processing and rendering of the data. Direct volume rendering (DVR) visualizes the converted data after filtering. Final corrections are applied to the image in the post-processing stage before it is displayed on the screen. The clinician can view images rendered from the original and the filtered version of the data.

- Automatic clipping surface

The automatic visibility method builds on image-based direct volume rendering (DVR). In image-based DVR for each pixel of the image a ray is traversed through the 3D ultrasound data and visual contributions are accumulated along the ray. A transfer function assigns color and opacity to each data value. Amniotic fluid can be easily eliminated through the transfer function setup. Amniotic fluid has distinguishable low intensity values, and these are made transparent. Uninteresting outer structures cannot be separated from interesting structures through the transfer function as both have similar intensity values. Our visibility algorithm generates a clipping surface to automatically remove undesired structures and provide an unobstructed view of the desired structures (especially the face of the fetus).

- Direct volume rendering with advanced illumination model

The visual properties of the fetoscopic images and the requirements of the clinicians are analyzed in order to design and develop a method that can provide more realistic perceptual cues for the rendering and the interpretation of the images than current rendering methods. Direct volume rendering with an advanced illumination model which supports shadows, movable light sources, and realistic rendering of the human skin is applied to provide an impressive experience for physicians and parents alike.

State of the Art

2.1 Medical Ultrasound Imaging and History

The development of modern ultrasound machines was driven by various factors and it was progressively achieved by the intensive endeavors and research of many scientists coming from different fields over several decades. Efforts to develop a device that would allow to navigate a ship in the sea became stronger after the disaster of the Titanic. Although the ideas of using sound pulse-echo ranging for detection of icebergs was proposed already in April 1912, there was no technology available that could implement it. The first active sound detection apparatus was developed in secrecy during World War I with the goal to detect submarines which presented the major threat in the naval war. Dussik [36] for the first time employed ultrasound in medical diagnosis. His hyperphonogram displayed the ultrasound attenuation image of the brain. However this method for transcranial imaging was not adopted because of the attenuation artifacts in the skull. The active development of pulse-echo ranging and detection devices continued also during World War II. Sonar and radar were developed as a defense against submarine and aircraft attacks. After the war, medical practitioners continued to explore possibilities to use the acoustic technology for probing of the human body.

Ludwig and Struthers [93] in their report describe the successful application of ultrasound for the detection of gallstones. They used a pulse-reflection method with a modified device that was originally proposed for finding of defects and artifacts in metals. An oscilloscope visualized the amplitude of the received echo over time. This method became also known as the A-mode and it is used for measuring of distances of structures with ultrasound. The first two-dimensional cross-sectional images, which were called somagrams, were published by Howry [59]. They were used for the visualization of breast carcinoma and soft tissue structures. Somograms can be considered as the first B-mode images. The images in the B-mode were showing echo amplitude along each traced pulse-echo signal coming from the transducer. The pulses were transmitted with periodic timing and displayed on the screen that visualized time traces of echoes arranged vertically to indicate the depth. Brightness of the trace was proportional to the amplitude of the

echo. The scanning device required scanning in a water tank and the transducer moving on a rail.

Although the A-mode (1D) and B-mode (2D) ultrasound became established methods after only a few decades, it took much longer until the first 3D ultrasound system was developed. The first ultrasound systems capable of 3D visualization began to appear with the progress in computer technology and algorithms. Baba et al. [5] reported the first 3D visualization of the fetus. Their transducer was mounted on a position sensing arm and the image was reconstructed with a minicomputer. The brightness of each pixel of the gray-scale image was proportional to the distance between the transducer and the soft tissue of the fetus. In 1989 the first commercial 3D scanner appeared on the market. The Combison 330 was produced by the Austrian company Kretztechnik which in 2001 was acquired by GE Healthcare. The 4D technology 'LIVE 3D' was invented by Kretztechnik in 1998 and was incorporated in the Voluson 730 system.

Although the ultrasound devices from the pioneering times were not as advanced as modern machines, the generated images were able to demonstrate the correspondence with the scanned anatomy and showed the potential for further development. The history of ultrasound development is shaped by many pioneers and is characterized by many important landmarks. A detailed overview does not fit into the scope of this work. For a more detailed overview of the history of medical ultrasound we refer the reader to other existing works [143] [155].

2.2 Ultrasound Visualization Pipeline

Ultrasound data is nowadays acquired in medicine for many different purposes such as diagnosis or navigation during surgery. Ultrasound is widely used because of its high availability and non-invasive character. During recent years, it became a standard diagnostic procedure during pregnancy. The ultrasound scanning of the prenatal development has two important aspects, i.e., a diagnostic and an entertaining one. Clinically, it is used for the assessment of prenatal development during pregnancy.

Ultrasound scanning became also very popular among parents who are interested in the health condition of their fetus during the pregnancy and want to see their unborn baby. Prenatal imaging centers provide specialized services to their customers. Besides standard diagnosis, they sometimes also offer images and videos as a present for patients and their relatives [91] [115] [126].

From a technical perspective, algorithms that allow to visualize the acquired ultrasound data are organized in a visualization pipeline. This pipeline is similar to visualization pipelines of other medical imaging modalities like CT or MRI. A medical visualization pipeline usually contains not only automatic algorithms but for some algorithms it requires also interaction from the user. A thorough understanding of visualization algorithms and their parameters is therefore becoming an important part of the training of sonographers.

The pipeline starts with the data acquisition. Ultrasound scanning is based on the physical phenomena of sound propagation and echolocation. Data is typically acquired by scanning with the hand-held transducer. Data acquisition produces so called *raw* data that has to be further processed by other algorithms before the images are produced. In the US visualization pipeline, it means a signal reconstruction from the measured data and resampling. The next

step after data acquisition in the general visualization pipeline is usually denoted as filtering in a broad sense. In the context of ultrasound, the filtering typically means noise suppression. At this stage, 2D or 3D data is usually already represented by samples spatially organized in a regular grid. Individual elements of a 3D volumetric grid are called *voxels*. The next step of the pipeline is classification. Classification is usually assigning additional properties, such as labels or colors, to data samples. Classification typically corresponds to segmentation or transfer function specification. Segmentation usually assigns unique labels to individual voxels of the volume. If a transfer function is employed, it usually maps voxels to colors and opacities which can be used in the rendering step. Another type of classification can be achieved with clipping methods. Clipping methods apply geometric primitives that exclude part of data from the visualization. These techniques usually require interaction from the user. The most typical example of this techniques is the definition of a region of interest (ROI) box. After classification, images are rendered from the data. Images are usually generated from 3D ultrasound data by volume rendering algorithms.

The rendering step completes the visualization pipeline. The generated images are evaluated by clinicians who perform the examination. In case of modern ultrasound machines, the parameters of visualization algorithms can be interactively changed at any-time and therefore a real-time feedback is required. Data is constantly acquired by the ultrasound transducer during the examination. A large amount of data requires a special consideration regarding the data flow. The constant data flow requires a flexible pipeline which has a high throughput and a minimal memory footprint. This topic is covered in Chapter 3, where concepts of a data streaming pipeline are proposed. This thesis has a goal to provide new visualization algorithms in the context of the visualization pipeline that is based on the streaming concept. In the following, we describe individual steps of the state-of-the-art ultrasound pipeline in more detail and explain where we integrated our methods.

2.3 Basic Principles of Ultrasound Imaging

Medical ultrasound imaging uses an ultrasound machine to reconstruct images of human tissues. The most common ultrasound imaging mode is based on the sound reflection, also called pulse-echo mode (see Figure 2.1). An ultrasound pulse is generated by the transducer which is usually positioned directly on the surface of the human body. The frequencies of diagnostic ultrasound are typically between 2 and 18 Mhz.

The ultrasound acoustic signal is typically generated and received with a transducer which uses an array of piezoelectric elements. Piezoelectric materials, such as quartz crystals, are used to convert an electric signal into an acoustic signal. The array of piezoelectric elements is also called the imaging system's aperture. The acoustic signal is typically sent and received by the electronically controlled and synchronized array of elements. Their signal is appropriately delayed in order to enhance the summed echo. Therefore it is also called a phased array. The process of steering the phased array in order to focus and enhance the waves is called beam-forming. The ultrasound beam is generated in pulses by the vibrations of crystals. Vibrations are deliberately damped to stop after the acoustic signal is sent. The length of a pulse is limited by the damping material and it is typically around 2 or 3 wave cycles of the ultrasound

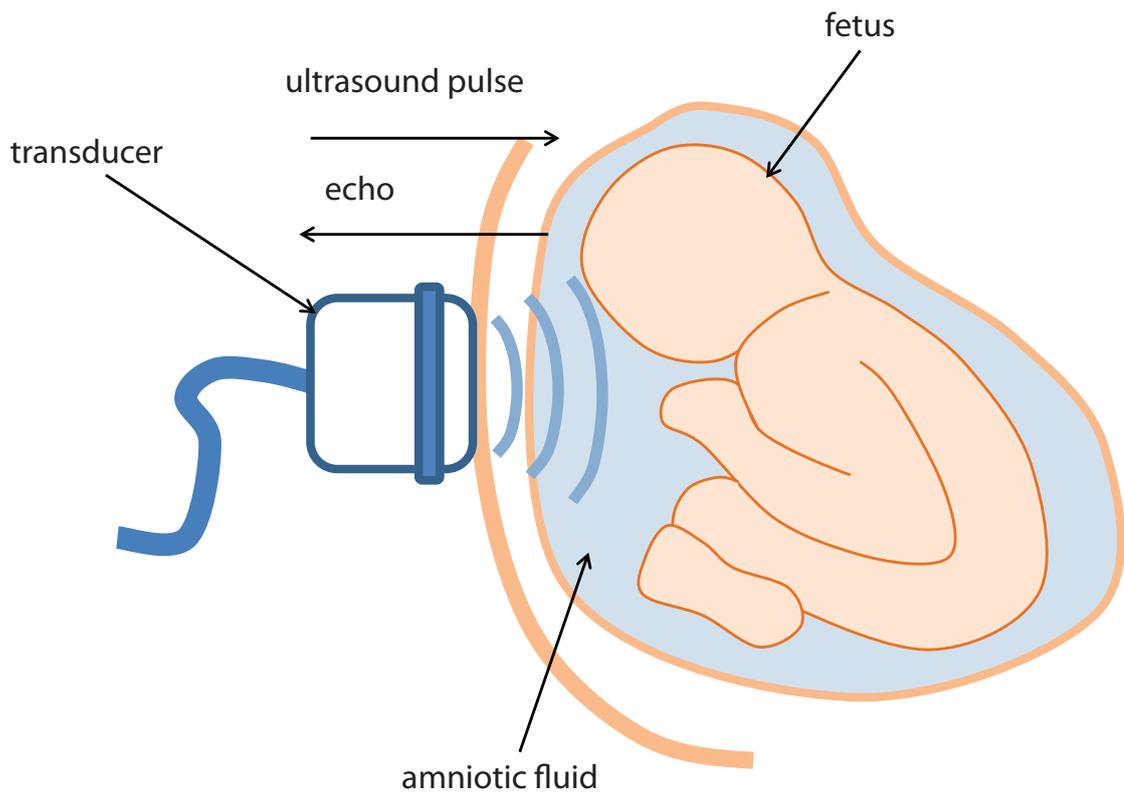


Figure 2.1: In pulse-echo mode of ultrasound, a transducer is positioned on the skin surface of the human body and it sends the ultrasound signal. Reflected echos from interfaces between tissues are received.

frequency. Waves of short pulses improve the resolution of the reconstructed signal. The pulse is formed with the rate that allows the ultrasound wave to travel to the scanned target and back. Pulse repetition frequency is 1-10 kHz for medical imaging.

The generated sound wave is assumed to travel in a straight line, also known as a scan line, at almost a constant speed with a mean value of 1540 m/s. The speed of ultrasound in tissues with a high content of water, such as human tissue, does not differ significantly from the speed of sound in water. Sound in water travels at about 1484 m/s. A sound wave which penetrates the human tissue has a compressional character and it propagates as a longitudinal wave with periodic compression and refraction. As the wave propagates through the body tissue, it is scattered, and its intensity is attenuated with distance. To compensate for the loss in the strength of the signal, a time-gain-compensation correction is applied when the signal is reconstructed. The character of the propagation of the ultrasound wave depends on the mechanical properties of the tissue, such as density and elasticity. When the wave passes from one tissue to another, a portion of the signal is reflected as an echo back to the transducer and another portion travels further. The ability of a tissue interface to reflect the signal is also called echogenicity and it depends on the acoustic

impedance difference between the tissues. From an ultrasound perspective, acoustic impedance can be understood as the resistance of a tissue to the passage of ultrasound. The higher the difference of impedance, the stronger the reflection of the sound. The air has an extremely low acoustic impedance in comparison to other body tissues. Therefore the reflection is strong at interfaces with tissues that are filled with air, such as lungs. A water-based gel is applied on the skin before scanning, in order to eliminate any air cavities between the transducer and the tissue and to improve the penetration of the ultrasound into the body. Bone has a relatively high acoustic impedance in comparison to other tissues and therefore the sound reflection is also very strong.

The reflected echos are acquired and converted into electrical signals by the transducer and sent to the ultrasound machine to reconstruct the signal along the scan line. The signal is reconstructed based on the known speed of sound in human tissue. Several steps are involved in the reconstruction of the signals received by the transducer. The latest scanners employ fast digital filters in the reconstruction and post-processing algorithms. A more detailed description can be found for example in the work of Szabo [143]. The signals of each transducer element are processed into a single scan line. The scan line corresponds to the A-mode (amplitude mode) of the ultrasound scanning and it is the simplest mode of ultrasound. The reconstructed scan line represents scattered echoes from the tissues and in this way it stores the information about the interfaces between the tissues. The amplitude of the signal along the scan line depends on the reflected echo and it can be used to differentiate between the different tissues, e.g., soft tissue, bone, water, air etc.

2.4 Ultrasound Data Acquisition and Reconstruction

A region of the human body can be examined with ultrasound in several modes. The reconstruction of ultrasound data from the scanned region depends on the ultrasound scan mode. In conventional B-mode (brightness mode), the data is reconstructed into a 2D cross-sectional slice which is afterwards displayed as a gray-scale image. The brightness of an image pixel corresponds to the magnitude of the echo reflection. In 3D/4D ultrasound, the information is reconstructed into a 3D volume which can be used for image synthesis by volume visualization algorithms. The volume is constructed from slices, slices are constructed from scan lines and scan lines are constructed from samples that are taken through sampling the signal. The volumetric data is spatially characterized by the axial, lateral and elevational resolution, i.e., the distance between the samples. Ultrasound is a real-time imaging modality and therefore it is also characterized by a temporal resolution.

Modern ultrasound devices are digital imaging systems which acquire information by discrete sampling and reconstruction of the signal. The acquisition of samples with the ultrasound system's aperture is done one by one with a delay between the samples that corresponds to the sampling frequency. The concepts associated to the Nyquist sampling theorem apply to the reconstruction of the ultrasound signal. This means that if the sampling frequency is sufficiently high, then the signal can be reconstructed without error. Interpolation is used for the computation of signal values which lie between the acquired samples. Linear interpolation is usually used for signal reconstruction.

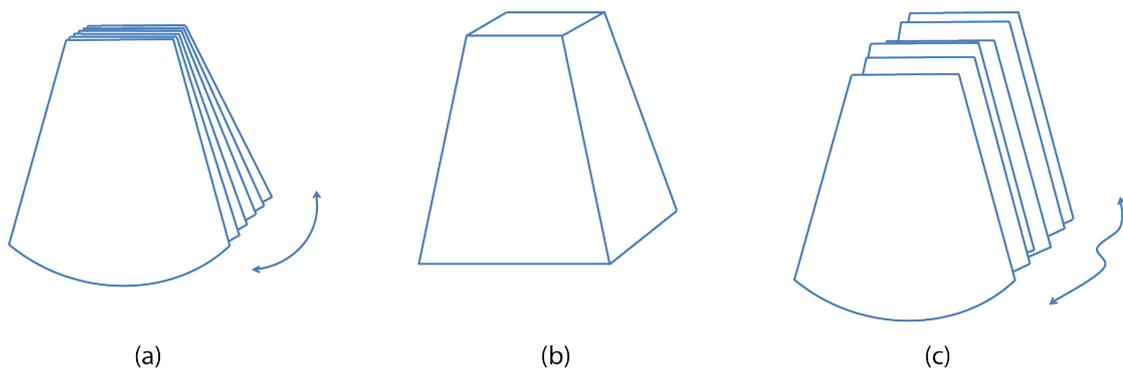


Figure 2.2: Different types of 3D ultrasound data acquisition. (a) mechanically swept transducer, (b) 2D transducer array, (c) freehand acquisition.

A standard 2D scan, also known as B-mode, is acquired with a 1D array of transducer elements. Typically at least 128 transducer elements are used in the configuration [113]. Acoustic signal samples are acquired by electronic focusing of the ultrasound beam using a simple scan line sweeping pattern. Data samples are taken along each scan line. Their axial resolution is limited by the signal pulse length. Shorter pulse lengths produce a better axial resolution. Scan lines are typically radially oriented with a certain angular distance. Therefore the lateral resolution of ultrasound data highly depends on the spacing between scan lines. The tissue that is located farther away from the transducer is scanned with a lower resolution. The axial resolution is usually better than the lateral resolution in ultrasound data. Contrast resolution of the ultrasound corresponds to the ability to distinguish between signal-amplitude sizes. Ultrasound imaging modality also has a temporal resolution. The temporal resolution allows to separate events in time and it is limited by the speed of ultrasound and the ability of the ultrasound device to reconstruct the signal. The ultrasound wave has to travel into the object and back along each scan line to generate one slice. The temporal resolution of the human eye is around 40 ms. This means that the real-time ultrasound imaging system has to generate images from the data at a rate of 25 frames per second (FPS) and higher.

The original sample positions of ultrasound data are typically defined by a curvilinear grid with polar coordinates. Raster displays usually require samples defined on a regular grid of voxels. A scan conversion algorithm transforms the original samples to a representation that is more suitable for display. Typically it resamples the original samples from the curvilinear grid into a regular grid with Cartesian coordinates [82].

3D volume data can be acquired with ultrasound using several methods [117], [48] [100] [161] [113]:

- Mechanically swept transducer

A mechanically swept transducer is the most common way of 3D data acquisition. A mechanism in the probe is moving the 1D transducer array in steps to produce volumetric data. The type of sweeping is predefined and it is accomplished by a stepper motor.

Co-planar scan lines of a slice are acquired as with the B-mode scan in each step of the motor. There are several ways of implementing the constrained sweeping which results into different geometries of transducers. If slices are acquired by wobbling of the array, the resulting data has a fan-like shape (see Figure 2.2(a)). Sliding of the transducer array results into the series of parallel slices. In addition, the array can also be rotated around an axis to produce conically shaped volume. The volume is produced in real-time, in a similar way as a B-mode image [24]. The difference is in the scan conversion algorithm, which has to be performed in 3D instead of 2D. The acquired data is usually resampled into a regular grid by a scan conversion algorithm before the image synthesis by a visualization algorithm is done. The ultrasound data used in this thesis were acquired by this type of transducers.

- 2D transducer array

Probes utilizing the 2D arrays of transducer elements can generate the 3D data without moving of the array (see Figure 2.2(b)). The acquisition of the volume is achieved by electronic focusing of ultrasound waves. The 2D array probes are relatively large because they require a large number of wires to be connected to the individual transducer elements. For example 128x128 elements would require 16384 connecting wires. Another major issue is related to the speed of the beam forming. A high number of 2D array elements would require a parallel beam former and parallel processing of the signal in order to maintain the real-time character of scanning. These technical limitations have an impact on the spatial resolution of the acquired volume and the final image. 2D arrays usually produce a pyramidal volume [87] [133] [159] [158] which is scan converted into a regular grid before volume visualization. Currently, 2D transducer arrays are mostly used for echocardiography because of their faster acquisition rate in comparison to the mechanically swept transducers [35]. The 2D array transducer can eventually replace the mechanically swept transducer if it improves in terms of performance and production costs [86].

- Freehand acquisition

Freehand acquisition systems use a standard transducer with a 1D transducer array which performs a standard B-mode scan and moves in an arbitrary way [48] (see Figure 2.2(c)). The position of the probe is usually tracked by a tracking device. This is sometimes also called tracked ultrasound. It can be also computed by an image-based algorithm, which is called sensor-less tracking. Sensor-less tracking methods are usually based on an automatic algorithm which uses the scanned image data for reconstruction of a 3D volume. In the past, systems based on ultrasound noise analysis by decorrelation [146] or linear regression [114] were developed. Systems with sensor-less tracking are worse in accuracy than tracked ultrasound systems using external sensors.

In systems based on tracking, an additional sensor is typically attached to the transducer. This is either a magnetic [99] or optical sensor [145] [22], but also mechanical and acoustic sensors can be used [134]. The position information of the sensor is used to calculate the 3D coordinates of each voxel. Optical trackers require a clear line of sight between the

sensor and the tracking device which can be a drawback for their application. They provide a more accurate calibration than magnetic-sensor based systems. The advantage of a magnetic sensor is that it does not require a clear line of sight. The presence of metallic material, which is very common in clinical environments, can have a negative influence on the scanning with magnetic sensor tracking. Before tracked free hand scanning is done, the system must be calibrated in order to obtain the transformation necessary for 3D position computation and volume reconstruction. Spatial calibration phantoms are often used for system calibration [100]. The data from free hand acquisition is usually resampled to a regular grid. Several possible freehand 3D reconstruction algorithms exist [134].

The main advantage of the freehand systems is that the 3D data can be acquired with a 2D ultrasound machine. Another advantage is that the precise tracking allows to compute positions in fixed external coordinates. This allows to register ultrasound data with data from other imaging modalities like CT and MRI [16] [4]. The registered information can be used for surgical planning or to improve stored medical records of patients in databases. Freehand acquisition ultrasound can also acquire 3D data of arbitrary dimensions. A minor disadvantage is that it requires external equipment for tracking which has an impact on the mobility of the system. The major disadvantage is that a 3D volume is usually not reconstructed in real-time, as with mechanically swept arrays or 2D arrays. A clinician has to move the transducer along a smooth trajectory and with a constant pressure in order to avoid distortions in the reconstruction. This limits the application of 3D data acquisition with freehand systems to static structures and also compromises the real-time character of ultrasound scanning.

2.5 Ultrasound Imaging Data Characteristics

The acquired and reconstructed medical ultrasound data is influenced by several factors which have a direct impact on the visual quality of the displayed images. This includes conventional B-mode images and also images synthesized by any volume visualization algorithms applied to 3D ultrasound data. It was already mentioned in the previous section that 3D volume data is usually acquired by mechanically swept transducer. This includes datasets which are considered in this thesis as well.

Ultrasound data acquisition depends on the physical properties of the transducer and the properties of pulse formation. Inherently, it is dependent also on the propagation and interaction of sound in tissues. Additionally, it is dependent on the ability of the transducer to detect the echo. Furthermore, it is dependent on the signal processing and data reconstruction algorithms. And finally, the displayed image is dependent on the properties of the visualization algorithms.

Ultrasound scanning and data reconstruction relies on a simplified model of sound propagation with several assumptions. These assumptions allow to compute a location and an intensity of each echo and to reconstruct the ultrasound data. The model assumes that the speed of the sound wave in the human tissue is constant and that sound waves travel along straight lines. Furthermore, attenuation in the human tissue is also assumed to be constant. The model assumes also that the detected echo traveled back only after a single reflection.

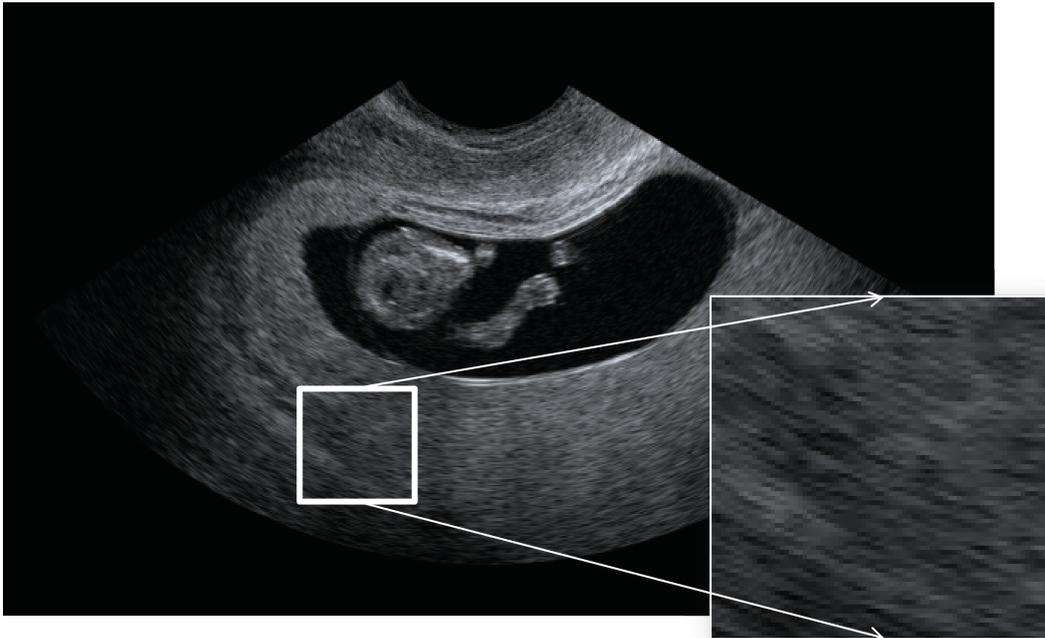


Figure 2.3: Illustration of speckle noise artifacts on a slice of ultrasound data. A speckle is a characteristic pattern that appears in ultrasound images.

In real ultrasound scans, these assumptions are not fully maintained. This gives rise to several types of artifacts in the ultrasound data and displayed images. In medical imaging, the term artifact is typically used to describe any feature of an image that does not represent the anatomical structures which are visualized. Artifacts decrease the ability of the human observer to see the desired anatomical structures. There are different types of artifacts in ultrasound data which are caused by the complex nature of sound propagation in real tissues. Ultrasound artifacts can be classified in the following way [106] [143] [43]:

- Speckle noise artifacts

Speckle noise is a characteristic pattern that appears in ultrasound data [1]. It is a random granular pattern which appears usually as a textural overlay in ultrasound images (see Figure 2.3). The texture does not correspond to underlying structures. It appears because of complex interference effects of the sound waves caused by diffuse Rayleigh scattering with sub-resolution structures [151]. The size of the structures is an order of magnitude lower than the ultrasound wavelength. The speckle noise has a negative impact on interpreting ultrasound images. It decreases the contrast of images and impacts the distinction of tissue boundaries.

- Attenuation artifacts

Attenuation artifacts belong to the most prominent artifacts of ultrasound imaging. The value of the data sample, and thus the brightness of the image, is dependent on the strength of the echo. The ultrasound wave attenuates in the tissue with depth, i.e., length of travel. Although a time-gain-compensation is applied for the reconstruction of the signal, the attenuation does not happen uniformly in all tissues. When the wave encounters tissue which attenuates it in a different way than water-based tissues, the attenuation artifacts appear in the reconstructed signal. Attenuation artifacts are also known as shadowing artifacts.

- Propagation artifacts

An ultrasound pulse does not travel along a single line, but has a complex 3D shape. Beam width propagation artifacts can be caused by echos coming usually from a strong reflector that are detected in the peripheral field of the focused beam by the transducer. Beam width artifacts appear because the image localization software can not distinguish between the non-overlapping objects and displays them as overlapping. These artifacts can be removed by adjusting the focal zone of the beam. Reverberation artifacts may appear when the primary ultrasound wave is repeatedly reflected back and forth before it is returned and detected by the transducer. Instead of one echo, multiple echoes are detected and displayed. This is caused by the refraction of the sound when it travels between tissues with different speeds of propagation. Displacement propagation artifacts can appear because of the non-uniform speed of sound in human tissues and refraction of the sound signal at the tissue interface.

The so far mentioned artifacts are characteristic for all types of ultrasound data. There are artifacts that appear in ultrasound images only in examinations with 3D ultrasound. Volume visualization algorithms are used for image synthesis from 3D ultrasound data. The presentation of images rendered from 3D data can become confusing and can give rise to additional artifacts. The artifacts that are unique for 3D ultrasound can be classified in the following way [106]:

- Acquisition artifacts

3D ultrasound data is typically acquired by sequential B-mode scanning of the tissue with a mechanical sweeping of the transducer. The acquisition rate of the data is limited by the speed of the moving transducer. If a motion occurs in the tissue, such as cardiac motion or respiration, it can give rise to motion artifacts. Motion artifacts are difficult to remove.

- Rendering artifacts

The advantage of 3D ultrasound is that it allows to display the complex shape of anatomical structures in one image and give a complete impression of the imaged anatomy. Volume visualization algorithms, which are used for the rendering of the images, require the specification of additional parameters. The quality of a rendered image depends on the choice of parameters, such as transfer function and lighting. An excessive choice of the parameter values, such as transfer function thresholds, can cause rendering artifacts. The anatomical structures studied in the rendered images can become difficult to interpret also

because of the complexity of ultrasound data. Structures rendered in the image from a certain perspective may appear as defects or additional features on otherwise simple surfaces. Adding visual cues that improve the visual presentation and depth perception can be useful for ultrasound volume visualization [105].

- Editing artifacts

With 3D visualization tools, it is possible to manipulate the data in order to produce good views of the studied anatomy. It is possible to manually define a region of interest (ROI) in the 3D data in order to limit the part of data that is rendered. Well defined ROIs can improve the view of the anatomical structures and improve the rendering performance, since only data in the ROI is rendered. But excessive ROI, which accidentally cuts through an important object and removes a part of it, leads to rendered images where important structures are missing and gives rise to editing artifacts. If the studied anatomical structure is obstructed by an occluder, and a simple ROI cannot be applied, it is also possible to manually remove the occluder with the use of an 'electronic scalpel' [102]. This improves the quality of the rendered images if it is used carefully. In some cases an inadequate use of this tool can remove too much of an important structure. Editing artifacts, which appear on the displayed images, are usually readily recognizable and can be interactively corrected. In some circumstances they can affect the examination and complicate the diagnosis with 3D ultrasound.

Despite many artifacts, which are detrimental to ultrasound imaging, several decades of intensive development succeeded to establish this modality in clinical practice. A lot of possibilities of ultrasound application are reported in the published literature [113]. Ultrasound imaging has improved in resolution, artifacts suppression techniques and visualization algorithms. It is also expected that this trend will continue in the future.

2.6 Noise Reduction Methods

Ultrasound data exhibits a low signal-to-noise ratio due to the properties of the acquisition. Noise has a negative effect on the final quality of the generated images and it decreases their discernability. Noise reduction methods are applied in order to minimize the negative effects. In ultrasound data, noise is present as high frequencies. Low pass filters are usually applied in order to improve the signal-to-noise ratio.

Filtering of the signal $f(x)$ corresponds to a convolution with a filter kernel $h_N(x)$:

$$c(x) = f(x) \otimes h_N(x) = \int_{-\infty}^{\infty} f(\tau)h_N(x - \tau)d\tau \quad (2.1)$$

In the discrete domain, the convolution is defined as:

$$c(x) = \sum_{k=-K}^K f(x_k)h_N(x - x_k), \quad (2.2)$$

and it corresponds to the *weighted average* of the neighboring samples. Where the term h_N corresponds to the normalized filter kernel. It defines the weights for the weighted average with coefficients of a filter kernel. The size K of the discrete filter kernel corresponds to the size of the neighborhood that is considered. Usually, only a local neighborhood around the sample is considered. Therefore this type of filtering is also called *local filtering*.

When the same filter kernel is applied to every voxel the filter is called *space invariant*. Many filter kernels have been proposed for noise reduction. The most simple one is called *average filtering*. The filtered sample is computed by simply averaging of neighboring voxels. Because of simplicity and effectiveness, this is the type of filter which we apply for filtering of ultrasound data in this work. Ultrasound data has usually highly anisotropic voxels. This means that the distance between slices of the volume is much larger than spacing within the slice. The voxels in neighboring slices should not be considered for filtering. Therefore, we apply the filter only to each slice and we do not consider neighbors from adjacent slices.

The Gaussian filter, also called binomial filter, is also a popular choice for ultrasound data filtering [125]. The filter kernel is constructed by discretization of a continuous Gaussian function:

$$g(x, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}}, \quad (2.3)$$

where σ corresponds to the standard deviation.

A lot of research for noise reduction in ultrasound data was developed with respect to speckle noise (see Section 2.5). Many speckle noise reduction methods have been proposed. The detailed overview of this methods does not fit into the scope of this work. There are methods based on adaptive filtering [9] [92] [37], methods based on anisotropic diffusion [2], and region growing methods [25].

Local filtering is often used for noise reduction. Local filtering can be applied also for other purposes than noise reduction. In this thesis, we will use local filtering for 2D surface reconstruction in Chapter 4 and light scattering approximation in Chapter 5.

2.7 Classification Methods

Different types of classification methods can be applied to ultrasound data. In general, all classification methods are based on a priori knowledge about the characteristics of the data or the imaged anatomy. In this work, we distinguish between three different classification methods, i.e., segmentation methods, transfer functions and clipping geometry.

Many sophisticated algorithms were developed in order to segment various anatomical tissues in ultrasound data. A very good survey of several ultrasound segmentation methods is given by Noble and Boukerroui [109]. In this work we do not focus on the segmentation of ultrasound data.

Using a transfer function is a data classification applied in direct volume rendering (see Section 2.8). Transfer functions assign optical properties, such as colors and opacities, to the data samples and they are applied directly during image synthesis. The most simple transfer function is a 1D function that is based on the intensity value of the data [85]. More complex, also called multi-dimensional transfer functions, include other properties derived from the data, for example

gradient magnitude or gradient divergence [68]. Some transfer functions are based on segmentation information which is extracted beforehand by a segmentation algorithm. Transfer functions are usually implemented as interactive widgets. The possibility to interactively control the transfer function in real-time, during the rendering of ultrasound data, is an important requirement. Visualization researchers developed a lot of specialized algorithms for ultrasound data visualization. In Chapter 5 we propose a simple extension of the 1D opacity transfer function with parameters that correspond to the optical properties of human skin.

Fattal and Lischinski [41] developed a variational classification method for 3D ultrasound data visualization. They propose an opacity classification method for smooth rendering of tissue surfaces based on the variational principle. Their method results in an efficient extraction of geometric surfaces from the data.

Hönigmann et al. [58] extend the basic 1D opacity transfer functions and propose the concept of adaptive opacity-based transfer functions. Their method works by analyzing a small neighborhood around a sample in the view direction. They apply a scale space filtering approach for the voxels in a local neighborhood to detect the surface of interesting tissue. In the next step they modify the opacity of voxel intensities prior to the surface. Their method was developed for rendering of tissue surfaces from ultrasound data.

Petersch and Hönigmann [110] develop a method for visualization of vascular structures in combination with silhouette rendering. Vascular structures are rendered in the context of surrounding tissues which are displayed only with silhouette mode. They propose the modification of silhouette opacities based on the gradient and the viewing direction.

Although many of the developed transfer function methods show promising results and improved visual quality of the renderings, sometimes they rely on pre-computations. In live scanning with ultrasound, where the image has to be rendered in real-time, this compromises their application and they can not be used.

Volume data can also be classified by a clipping geometry. The clipping geometry defines the part of the volume that is removed from the visualization. Only the volume, which is defined inside the borders of the clipping geometry, is further processed by other visualization algorithms. The tools for manipulation of the clipping geometry are usually interactive. They allow to specify the size and the shape of geometric primitives which define the borders of the rendered volume. A typical clipping shape, which is used in the ultrasound visualization, is a ROI box. The ROI box is implemented with a deformable clipping plane, to allow a manual adjustment for the irregular shape of features of interest. The ROI box is typically implemented as an interactive widget and provides the user with instant feedback by showing the resulting 3D rendering. However this introduces also a risk of editing artifacts which were mentioned in Section 2.5. Furthermore, it also increases the complexity of the investigation because the sonographer has to manually adjust the ROI box while holding the transducer. In this thesis (see Chapter 4) we propose an automatic method that can clip the volume in order to visualize the fetus.

2.8 Direct Volume Rendering

Direct volume rendering (DVR) is a method that is applied for rendering of ultrasound volume data. This method is in ultrasound terminology usually called *surface render mode*. DVR algorithms apply an optical model adapted from optics in which light is propagating through the media (volume data) along straight lines, called *rays*. It is interacting with the media according to the optical properties assigned by an optical model. Usually three types of interactions of light with media are considered in volume rendering: emission, absorption and scattering.

Basic Optical Model

A simple emission-absorption optical model assumes that the media consists of small particles which simultaneously emit and absorb the light [96] [104] [97]. The particles emit the light with intensity $L_E(s, \vec{\omega}_V)$ and they are the only light sources in the scene. Their luminance corresponds to the amount of light that is emitted at the position s in the direction $\vec{\omega}_V$. Since particles are considered to be opaque, they can also occlude and absorb the light traveling through the media. If the incoming light hits the particle, it is absorbed, and the outgoing light intensity is decreased. The attenuation of the light by the particles is expressed with an extinction coefficient $\tau(s)$. It corresponds to the attenuation of a fraction of light per unit length Δs . The extinction coefficient depends on the density of the particles and their size. In the emission-absorption model, the extinction coefficient $\tau(s)$ corresponds only to an absorption coefficient $\tau(s) = \sigma_A(s)$. It represents the probability that the light is absorbed at the position s . In more complex models, which include scattering of the light, the extinction coefficient includes also a scattering coefficient (see Chapter 5).

The emission-absorption optical model yields the differential equation for the transport of light [53] [104] [23]:

$$\vec{\nabla}_s L(s, \vec{\omega}_V) = Q(s, \vec{\omega}_V)\tau(s) - L(s, \vec{\omega}_V)\tau(s), \quad (2.4)$$

where $L(s, \vec{\omega}_V)$ is the light intensity, also called *luminance*, at the position $s = (s_x, s_y, s_z)$. The term $Q(s, \vec{\omega}_V)$, also called *source term*, represents in the basic optical model only the emitted light intensity $Q(s, \vec{\omega}_V) = L_E(s, \vec{\omega}_V)$. The light attenuation component is represented by the term $-L(s, \vec{\omega}_V)\tau(s)$. In the emission-absorption model it corresponds only to the light that is absorbed $-L(s, \vec{\omega}_V)\sigma_A(s)$ by the particles. In more complex models, which include scattering of the light, source term and light attenuation component consider also scattering of the light (see Chapter 5). The left hand side corresponds to the dot product between the light direction $\vec{\omega}_V$ and the gradient of the luminance. It is a directional derivative of the luminance that represents a rate of change of the luminance in the direction $\vec{\omega}_V$. The gradient is computed as a partial derivative $\vec{\nabla}_s = (\partial/\partial x, \partial/\partial y, \partial/\partial z)$, with respect to the position s .

The solution of the differential equation along the view direction $\vec{\omega}_V$, between the initial position and the eye position V , is the *volume rendering integral* [104] [127]:

$$L(V, \vec{\omega}_V) = L(0, \vec{\omega}_V)e^{-\int_0^V \tau(t)dt} + \int_0^V Q(s, \vec{\omega}_V)\tau(s)e^{-\int_s^V \tau(t)dt} ds, \quad (2.5)$$

where the first term corresponds to the light $L(0, \vec{\omega}_V)$ coming from the background in the direction $\vec{\omega}_V$, multiplied by the transparency of the medium between the initial point and the eye V . The second term is the integral of the contribution of the light intensity $Q(s, \vec{\omega}_V)$ at each position s in direction $\vec{\omega}_V$, multiplied by the transparency of the medium between a position given by s and the eye position V . The exponential term represents the optical depth, also called *transparency*, $T(s_a, s_b)$ of the interval [53]:

$$T(s_a, s_b) = e^{-\int_{s_a}^{s_b} \tau(t) dt}, \quad (2.6)$$

and it corresponds to the probability that the light ray travels a distance between positions s_b and s_a without being absorbed. After substitution with the transparency term, the equation can be written as:

$$L(V, \vec{\omega}_V) = L(0, \vec{\omega}_V)T(0, V) + \int_0^V Q(s, \vec{\omega}_V)\tau(s)T(s, V)ds, \quad (2.7)$$

If we consider light transport within the i th unit length interval $\Delta s = s_i - s_{i-1}$, we can write the equation:

$$L(s_i, \vec{\omega}_V) = L(s_{i-1}, \vec{\omega}_V)T(s_{i-1}, s_i) + \int_{s_{i-1}}^{s_i} Q(s, \vec{\omega}_V)\tau(s)T(s, s_i)ds. \quad (2.8)$$

In the discrete domain, the volume rendering integral is usually solved by a numerical approximation with a Riemann sum. The numerical solution of an integral of a function $f(x)$ can be expressed as a finite sum of pieces:

$$\int_{s_a}^{s_b} f(s)ds \approx \sum_{i=0}^{(s_b-s_a)/\Delta s} f(s_i)\Delta s, \quad (2.9)$$

where $(s_b - s_a)/\Delta s$ is the number of samples of the function used for integration such that $\Delta s = s_i - s_{i-1}$.

The volume rendering integral approximated with a Riemann sum can be expressed as [127]:

$$L(V, \vec{\omega}_V) \approx L(0, \vec{\omega}_V) \prod_{i=0}^{V/\Delta s} e^{-\tau(s_i)\Delta s} + \sum_{i=0}^{V/\Delta s} Q(s_i, \vec{\omega}_V)\tau(s_i)\Delta s \prod_{j=i+1}^{V/\Delta s} e^{-\tau(s_j)\Delta s}, \quad (2.10)$$

where Δs is the *sampling distance* between the corresponding samples. The exponential extinction term is usually approximated with the first two terms of the Taylor series as opacity [127]:

$$\alpha(s_i) \approx 1 - e^{-\tau(s_i)\Delta s} \approx 1 - (1 - \tau(s_i)\Delta s) = \tau(s_i)\Delta s. \quad (2.11)$$

We can also extend the computation of the volume rendering integral from an achromatic light $L(V, \vec{\omega}_V)$ to a chromatic light $L(V, \vec{\omega}_V, \lambda)$. After substitution of the source term Q with the

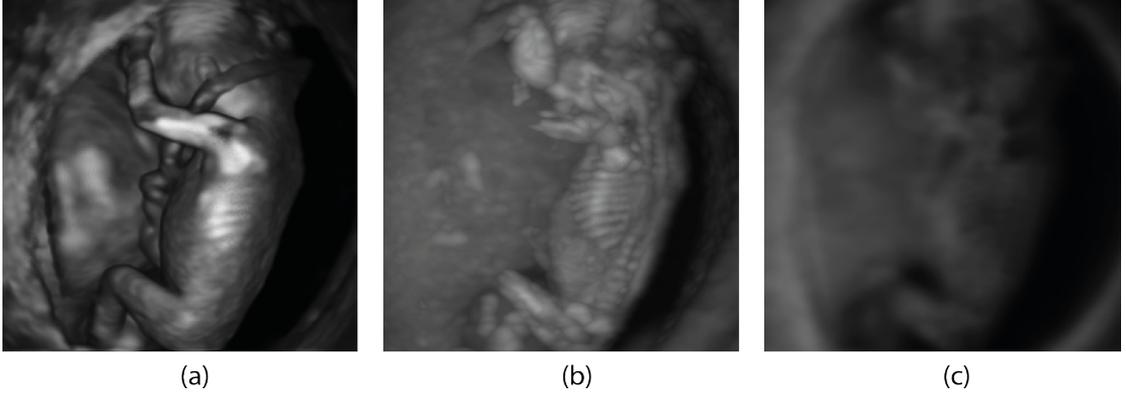


Figure 2.4: Direct volume rendering with different compositing operators. (a) front-to-back compositing, (b) maximum intensity projection (MIP) compositing, (c) average compositing.

sample color C and approximation of the exponential extinction term with the opacity term α we obtain [104] [127]:

$$L(V, \vec{\omega}_V, \lambda) \approx L(0, \vec{\omega}_V, \lambda) \prod_{i=0}^N (1 - \alpha(s_i)) + \sum_{i=0}^N C(s_i) \alpha(s_i) \prod_{j=i+1}^N (1 - \alpha(s_j)), \quad (2.12)$$

where $C(s_i)$ and $\alpha(s_i)$ correspond to colors and opacities assigned to discrete samples s_i by the transfer function. $\lambda = \{R, G, B\}$ corresponds to the evaluation of the final light contribution for a color image, i.e., R red, G green, and B blue channel. $N = V/\Delta s$ discrete samples are taken with equidistant intervals.

The discrete version of the volume rendering integral can be efficiently solved by an iterative algorithm using back-to-front and front-to-back *compositing* operators. Additionally, there are several variations of compositing operators that allow to render volume data.

- *Back-to-front* compositing

If the volume rendering integral is evaluated back-to-front, we can use the *over* operator:

$$C_i = C(s_i) \alpha(s_i) + (1 - \alpha(s_i)) C_{i-1}, \quad (2.13)$$

$$\alpha_i = \alpha(s_i) + (1 - \alpha(s_i)) \alpha_{i-1}, \quad (2.14)$$

where (C_i, α_i) are color and opacity of the newly composited values based on previous values (C_{i-1}, α_{i-1}) and values of the current sample $(C(s_i), \alpha(s_i))$ obtained by a transfer function.

- *Front-to-back* compositing

If the volume rendering integral is evaluated front-to-back, we can use the *under* operator (see Figure 2.4(a)):

$$C_i = (1 - \alpha_{i-1}) C(s_i) \alpha(s_i) + C_{i-1}, \quad (2.15)$$

$$\alpha_i = (1 - \alpha_{i-1})\alpha(s_i) + \alpha_{i-1}, \quad (2.16)$$

where (C_i, α_i) are color and opacity of the newly composited values based on previous values (C_{i-1}, α_{i-1}) and values of the current sample $(C(s_i), \alpha(s_i))$ obtained by a transfer function. In the case of front-to-back compositing, it is possible to use *early ray termination* [85]. Early ray termination interrupts the compositing once the accumulated value of transparency becomes zero, or small enough so that it has a negligible effect on the final result.

- *First hit compositing*

The first hit compositing operator allows to reconstruct the isosurface that corresponds to a certain intensity threshold. We interrupt the computation when the sample along the cast ray exceeds the threshold value. This compositing is similar to polygonal isosurface extraction with indirect volume rendering methods.

- *Average compositing*

Pseudo X-ray images can be rendered with the average compositing operator (see Figure 2.4(c)). The samples along each ray are averaged and the final images have a similar appearance as X-ray images.

- *Maximum intensity projection (MIP) compositing*

MIP is using the *maximum* compositing operator which searches for the maximum along the ray. In order to find the maximum along each ray, the whole volume must be traversed. This compositing operator reveals structures in the volume with the highest intensities (see Figure 2.4(b)). It is useful for rendering of bone structures (US, CT) and contrast-enhanced vascular structures (CT angiography). Rendering with MIP compositing does not allow to include local illumination. However, final image pixels can be colored according to the depth of the corresponding maximum. This approach is used in order to enhance depth perception.

- *Maximum intensity difference accumulation (MIDA) compositing*

MIDA is combining the *over* compositing operator with the MIP maximum compositing operator [18]. It modulates the accumulation of sample contributions along the ray with a term that corresponds to the change of the maximum value. Every such change along the ray is classified with the term $\delta(s_i)$:

$$\delta(s_i) = \begin{cases} f(s_i) - f(s_j), & \text{if } f(s_i) \geq f(s_j); i \geq j \\ 0, & \text{if otherwise,} \end{cases} \quad (2.17)$$

where $f(s_i)$ corresponds to the current value of the sample and $f(s_j)$ corresponds to the current maximum value along the ray. The MIDA accumulation modifies the front-to-back compositing operator with the $\delta(s_i)$ term:

$$C_i = (1 - \delta(s_i)\alpha_{i-1})C(s_i)\alpha(s_i) + \delta(s_i)C_{i-1}, \quad (2.18)$$

$$\alpha_i = (1 - \delta(s_i)\alpha_{i-1})\alpha(s_i) + \delta(s_i)\alpha_{i-1}, \quad (2.19)$$

The main advantage of MIDA is that it does not require the specification of a transfer function and it supports rendering with local illumination.

Basic Illumination Model

The emission-absorption model is a common model used in volume visualization. In the basic model, the light accumulated along the view direction is the consequence of the light emitting particles in the volume. It is possible to include external light sources into the model. The computation of the source term $Q(s, \vec{\omega}_V)$ in the volume rendering integral given in Equation (2.7) can be extended to [97]:

$$L(V, \vec{\omega}_V) = L(0, \vec{\omega}_V)T(0, V) + \int_0^V (L_E(s, \vec{\omega}_V) + L_L(s, X_L, \vec{\omega}_V, \vec{\omega}_L))\tau(s)T(0, s)ds, \quad (2.20)$$

where the light accumulation depends on the reflected intensity of the light $L_L(s, X_L, \vec{\omega}_V, \vec{\omega}_L)$ coming along the direction $\vec{\omega}_L$ from an external light source positioned at X_L . If the medium contains only particles that absorb and reflect the external light, we can omit the emission term $L_E(s, \vec{\omega}_V)$ of the particles.

In order to avoid costly computations of *global illumination*, the illumination comes from the external light source estimated with the *local illumination model*. The global illumination model computes shadows and advanced light scattering effects which are not produced by the local illumination model. The local illumination is computed based on the local neighborhood around a sample. Since these illumination models were originally designed for surface geometry, each sample in the local illumination models is associated with a surface. Gradient information has to be computed for the sample to approximate the surface normal. The *Blinn-Phong* model [15] is the most widely applied model for local illumination:

$$L_L(s, X_L, \vec{\omega}_V, \vec{\omega}_L) = \sigma_L(s, X_L)(k_A C(s) + k_D(\vec{\omega}_L \cdot \vec{N}(s))C(s) + k_S(\vec{H}(\vec{\omega}_V, \vec{\omega}_L) \cdot \vec{N}(s))^p C(s)), \quad (2.21)$$

where $C(s)$ corresponds to the optical property of the volume. It is the so called *reflective color* that it is assigned by a transfer function. The position of the external light source is defined by X_L and the attenuation of the light coming from the light source along $\vec{\omega}_L$ is defined by the attenuation term $\sigma_L(s, X_L)$. The attenuation term is defined as a function of the distance $d(s, X_L)$ between the position s and position of the light source x_L :

$$\sigma_L(s, X_L) = \frac{1}{k_0 + k_1 d(s, X_L) + k_2 d(s, X_L)^2}, \quad (2.22)$$

where k_0 is a constant attenuation, k_1 is a linear attenuation and k_2 is a quadratic attenuation. The ambient light contribution k_A is represented by a constant value. The term $k_D(\vec{\omega}_L \cdot \vec{N}(s))$

corresponds to the diffuse light component and it is defined by Lambert’s cosine law. It is the dot product between the light ray vector $\vec{\omega}_L$ and the surface normal $\vec{N}(s)$. The normal vector is approximated by the gradient $\vec{N}(s) = \vec{\nabla}f(s)$. The term $k_S(\vec{H}(\vec{\omega}_V, \vec{\omega}_L) \cdot \vec{N}(s))^p$ adds specular highlights based on the dot product between the surface normal $\vec{N}(s)$ and the half-angle vector $\vec{H}(\vec{\omega}_V, \vec{\omega}_L) = (\vec{\omega}_V + \vec{\omega}_L)/|\vec{\omega}_V + \vec{\omega}_L|$. The specular exponent p is a value that controls the size of specular highlights.

Direct Volume Rendering Algorithms

There are several approaches how to implement DVR. The algorithms for DVR can be divided into two classes, i.e. *image-based* methods and *object-based* methods. Many different methods were proposed in each class. Engel et al. [39] provide an excellent overview of many DVR algorithms which can not be presented in the scope of this thesis.

Ray casting is a widespread image-based method. Rays are cast for each pixel of the rendered image and samples are taken along each ray (see Figure 2.5(a)). Transfer functions assign optical properties, such as colors and opacities, to every sample. Their contributions are evaluated with compositing operators depending on the order of evaluation, i.e. back-to-front or front-to-back. After the traversal of all rays, the final image is displayed on the screen. Ray casting can be easily implemented on the programmable hardware of modern graphical processor units (GPU). A method for the visualization of the fetal face, introduced in Chapter 4 of this work, is based on a ray-profile analysis with ray casting.

Slice-based approaches, also called texture mapping approaches, belong to the category of object-based methods. In slice-based approaches, the volume is intersected with several slices defined by polygons. These polygons represent the so called *proxy geometry* of the volume. Depending on the orientation of the slice polygons with respect to the viewer, we distinguish between *axis-aligned slicing*, *view-aligned slicing* and *half-angle slicing*. Axis-aligned slicing was popular before 3D textures were introduced on GPUs. It can be implemented with 2D texturing hardware. The orientation in this setup is fixed to the major axis of the volume and it changes during rotation. View-aligned slicing (see Figure 2.5(b)) became prominent with the possibilities of 3D texturing hardware because of the better quality of the resulting images.

The orientation of slices in half-angle slicing depends on the light orientation of the external light source in the scene. Slices are orthogonal to the half-way vector between the light source vector and the view vector (see Figure 2.5(c),(d)). It was developed in order to integrate advanced illumination effects for volume illumination. We use this type of volume rendering approach as the basis for rendering with advanced illumination effects which is discussed in Chapter 5. In half-angle slicing, slices are textured with samples from the volumetric data. Each slice is projected onto the screen and composited with the appropriate operator. The volume rendering integral can be easily evaluated by compositing on the GPU. Alternatively, the integral can be also evaluated by programmable processors with a slice-based *sweeping* algorithm. The slice-based sweeping algorithm evaluates the integral in an iterative fashion. The advantage of this approach is that it allows to render the data which is provided by slices, one-by-one. Therefore it does not require a full in-memory representation of the volume. This type of rendering approach is also suitable for the pipeline that we describe in Chapter 3

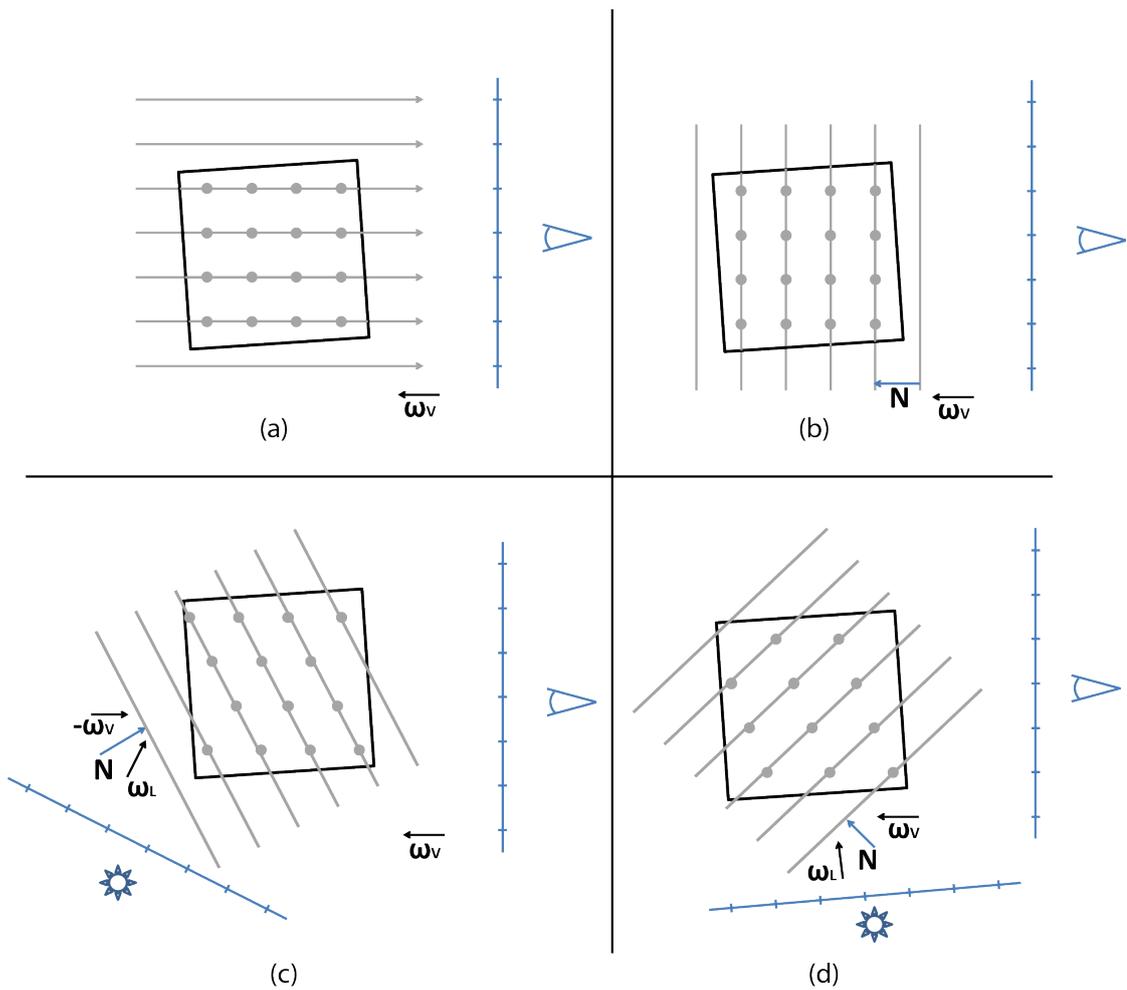


Figure 2.5: Image-based and object-based DVR algorithms. (a) Image-based ray-casting. (b) Object-based view-aligned slicing. (c)(d) Half-angle slicing - light in the back hemisphere, light in the front hemisphere.

After rendering with a DVR algorithm, images can be further post-processed for brightness and contrast adjustment. Finally, images are displayed and evaluated by the clinician on the screen of the ultrasound machine. In the following section we explain the importance of 3D ultrasound imaging within the context of displaying human prenatal anatomy.

2.9 3D Ultrasound Imaging and Human Prenatal Anatomy

The recent 3D ultrasound technology allows efficient examination of human prenatal anatomy [79]. In this work we focus on the visualization of fetal anatomy from ultrasound data. Our goals include the more realistic rendering of soft tissues and the development of an algorithm that can automatically detect and visualize the face of the fetus. For rendering of convincing images it is important to understand the imaged anatomy and to analyze the information that can be contained in current ultrasound data. The purpose of this section is to illustrate the current possibilities of ultrasound imaging in studies of the prenatal development. From a technical perspective, our main challenge is to achieve a realistic rendering of human skin and to find features that could help us to identify the face of the fetus in the ultrasound data. Therefore, in this context, we will also explain anatomical properties of the human skin that are responsible for its visual appearance.

Image data for diagnostic purposes is usually acquired in a clinical environment. The first trimester ultrasound examination (6-12 weeks of pregnancy) is usually performed to confirm the gestational age of prenatal development. The fetus is usually measured during the examination and diagnosed for abnormalities. A detailed fetal survey is an examination performed in a later age of gestation to evaluate the development of internal and external fetal structures. The fetal survey may also reveal the gender of the fetus. The image acquisition is usually done by sonographers who are trained to perform obstetrical examinations with ultrasound machines. Clinicians, who evaluate the images, also need a special training for a confident assessment of the ultrasound images. During education, they study atlases [77] with images acquired with fetoscopy alongside with state-of-the-art ultrasound images to have a clear understanding of fetal anatomy and development. Fetoscopy is an invasive procedure in obstetrics that is performed during pregnancy to visualize the fetus. A small camera is inserted through the abdominal wall and uterus into the amniotic cavity in order to directly screen the fetus and placenta. The fetoscopy can provide the doctor with a clear direct view on the developing fetus and placenta. The comparison between fetoscopic images and ultrasound images is an essential part of the training for all clinicians involved in fetal ultrasound. Performing ultrasound examinations where abnormalities are found requires extensive knowledge of the fetal anatomy and a lot of experience.

Kurjak and Azumendi [77] give an extensive overview of the current ultrasound imaging possibilities with the focus on human prenatal anatomy. Human prenatal development during pregnancy is divided into the pre-embryonal, embryonal and fetal period. All periods are characterized by many events with radical changes of the anatomy. Examinations with ultrasound are nowadays performed almost in every week of the pregnancy.

In the pre-embryonal period, 3D ultrasound can be used for the visualization of an ovary with a transvaginal ultrasound scan. A visible sign of the pregnancy can be observed during the 5th week of pregnancy, when the gestational yolk sac can be visualized and measured from 3D ultrasound scans [6]. Surface images seem to be beneficial in the evaluation of the yolk sac. In the 6th week the embryo appears as a C-shaped curve as the head grows. The amniotic membrane and umbilical cord can also be clearly seen. Between the 7th and 10th week the trunk and neck begin to straighten, the limbs start to develop and the facial features become more prominent on surface images. The fetus also develops a cartilaginous skeleton. During the 11th

and 12th week the arms and legs continue to develop, the neck is well defined, the face is broad and the sex can be distinguished. It is possible to analyze limbs, especially to count fingers and toes on rendered surface images.

In the fetal period the development of limbs, head and face can be efficiently visualized with ultrasound images. It is possible to visualize fetal hands, feet, fingers and toes. Bones of the fetus begin to gradually develop from the cartilage and they can be visualized starting from the 13th week. The fetus can be examined for skeletal malformations. The skeletal bones become harder during the pregnancy so they can carry the body weight. The fetal skull is the largest bony structure in the skeleton of the fetus. It protects the fetal brain, which is exposed to a significant pressure from maternal tissues during pregnancy and vaginal birth [30].

A detailed evaluation of the fetal face from 3D ultrasound can be done between the 19th and 24th week. The renderings of the face from 3D ultrasound are used to show relationships between facial structures, e.g., nose, mouth and eyes. The face is also assessed for malformations like cleft lip or other abnormalities. 3D ultrasound visualizations can also be used for a clear depiction of cranial structures and bone plates and offer an improved understanding of the cranial anatomy [33] [12]. In Chapter 4, we demonstrate how to use the strong echo response coming from bone structures in order to automatically visualize the face of the fetus from ultrasound data.

Another advantage of 3D/4D ultrasound is that it allows to study fetal behavior and facial expressions. These expressions are expected to have a diagnostic value [80] and they became an interesting subject of the current research. The surface render mode is also the preferred method for this type of examinations [83].

In general, a good visualization of fetal surfaces requires a sufficient amount of amniotic fluid in front of the target organ [148]. The ROI box has to be properly defined for a good view on the studied structure. 3D ultrasound imaging introduces a new set of parameters and controls which have to be learned [78]. The acquisition quality is also influenced by the expertise of the sonographer and therefore scanning with ultrasound has to be trained. Despite these challenges, 3D ultrasound imaging of prenatal development proved its value in clinical environments and became also a very popular facility among future parents. The main aspect for the parents is the appearance of the fetus on the image. However, current images provide a plastic-like unrealistic appearance of the skin surface. In order to improve the realism of prenatal human skin rendering, we need to understand the development and the optical properties of skin.

Human skin starts to develop from the early stages of the embryo. It arises by juxtaposition of the prospective epidermis and prospective mesoderm which are brought into contact [98]. Mesoderm is one of the primary cell layers in the very early embryo. The development of skin is driven by several signals as it grows additional layers of cells. At the end of the 11th week flat and polygonal cells can be observed on the surface. Touch pads on the hands go through their greatest development in the 15th week. Between the 12th and the 16th week the fetal skin is transparent and fine hair called lanugo appears. The inner surface of the epidermis becomes irregular as the hair follicle starts to develop. Skin begins to produce vernix in the twentieth week. Vernix is a white substance that covers the skin to protect it from the amniotic fluid. At the end of the 23th week the fetus is beginning to have the appearance of a new born infant. The

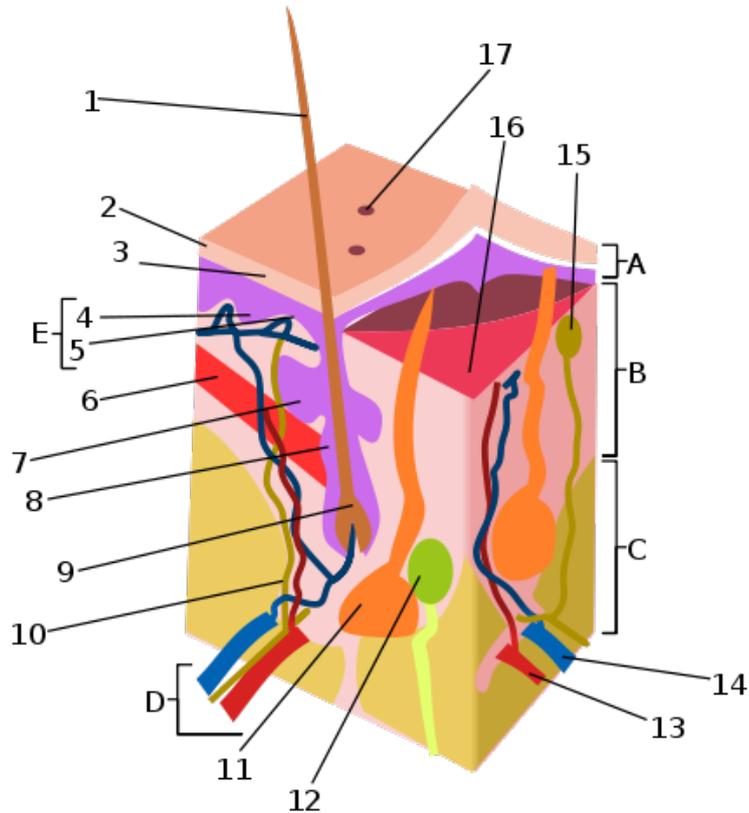


Figure 2.6: Anatomy of the human skin [154]. A. Epidermis, B. Dermis, C. Subcutis/Hypodermis, D. Blood and Lymph Vessels, E. Stratum Germinativum, 1. Hair Shaft, 2. Stratum Corneum, 3. Pigment Layer, 4. Stratum Spinosum, 5. Stratum Basale, 6. Arrector Pili Muscle, 7. Sebaceous Gland, 8. Hair Follicle, 9. Papilla of Hair, 10. Nerve Fiber, 11. Sweat Gland, 12. Pacinian Corpuscle, 13. Artery, 14. Vein, 15. Sensory Nerve ending (for touch), 16. Dermal Papillary, 17. Sweat Pore.

skin becomes less transparent while the fat layers begin to develop. Between the 33th and the 36th week the lanugo hair disappears and the skin is becoming less red and wrinkled.

From an optical perspective, human skin is a complex organ with specific inhomogeneous structural and biophysical characteristics. Several photobiological processes affect the appearance of the skin. The anatomy of the human skin can be described with a hierarchical representation that is based on the scale of the optical processes [10] [62]. There are three main layers of skin: the epidermis, the dermis and the hypodermis (see Figure 2.6). Each of these layers has distinct optical properties.

The epidermis is the outermost part of skin where the main substance regarding optical properties are the melanin pigments. The task of these pigments is to protect underlying cells from ultraviolet light. The absorption of the light increases towards the range of the blue light

in the wavelength spectrum, which has a shorter wavelength. The epidermis spectral absorption depends on the number of melanin pigments and it is different for every type of skin. The dermis, as the second layer of skin, is mostly responsible for the regulation of the temperature and the provision of nutrition to the epidermis. It forms about 90 percent of the total skin mass. The amounts of hemoglobin, carotene and bilirubin in the human blood are the most important factors that influence the optical properties of this layer. Particularly the skin color of Caucasians is influenced by these factors, since they have a smaller amount of melanin in the dermis. The dermis spectral absorption mostly depends on the amount of hemoglobin, with red light penetrating deeper than blue light. The function of the hypodermis is to allow the movement of skin over underlying tissues. Its contribution to the optics of the skin can be neglected.

Several of the aforementioned examples illustrate possible applications of 3D ultrasound imaging of prenatal macro-scale anatomical structures. However, ultrasound imaging possibilities are limited by the spatial resolution of the data [143]. Even though the resolution of the ultrasound data is sufficient for the rendering of the fetal skin surface, it is not sufficient for capturing of all underlying anatomical details that are responsible for its visual appearance. Optical effects in the skin happen on the cellular level in the scale of micrometers. The best lateral resolution of ultrasound data is 0.3-3 mm, depending on the ultrasound frequency. Therefore the optical effects of the skin illumination have to be approximated. In Chapter 5 we develop a suitable rendering method for ultrasound data that approximates the appearance of the prenatal human skin with advanced illumination effects.

Streaming of Ultrasound Volume Data

3.1 Introduction

Progress in ultrasound technology can not be viewed in isolation. Many technological improvements coming from other fields shaped the development the ultrasound technology, although they did not directly focus on it. For example, modern GPUs enable rendering of the acquired ultrasound data in real-time, at the frame rate of acquisition. Modern ultrasound machines use internally a lot of commercially available commodity hardware. The density of transistors of computing hardware and memory capacity has followed Moore's law, doubling the performance of computing hardware every 18 months. Although the main memory capacity of modern computers is constantly growing, the developers and users of data manipulation and visualization tools still struggle with the problem of its shortage. The reason for this is that not only memory chip technology has developed fast, but also the capabilities of scanning devices have increased as well which resulted in finer data resolutions. Additionally, the demands on the visualization quality, the level of detail and information content increased accordingly. This resulted in complex algorithms (see Section 2.2) requiring several copies of the data-of-interest in computer memory, eventually in different forms, too.

This problem was addressed by the developers in many cases usually starting from a thorough analysis of the specific task and often leading to its split into smaller ones. Depending on the data domain, spatial, temporal or spectral subdivision is used, which means that only a certain part of the data is readily available in the main memory and the rest is stored out-of-core [40]. Another feasible approach to the problem is data compression, e.g., two-level hierarchy [7], full hierarchy [45], run-length techniques [103] leading to compact data representations. Both approaches trade off memory consumption for time complexity. Usually, a less demanding storage representation leads to a higher computational burden and vice versa. Visualization data representations range from full memory representations at the one end to a storage of compressed data on an external medium at the opposite end. Full memory representations of the data require large memory but provide a minimal access overhead because of the direct access to each data

element. Compressed data representations require only minimal memory consumption but have a slow sequential access to data elements.

Volume visualization in a broader sense incorporates preprocessing operations [67] aiming at data restoration, e.g., noise reduction, interpolation, edge detection, feature extraction, segmentation, and others. These operations range from trivial, single-pass operations, such as average filtering and thresholding, to sophisticated ones requiring several sequential volume passes in different voxel orders, e.g., distance transforms.

In this chapter, we advocate slice-based streaming as a suitable way for working with grayscale, color, and, in general, multiband volumetric data sets. Numerous popular algorithms require access to only a small and localized subset at a time and can still be correctly performed. Our technique fits well with the well-known data flow visualization model, adopted by many general purpose visualization systems, e.g., VTK [129], ITK [60], AVS, OpenDX, SciRun, MeVisLab [14]. Its main feature is that each processing module stores only the algorithmically minimal amount of data required for performing the assigned task. In contrast to other approaches [81], we do not subdivide the data in arbitrary subvolumes, but rather set the data granularity to the level of slices. This means that for a volume of n_z slices of $n_x \times n_y$ voxels each, the minimal stored data unit is one slice of size $n_x \times n_y$. Some algorithms might require more than one slice, typically a few, to be stored in the main memory. Here we assume that the number of such slices represents data quantities small enough to fit into the main memory of the system. This is not a severe limitation, since we can easily store numerous slices today. Therefore, the main advantage of our approach is that the maximum size of data allowed for processing is not set any more by the main memory size but rather by the data source capacity.

Image processing operations are usually classified as point, local and global ones according to the area influencing the output voxel values. The streaming approach fits ideally to the first two categories. However, we observed that many global operations can also be implemented according to the scheme typical for local operations. This can eventually incur the need for two or multiple data passes, some of them in a reversed order. In this case we rely on out-of-core storage of intermediate results, which can be fortunately implemented in a transparent way. Thus, such modules can be included in the aforementioned streamed data-flow networks too, with a possible processing slowdown due to the buffering. Therefore, they are not suitable for real-time processing of the data.

Binding the algorithms to slices rather than to blocks brings certain simplifications. Most common data processing filters in this case require an easily predictable amount of main memory to execute, which is specified simply by the slice size or a known multiple of it. This knowledge can be used with advantage in building pipelines of modules, if the amount of available memory should be considered, in order to avoid performance deterioration. Further, for local operations, an additional layer of voxels of certain thickness, ranging from a single to numerous voxels, should be taken into account around each voxel. In the case of blocking this may introduce a significant increase of memory consumption, growth of algorithmic complexity and slowdown, since the layer should be added in all three dimensions, while in the slice approach it is added only in one dimension. Finally, numerous algorithms, such as region labeling, distance transforms, separable and finite impulse response (FIR) filtering, are inherently sequential, albeit

multiple passes may be required. Decomposing of the volume into blocks breaks this sequentiality and requires additional algorithmic complexity to overcome this obstacle and possibly leads to a slowdown.

3.2 Related Work

The concept of stream processing has existed since 1960 [139] and continues to be an active research topic today. Streaming algorithms can succeed only if the streams exhibit sufficient spatial or temporal coherence, i.e., the correlation between their basic entities and the proximity of their representations in the streams is strong enough. Basically, we encounter the notion of streaming in two different contexts. In multimedia processing, video and audio data streams are treated as endless ones, often starting in a specific capturing device and ending in a presentation device. In such streams various operations as compression, decompression, encoding, and scrambling are usually applied. This type of streaming is typical also for ultrasound scanning. Another context of streaming is related to out-of-core processing. This type of streaming is concerned with general visualization and data processing algorithms which does not aim for real-time performance. The requirement of real-time performance makes the design of algorithms for ultrasound challenging.

Streaming is applicable if the processing technique requires to see just a small localized data subset to perform. Thus, an arrangement is possible, where data flows through the processing unit, and at a certain moment a part of it is still waiting for processing outside of it, a part is loaded into the processing unit and is being processed, and the rest has already been processed or is being processed by another unit. This concept has been applied in different application domains, with largest impact in processing of, on the one side, unstructured triangular and tetrahedral meshes and, on the other side, volumes organized in structured grid. The proposed technique belongs to this group of algorithms.

Isenburg and Lindstrom [63] used streaming computation for Delaunay triangulations. They observed that large real-world data sets had inherent topological coherence, i.e., locality in vertex references, and spatial coherence, i.e., locality in vertex positions. Spatial finalization tags were inserted into the stream after three passes over the data to reflect the spatial coherence. These tags tell the program processing the stream that it may complete local computations, output partial results, and free some data structures. In their case, the Delaunay triangulation was computed, so an incremental triangulator read the finalized points from the stream and triangulated them.

Streaming of unstructured data was treated in work by Ahrens et al. [3]. Instead of splitting the data into blocks they were divided into pieces that represented a fraction of the total data set. Ghost cells were used when algorithms required neighborhood information. This architecture was implemented in the Visualization Toolkit (VTK) [129] together with streaming of structured data based on the amount of random-access memory. It included also support for message passing interfaces to allow for distributed parallel systems.

Ibaria and Lindstrom [61] presented a simple method for compressing very large and regularly sampled scalar fields. The method is based on the Lorenzo predictor, which estimates the values of the scalar field at each sample from the values at processed neighbors. The proposed algorithm is well suited for out-of-core streaming compression and decompression.

Previous work of Law et al. [81] addressed streaming of regularly sampled volumetric data sets. The data is split into blocks which are processed sequentially or in parallel on an architecture using a three-step pipeline update mechanism. The pipeline is demand driven and executes only when data is requested. In the first step, a request travels upstream and updates information that determines the data set's characteristics. The second step determines what input extent is required for the algorithm to generate its output. A negotiation process enables the configuration of the streaming based on a memory limit. In the final step, the pipeline begins to send the data for processing according to the requested data extent.

3.3 Stream 3D Operations

As already mentioned in the introduction, volume (image) processing operations can be classified into three categories:

- In *point operations* the resulting value of a voxel (pixel) depends solely on the input value at the same location as the location of the input voxel. Typically, these operations modify the voxel density, as for example in contrast enhancement, histogram equalization, thresholding and others.
- In *local operations* the resulting voxel value depends on the values in a neighborhood of the input voxel. A typical example is convolution, where the computations are performed within the extent of a convolution kernel. Such operations are used for various purposes such as noise removal, smoothing, sharpening or edge detection. They can be further classified as separable (e.g., Gaussian smoothing), where the required three-dimensional operation can be implemented by the successive application of 1D operators along the x, y and z directions, and nonseparable (e.g., Laplacian operator), where such a successive application is not possible. The former leads to $\Theta(k)$ run time complexity, while the latter leads to an order of $\Theta(k^3)$, where the filter kernel consists of $k \times k \times k$ voxels. Other local operations include specific techniques, such as local filtering (averaging, median, min, max filters), gradient magnitude computation, morphological operations (erosion, dilation etc.) and others.
- In *global operations* the resulting voxel value depends on the values of either all voxels in the volume-of-interest (e.g., Fourier transform), or at least the area-of-influence cannot be specified in advance (distance transforms, watershed transform). Thresholding and voxel density modifications are commonly classified as point operators. However, if the threshold value or the density transformation function is estimated by data analysis, the operations have to be viewed as global ones.

It is obvious that in order to implement a point or local operation, we do not have to store in memory the full data volume. It suffices to store just a part of it - in the case of a point operation, just one voxel - while for local operations values from a k^3 neighborhood. However, from an implementation point of view, it is more convenient to read and write the data in the basic units of 2D slices with one constant coordinate, usually denoted as z .

In the following, we introduce a finer-grained classification of operations based on the minimally required data volume which should be stored in the main memory of a computer without the threat of a significant performance degradation. While a small part of the data resides in the memory, the rest of the input volume waits to be processed on another unit and the newly calculated values are already sent out. This slice-based streaming approach is different from the traditional full in-memory representation, and its main advantage lies in its lower memory requirements.

Basic Streaming

All point and local operations can be computed on-the-fly by simple algorithms, such as thresholding or low-pass filtering, without any special considerations on data access. Point and local operations can be implemented with the slice-based streaming approach which requires only a single-pass of each algorithm through the data. More complex operations can be easily created by concatenation of simpler modules. *Data sources* are special cases of modules that correspond to algorithms which produce the volume data stream from other types of data representations, e.g., voxelization of geometric models [136]. A module that is responsible for the reconstruction of the acquired US acoustic signal into a volume is also considered as the data source module. *Data sinks* are modules that correspond to algorithms which produce other than volume data representations, e.g., a threshold value, histogram or a rendered image. Data sources and data sinks correspond to trivial cases when the volume data stream either starts or ends.

Buffered Streaming

In a global operation the output value for any voxel depends potentially on the values of all other voxels in the volume. In general, in order to implement such an operation we would need the direct access to all voxels, which can be efficiently implemented only if they are present in the main memory. However, we observe that many global operations can be implemented in two or more passes over the data, either by rereading the input data or by buffering intermediate results. In the second case it may be useful to read the buffered data in reverse order. We identify following buffered streaming operations:

- *Multiple Read Operations*

In these operations data analysis is performed first, followed by data processing on the basis of the obtained results. Typical examples are histogram based modifications such as histogram equalization, optimal classification and thresholding, density stretching and many others. Multiple read operations are dependent on a numeric quantity of interest derived in the first pass. They derive this quantity from the data and use it as a parameter in the second pass. Data streams are not rewindable. Therefore, in order to implement such an operation in the data flow framework, one has to store the intermediate data completely in the main memory. In the case when the intermediate data size exceeds the capacity of the main memory, one has to buffer the intermediate data in a temporary file. Then the second pass, and perhaps further ones, obtain the data from there instead of from the

original stream. Multiple read operations usually lead to out-of-core processing which is not suitable for real-time visualization.

- *Buffering of Intermediate Results*

It is often necessary to store intermediate results per voxel and to obtain the final value only in a second pass through the data. In such a case it is again necessary to either represent the intermediate data completely in the main memory or to buffer the intermediate data in a temporary file. This technique can be explained through the example of region labeling, which takes as input volumes with voxels classified into foreground and background ones. The goal of labeling is to assign unique labels to mutually separated foreground regions. In the first pass, based on a neighbor analysis, we store partially labeled foreground regions. Several distinct labels may be assigned to a single isolated region. Regions are uniquely labeled only in the second pass, using additional region correspondence information which was stored in the first pass along with the pre-labeled volume data. Buffering of intermediate results is usually not suitable for real-time visualization.

- *Stream Reversion*

In some operations, the desired effect can be similarly obtained in two phases, but with the second one applied on the result of the first one in reversed order. A typical example of this approach is given by the various types of distance transforms [65]. Here, instead of finding the nearest surface point for all voxels, distances are locally propagated over the volume in several runs in different directions. Another example is given by the FIR filtering implementation of various filters (e.g., Gaussian filter), which requires forward and backward pass along directions parallel to the coordinate axes. Stream reversion usually leads to offline visualization algorithms not capable of real-time performance.

Parallel Streams

In all operations introduced above we assumed that the processed data is scalar. However, processing of color and multiband data is a common task, too. In order to keep the streaming of data simple, instead of introducing general multiband operations, we introduce the operations of stream splitting and merging. Stream splitting is an operation that corresponds to the output of the algorithm which generates more than one value per voxel. Stream merging is an operation that takes voxel values from several streams and computes one voxel out of them. For example, computation of the gradient can lead to stream splitting and the computation of the gradient magnitude can lead to stream merging. If modules of the streaming pipeline work in parallel, implementation of stream splitting and stream merging requires synchronization mechanisms for communication between modules.

3.4 Streaming Architecture of the Ultrasound Visualization Pipeline

Recent development of ultrasound scanning devices allows to capture the high-resolution data of the moving fetus in real-time (4D ultrasound). A 4D acoustic probe called a transducer, is used to constantly acquire the volume data. After the volume is acquired by the transducer, it is provided to the visualization pipeline of the ultrasound machine. Every stage of the pipeline modifies the data. The pipeline requires several different instances of the data for each stage. Hardware limits render this type of processing impossible for real-time processing. Such a large amount of data requires special strategies for handling of the data flow. The streaming concept, introduced in the previous section of this chapter, provides a solution for efficient processing of ultrasound data. In the following, we present considerations made for the architecture of our visualization pipeline with regard to the streaming concept.

Our architecture can be subdivided into four main *modules* which correspond to different stages of our visualization pipeline as illustrated in Figure 3.1. The volumetric ultrasound data is constantly acquired by the 4D ultrasound transducer. Afterwards, it is sent to the ultrasound machine for imaging. The amount of data that is generated by the transducer requires adequate computational power for real-time imaging with DVR. Therefore, GPU utilization was proposed for real-time interactive visualization of ultrasound data [76]. Modern GPUs are processors with dedicated memory and are capable of highly parallel computations. However, their memory is also limited and does not allow full in-memory representation of the data for each stage of the pipeline. To maximize the visualization performance and to take advantage of the parallelism of modern GPUs, we process the data in our pipeline with the slice-based streaming. The acquired data is processed in *slabs*. Every slab consists of a number of slices that can fit into the memory of the GPU. When a slab has been processed by one module, it moves further down the pipeline and is processed by another module. This approach allows to minimize the memory footprint of the pipeline and to optimize the utilization of hardware resources. However, this approach does not allow a global access to the whole volume. This constraint has to be considered by all algorithms applied in later stages of the pipeline.

Scan Conversion Module

There are various types of transducers that are applied for the examination of different parts of the body, e.g., abdominal transducers, endocavity transducers. These transducers differ not only in physical shape but also in the data acquisition geometry. Raw acquisition data is represented as volumes in curvilinear coordinate systems (see Section 2.4). Modern 4D ultrasound transducers can produce sequences of volumes with high frequency. The data is sent to the ultrasound machine for further processing and visualization. Many volumetric visualization methods are designed to work with data samples in Cartesian coordinates. To allow unified access to the ultrasound data for all visualization algorithms in the pipeline, we resample the data to a Cartesian grid. The resampling from the curvilinear representation to the Cartesian representation is called scan conversion (see Section 2.4).

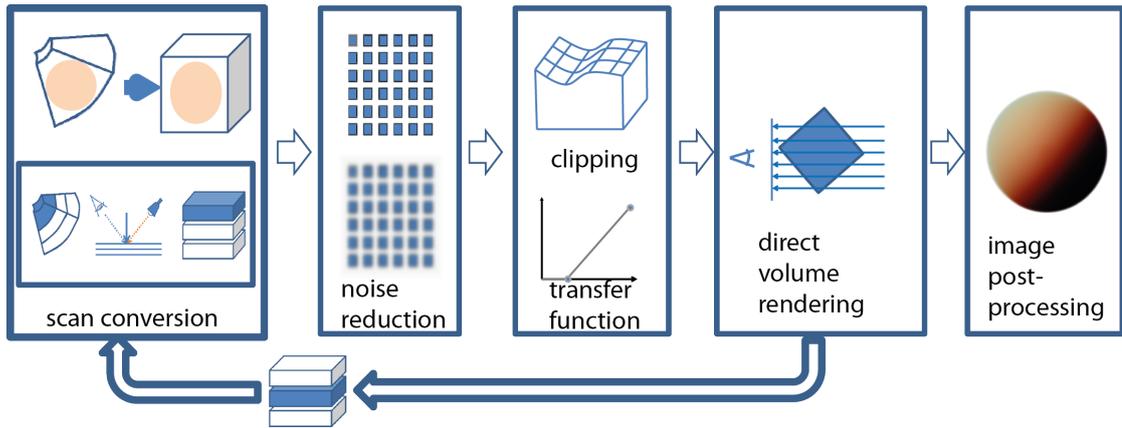


Figure 3.1: Process diagram of the visualization pipeline with streaming of slices organized in slabs. The original ultrasound data is reconstructed with the scan conversion module, filtered with the noise reduction module, classified in the module for clipping surface computation, rendered with the DVR module. The final image is post-processed before the presentation on the ultrasound monitor.

Our approach is similar to the approach proposed by Kuo et al. [76] In their pipeline, they compute Cartesian coordinates from the spherical coordinates of the original curvilinear grid directly on the GPU. They generate a proxy geometry of view-aligned slices that intersect the pyramidal scan frustum. Slices are textured with samples interpolated from the original data, and stored in a 3D texture, by simple coordinate transformation from polar to Cartesian coordinates. In the next step they perform slice-based volume rendering (see Section 2.8).

In our pipeline, we compute the coordinate transformation for scan conversion based on the proxy geometry of half-angle slicing (see Section 2.8). We convert original US data in slabs and store half-angle slices in a texture memory of GPU for further processing in the following modules of the pipeline. This means that our half-angle slices intersect the pyramidal scan frustum perpendicular to the vector that is half-way between the view direction vector and the light direction vector.

In streaming terminology, this module is the start of the data stream which produces a volume by resampling. The resampling algorithm requires, for each interpolated sample, surrounding samples from a local neighborhood. It can be implemented with basic streaming. A slab with a number of slices is scan converted and provided to the next stage of the pipeline. The scan conversion is a computationally demanding step but it can be efficiently implemented on the GPU.

Noise Reduction Module

In this stage, we filter the data in the slab to improve the signal-to-noise ratio and to reduce artifacts before the rendering of the data. We apply a simple averaging filter at each slice of the

volume (see Section 2.6). As discussed in this section already, this type of operation is a basic streaming algorithm and it can be processed in one pass through the slices of the volume. The filtered version of the data is used for gradient estimation. This improves the visual appearance of illumination effects when a simple illumination model is used. However, the application of a noise suppression filter leads to a loss of information which can be important for the clinical evaluation. As there is a trade-off between noise suppression and information loss, we render the original and the filtered data in the next stage of the pipeline simultaneously. The clinician can switch between the filtered and non-filtered image in the post-processing step or combine them with blending.

Classification Module

In this section the algorithm for the automatic computation of the clipping surface is executed. In streaming terminology, it is a data sink module that can be implemented as a basic streaming algorithm. One volume data stream ends because the algorithm produces a non-volume result. The algorithm has to pass through the whole volume with DVR. Details of the algorithm are discussed in Chapter 4. In the current implementation of the pipeline, this would mean that a full scan-conversion of the data would need to be done twice. The first time, for the estimation of the clipping surface and the second time for the actual rendering. The scan conversion algorithm is computationally very expensive. Therefore we decided to use a volume with lower resolution for classification. This significantly reduces the computational load. After the clipping surface is computed, it is provided to the rendering algorithm. In our system, we use a simple opacity-based transfer function for classification of samples that does not require any special algorithmic considerations or pre-processing.

Rendering Module

A direct volume rendering (DVR) algorithm visualizes the scan-converted data from the sequence of slabs that are provided to it one-by-one. Slabs are discarded from the memory to free space for the new data. It is not possible for the rendering algorithm to access the data from the previous slabs. Slice-based volume rendering algorithms are ideal candidates for our streaming pipeline. From the streaming perspective, the slice-based DVR module belongs to the category data sink and it can be implemented as a basic streaming algorithm. Details of the implemented algorithm for our system are discussed in Chapter 5.

The rendering stage has to be able to simultaneously process slabs of the original and of the filtered data. After processing of the last slab of the volume, the output images enter the final, post-processing stage of our pipeline. In the post-processing stage, final corrections are applied to the output image that appears on the screen of the ultrasound machine. The clinician can view images rendered from the original and the filtered version of the data. According to the level of noise that is present in the data, he/she can choose between the filtered and the original version. Furthermore, the clinician can blend between the images rendered with different algorithms which can be desirable in some studies.

3.5 Conclusion

In this chapter we provided a formal description of volume processing operations. We proposed an approach based on slice-based streaming which provides a solution for the memory shortage problem in the case of processing and analysis of volumetric data. In our version of streaming, data flows through independent processing modules which store a minimal fraction, i.e. a slab, of the whole data set, with a slice as a basic data entity. Such an approach allows to address the problem of high throughput of high resolution data if it is implemented on the GPU. We use the basic streaming concept for the design of the visualization pipeline in our system. We propose a novel pipeline that is based on the main goals of this thesis. Special considerations are made in the design regarding our rendering and classification algorithms. In the following chapters, we discuss how we integrate these algorithms into our pipeline.

Smart Visibility for Prenatal Ultrasound

4.1 Introduction

Prenatal ultrasound of the human fetus is a standard diagnostic procedure in obstetric medicine. Current 3D/4D ultrasound scanners can produce volumetric data of the moving fetus in real-time. With 3D/4D ultrasound it is possible to get 3D renderings that can show relational anatomy in one image and increase the confidence in the diagnosis. The current procedure of scanning with US requires a lot of training for medical personnel that has to take care of patients and also control the machine. Usually it is difficult to locate the fetus on a 2D reformation of the streamed data because the fetus is always occluded by surrounding tissue and is also changing its position during the scanning time. Fetus and surrounding tissues have the same intensity values of the ultrasound signal. The fetus is typically embedded in amniotic fluid. Furthermore the fetus and amniotic fluid are embedded in surrounding occluding structures (womb, placenta,...).

The operator captures the volume by moving the transducer. The fetus has to be located inside of the scanned region in order to render expressive images. In 3D rendered images, surrounding structures generate undesired occlusions. One aspect, which is very important for parents, is that they are interested in the face of the fetus. Because of occluding tissues, currently it is necessary to define a region of interest (ROI) that allows to visualize the fetus in a 3D rendering without any occluders. In practice it means to interactively define the cubical ROI with an adequate size and position. A typical region of interest is shown in Figure 4.1(a)(b). The ROI box is implemented as an interactive widget with parameters mapped to the interface of the ultrasound machine. Additionally, because of the irregular shapes of many occluders, the front face of the ROI box can be deformed as well. The face of the ROI box can be deformed by moving of a control point in the interactive widget. The surrounding tissue in front of the fetus is obstructing the view and therefore it is not visible. The fetus becomes visible, when the face of the ROI box is deformed, as shown in Figure 4.1(c)(d).

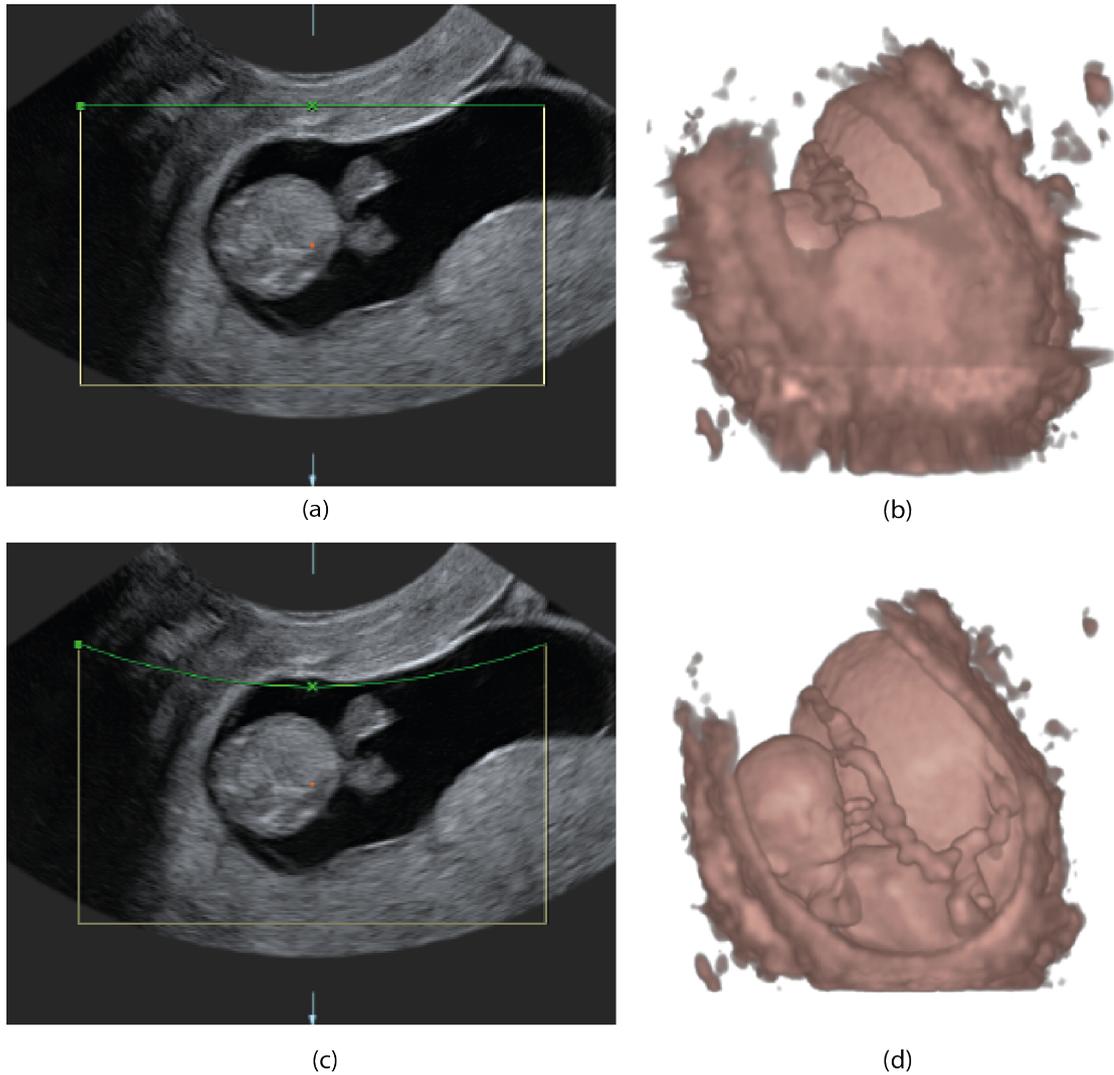


Figure 4.1: Manual ROI box definition. (a) a scalable ROI box defined around the fetus by the operator with (b) 3D rendered image, (c) front face of the ROI box deformed by moving of the control point to remove the occluder in front of the fetus, (d) corresponding 3D rendering with the deformed ROI box face.

There are also more advanced clipping techniques than ROI box, such as the 'electric scalpel', that can be applied in order to remove parts of the volume in difficult scenarios [102]. The user can define the geometric shape of clipping primitives by drawing their contours on the slice. The shape of the clipping object can be extruded in to order create a 3D object from the basic 2D shape. This object is used afterwards to remove part of the volume. 'Electric scalpel' usually requires even more interaction from the user and it is not suitable for real-time interaction.

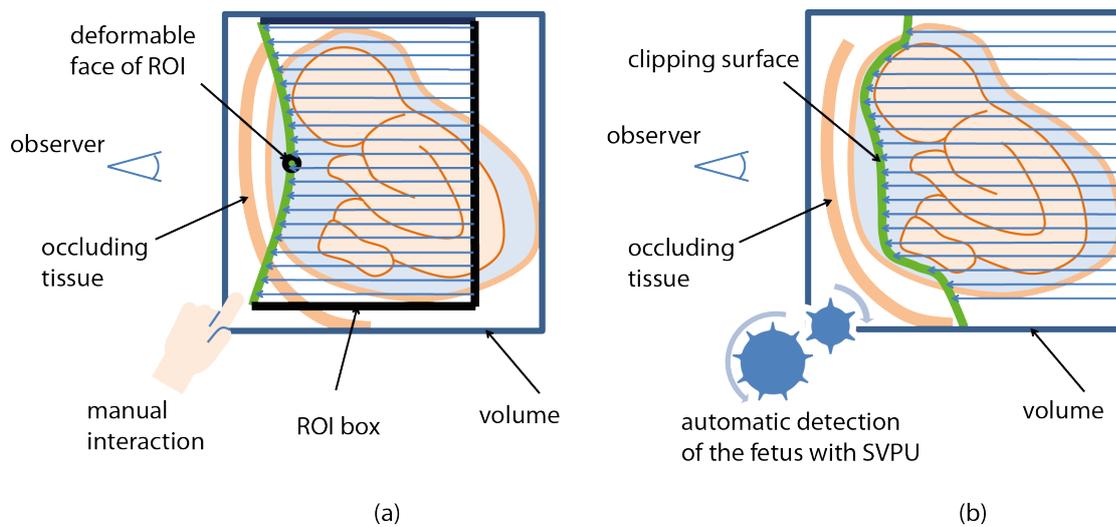


Figure 4.2: Comparison between the standard method for classification of volume data with the ROI box and the novel smart visibility method for prenatal ultrasound (SVPU). (a) a scalable ROI box is manually defined around the fetus by the operator in order to remove the occluder (b) an automatic algorithm of SVPU computes the clipping surface in order to remove the occluder in front of the fetus.

We have developed a new method called smart visibility for prenatal ultrasound (SVPU). Our method can produce a 3D rendering of the human fetus from 3D US data. It automatically gives an unobstructed view on the fetus and eliminates amniotic fluid and surrounding tissue. SVPU generates a clipping surface to automatically remove undesired structures and provide an unobstructed view of the desired structures (especially the face of the fetus). Figure 4.2 illustrates the basic principle of the SVPU in comparison with the manually defined ROI box. Several requirements for the method had to be met during the development:

- *Face detection* - the method should be able to display the face of the fetus, if the fetus is present in the captured volume.
- *Real-time response* - SVPU has to work interactively because of the character of the data that is produced on-the-fly.
- *Automatic performance* - the algorithm should work automatically without additional tweaking of parameters in order to minimize the interaction for the operator.
- *Robust to editing artifacts* - editing artifacts (see Section 2.5) appear because of excessive cutting of important features with visualization tools. If they appear, the method should provide an indication of the missing features.

The SVPU method was tested with several static volumes and also sequences with various data characteristics. The results show that it reliably detects faces on all datasets if the fetus is present

inside the scanned volume. In the following chapter we present related work and describe the method in more detail. Afterwards results that were achieved with the method are presented and compared to the currently used clinical solution. The final part of the chapter contains the conclusion and a short discussion of future work.

4.2 Related Work

Visibility techniques: In 3D visualizations, objects of focus may be obstructed by some other objects of lower importance. Exploded views are sometimes applied in visualization to tackle the problem of occlusion. These techniques and their applications to DVR were discussed in the works of Correa et al. [29] and Bruckner and Gröller [17].

A usual way to eliminate the objects that are occluding important objects is the application of cutaway techniques. Ghosting techniques are usually applied together with cutaways in order to provide a context for the rendering. They provide an impression of the structure while the object of interest is still visible. If the object of lower importance is cut away, the ghost lines generated by ghosting techniques still show the structure of it. These methods are commonly applied by illustrators for visualizations in many areas.

Feiner and Seligmann [42] develop a set of techniques for supporting dynamic illustrations that maintain a set of visibility constraints by cutaways and ghosting techniques. Diepstraten et al. [32] discuss different approaches to generate cutaway illustrations. In previous work of Burns et al. [19], an adaptive cutaway method for comprehensibly rendering of polygonal scenes was proposed. The application of advanced direct volume rendering to live ultrasound data was presented by Burns et al. [20]. They propose an importance-driven approach that combines ultrasound data with 3D Computed Tomography scans to maintain the visibility of important features.

A lot of work done by visualization researchers was focused on smart visibility methods. An overview of various methods, that are often inspired by traditional illustrations, can be found in the work of Viola and Gröller [149]. Cutaway techniques were applied for the visualization of vessels in the work of Straka et al. [141]. Krüger et al. [73] developed combined visualization techniques for the surgical planning of lymph node removal from the neck. Several importance-driven visualization techniques were proposed by Viola et al. [150]. They require an explicit segmentation of objects in the volumetric data and automatically generate cutaway visualizations which can be combined with ghosting techniques. Clearview was developed by Krüger et al. [74] and it applies similar focus+context visualizations with cutaways and ghosting techniques for DVR.

Rezk-Salama and Kolb [120] address the problem of occlusion in direct volume renderings with opacity peeling techniques. A similar approach for peeling of features blocking the visibility of more important objects was applied by Malik et al. [95].

Wu and Qu [156] propose a framework for interactive editing of volumetric renderings to achieve comprehensive focus+context visualizations. They allow to delete features from the rendering by an interactive transfer function design. Mak et al. [94] propose a semi-automatic framework for visibility-aware DVR. In the work of Correa and Ma [26], the occlusion spectrums of datasets are analyzed. They demonstrate that 2D transfer functions can be designed in

order to visualize objects in DVRs depending on their relationship to other objects based on their mutual occlusion. Visibility histograms and visibility transfer functions are developed by Correa and Ma [27] [28]. With these functions, users can improve the visibility in the complex scenarios of volumetric datasets. Tikhonova et al. [144] develop exploratory techniques for time-varying data based on ray-attenuation functions. Kubish et al. [75] develop smart visibility techniques for tumor surgery planning.

Surface reconstruction: In this chapter we describe a method that is using an automatic cutting surface constructed from scattered data. The problem of surface reconstruction from scattered data is also called scattered data approximation problem. These problems are studied in the signal processing theory of nonuniform sampling and reconstruction. The theory of uniform sampling and reconstruction is far better studied by means of the Fourier analysis and provides tools for efficient signal processing. The theory of nonuniform sampling and reconstruction is not so mature. However, many methods have already been developed in this area also for computer graphics applications.

Seminal work was done in this field by Shepard [131]. Shepard defined an interpolation function as the weighted average of the data. Lee et al. [84] propose a method for scattered data interpolation with B-Splines. Pottmann and Wallner [111] introduce approximation algorithms for developable surfaces. In the work of Haber et al. [50] a method for the smooth approximation and rendering of large scattered data sets is proposed. Bertram et al. [13] present an adaptive smooth scattered data approximation method for large scale terrain visualization. Nielson [107] proposes a scattered data reconstruction for noisy data based on eigenvectors.

A large number of other methods has been developed in the field of scattered data reconstruction and can be found in the literature. Scattered data interpolation can be classified into several categories. A lot of excellent surveys exist that are related to the scattered data interpolation and reconstruction topic. We refer the reader to work of Powell [112] for methods based on radial basis function methods. A survey of Dyn [38] presents methods based on subdivision schemes. A survey by Bajaj [8] for approaches based on algebraic solutions. Techniques based on piecewise polynomial solutions are discussed by Schumaker [130]. A survey that attempts to bring all these various surface reconstruction methods together was done by Lodha and Franke [44], [90]. Glassner [49] provides also a good overview of methods developed for computer graphics applications.

Direct volume rendering: A lot of work was done in the visualization community related to DVR. A more detailed overview of basic DVR concepts is given in Section 2.8 and in Section 5.2 of this thesis. Engel et al. [39] provide in their work an excellent overview of many DVR algorithms. Nowadays, two main approaches exist to render the volumetric data, i.e., slicing and ray casting. Our SVPU method was developed as an extension to DVR. We demonstrate on two selected DVR methods that our SVPU algorithm can work in combination with both types of approaches. A seminal paper on displaying of iso-surfaces from volumetric data was presented by Levoy [85]. This image-based DVR method is based on ray casting and we show our results with the SVPU algorithm using this method. Directional occlusion shading (DOS) was proposed by Schott et al. [128] as an interactive method that can render a volume with an illumination similar to full ambient occlusion computation. The DOS method is based on slicing. We present our results also with the DOS method.

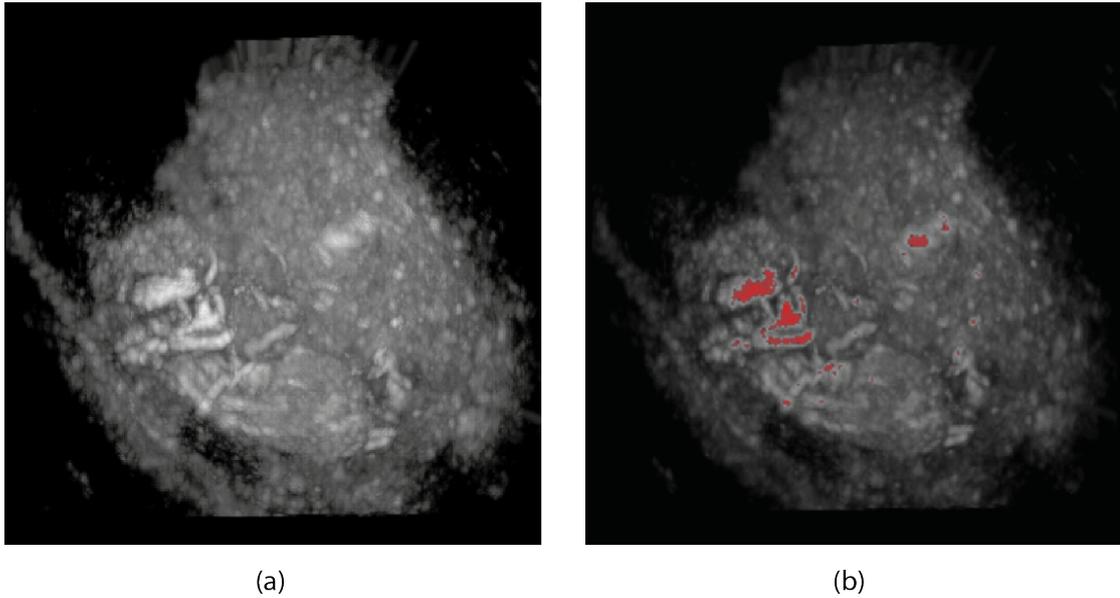


Figure 4.3: Detection of initial points. (a) Maximum Intensity Projection of a typical dataset, and (b) identified initial points displayed in red.

4.3 The Algorithm of Smart Visibility for Prenatal Ultrasound

Considering characteristics of the US data of the scanned human fetus, we can differentiate between several major types of tissues that are distinguishable in the data. Data includes usually surrounding tissues that are in the extent of the scanned region of the probe. A significant part of the space where the human fetus is located is usually filled by amniotic fluid. Depending on the period of the pregnancy, either the whole body of the fetus or a part of it is present in the scanned data. US has a relatively low signal-to-noise ratio (see Section 2.5) and the quality of the signal is usually degraded by several types of artifacts.

As mentioned in Section 2.9, bones of the fetus can be visualized from the 13th week of pregnancy. When the fetus is present in the scanned region, we notice that the bone tissue in its body is given by high intensities. The skull of the fetus and other bones of its skeleton are clearly distinguishable from other tissues by their high intensities. Based on this indicator, we recognize the presence of the fetus in the scanned data.

We propose the SVPU method that automatically gives an unobstructed view on the fetus. The method is based on image-based DVR. In image-based DVR for each pixel of the image a ray is traversed through the 3D ultrasound data and visual contributions are accumulated along the ray. A transfer function assigns color and opacity to each data value. Amniotic fluid can be easily eliminated through the transfer function setup. Amniotic fluid has distinguishable low intensity values, and these are made transparent. Uninteresting outer structures cannot be separated from interesting structures through the transfer function as both have similar intensity

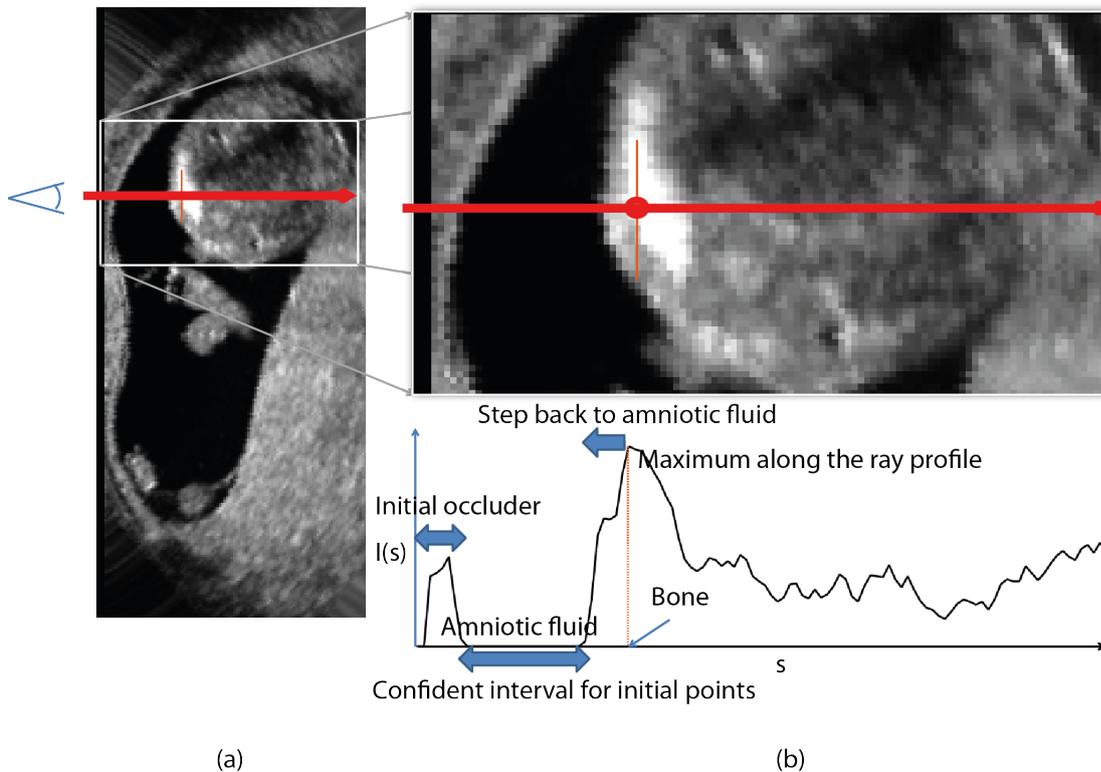


Figure 4.4: Ray profile analysis. (a) A typical slice through the ultrasound dataset. (b) Ray profile through the dataset illustrating the maximum intensity value identified in the bone tissue and the range of interval for positioning of initial points in the the amniotic fluid.

values. Here we need an automatically calculated clipping surface that removes the uninteresting structures and uncovers the interesting structures behind.

Our algorithm controls the start of the rays based on a clipping surface which is derived from initial points. Initial points are detected by a Maximum Intensity Projection (MIP), rendered from the observer's perspective. First, we calculate the Maximum Intensity Projection (MIP) image and store for each pixel the depth information to the maximum and the maximal intensity value along the corresponding ray. Figure 4.3(a) shows the MIP of a typical dataset.

We threshold the MIP image, so only very high intensity pixels are considered to have hit bone inside the fetus. This leaves only a few seed pixels where we know that we have hit the fetus. The detected initial points are shown in Figure 4.3(b).

For the seed pixels we step back from the depth of the bone tissue sample along the ray by a small distance, i.e., the depth information should now indicate a position somewhere in the amniotic fluid in front of the interesting structures and behind the uninteresting occluding structures (see in Figure 4.4). These positions are good anchor points, so called *initial points*, for the clipping surface.

Due to thresholding of the MIP image, only few pixels correspond to initial points. Iteratively we do a construction of the clipping surface. The depth values of seed pixels are propagated to neighboring pixels by iterative local filtering (convolution-based approach) until the entire image plane is filled. This gives a clipping surface which separates uninteresting from interesting structures. As the clipping surface might cut through the fetus at a few places, silhouettes of missing structures can be rendered in the final image.

The algorithm is conceptually divided into three main steps. In the first step the initial points are identified based on the MIP. The clipping surface where the rays start is constructed in the second step with the iterative local filtering of depth information. In the third step silhouettes of missing structures are rendered to preserve the context of undetected parts.

Detection of Initial Points

In a typical scenario the fetus is located in the womb with the occluder that hinders the visibility from the viewer's perspective. We try to find a set of points that serve as a rough estimate of the position of the fetus. It is done by the analysis of the ray profile which intersects the bone tissue. A typical ray profile through the dataset from the observer's perspective is shown in Figure 4.4.

The maximum intensity sample along each ray is defined as:

$$I_M(u, v) = \max(I(s_{i-1}(u, v)), I(s_i(u, v))), i \in [0, N], \quad (4.1)$$

where $s_i(u, v)$ are samples along each ray profile taken with equidistant interval $\Delta s = s_i - s_{i-1}$ for each pixel of the image with coordinates (u, v) . Total number of samples taken along each ray profile is N . The maximal value $I_M(u, v)$ along each ray profile is found by evaluating samples taken along rays from the position of the observer. Figure 4.5 illustrates all important parameters for the ray profile analysis. Distance of samples with maximal intensities from the observer are denoted by $D_M(u, v)$. The distance of the sample from the observer corresponds to the *depth value* of the sample. After finding of the maximal intensity for each ray profile, we store a MIP image $I_M(u, v)$. Together with the MIP values we store corresponding depth values of maximal intensities defined by $D_M(u, v)$.

Position of samples with maximal intensities correspond to the bone tissue of the fetus. Positions of maximal samples are located inside the tissue of the fetus and therefore we take a step back along the ray to a position before the fetus but after the occluder. This depth position is usually somewhere in the interval of the amniotic fluid.

The *interval of amniotic fluid* defines a range of depth values for initial points that are suitable for starting the ray. An interval $F(u, v)$ is identified along each ray between the occluder and the fetus:

$$F(u, v) = \{D_P(u, v) : P_{FO}(u, v) \leq D_P(u, v) \leq P_{FI}(u, v); \\ I(D_P(u, v)) \leq T_F; D_P(u, v) \leq D_M(u, v)\} \quad (4.2)$$

where $D_P(u, v)$ are depth values along ray profiles in the fluid interval. The interval of amniotic fluid corresponds to $P_{FO}(u, v) \leq D_P(u, v) \leq P_{FI}(u, v)$. The interval of amniotic fluid is in front of the fetus and behind the occluder. The interval does not contain any other occluder. That is guaranteed by a stepwise evaluation of samples along ray profiles during the ray analysis.

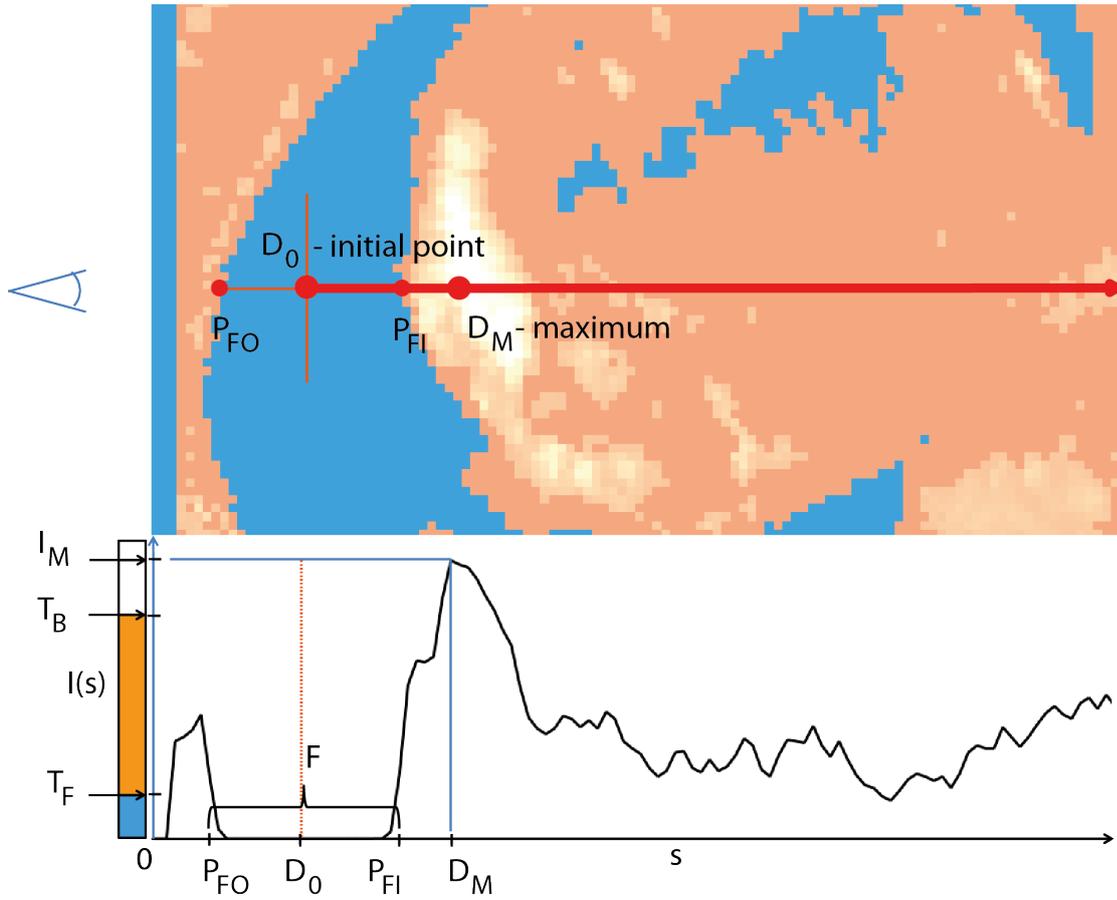


Figure 4.5: Parameters for ray profile analysis. A typical slice through the ultrasound dataset with classified tissues (blue - amniotic fluid, orange - fetus and surrounding tissue, white - bone tissue). The position of an initial point D_0 is in the interval F between P_{FO} and P_{FI} before the detected bone D_M of the fetus. Point is selected only if the maximal intensity value I_M along the ray profile satisfies the condition $I_M \geq T_B$.

Samples are evaluated with an equidistant sampling distance $\Delta s = s_i - s_{i-1}$ during the ray profile analysis. Boundaries of the interval are found by the thresholding $I(D_P(u, v)) \leq T_F$ on the step back from positions $D_M(u, v)$. Positions $D_M(u, v)$ are depth values of the MIP image which correspond to the bone tissue. The threshold T_F for entering and exiting of the amniotic fluid is defined by the transfer function applied also for DVR. Typically a simple 1D opacity transfer function is applied. In medical imaging this function is also called *windowing function*. The windowing function is defined with the *center* C_{WF} and *width* W_{WF} . The threshold of amniotic fluid T_F is given by:

$$T_F = C_{WF} - W_{WF}/2. \quad (4.3)$$

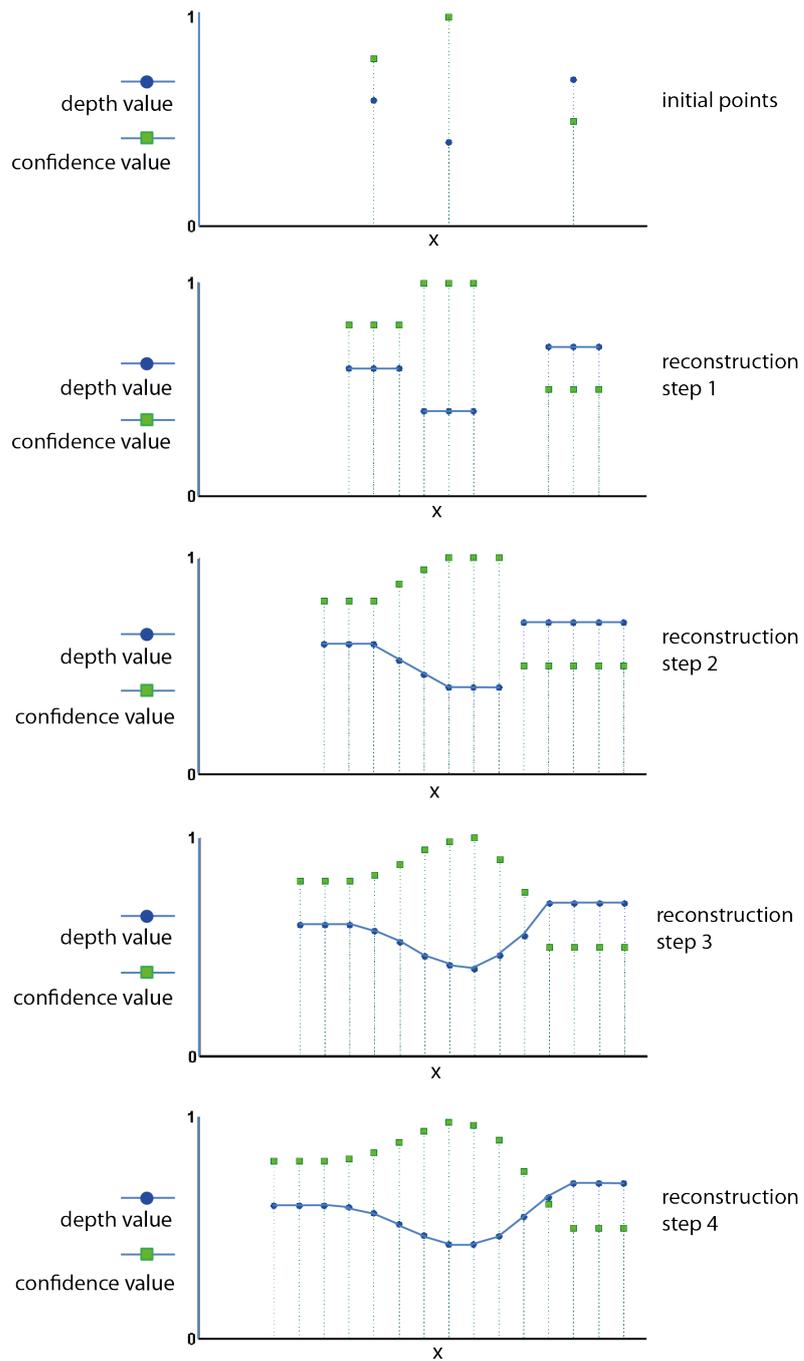


Figure 4.6: Illustration of the adaptive weighted average filtering with sparse initial points in 1D. Depth values are weighted with their confidence. Three initial values are reconstructed in four steps. Filter kernel has size of three samples.

The interval of possible positions along the ray profile corresponds to positions that were identified by stepping back from the depth of the bone tissue sample along the ray. The depth information of this interval should now indicate positions in the amniotic fluid in front of the fetus. We need to store only the boundary points $P_{FI}(u, v)$ and $P_{FO}(u, v)$ of the interval of all possible initial points along each ray that are directly in front of the fetus and behind the occluder. The *fluid entry points* are identified on the step back from the positions of the bone. This corresponds to the position that is closest to the boundary between the skin surface of the fetus and the amniotic fluid:

$$P_{FI}(u, v) = \max(F(u, v)). \quad (4.4)$$

We identify another positions in the amniotic fluid in front of the fetus and behind the occluder. These positions are at the boundary between the occluder and the amniotic fluid. We call these position the *fluid exit points*. The fluid exit point corresponds to the point that is farthest away from the fetus:

$$P_{FO}(u, v) = \min(F(u, v)). \quad (4.5)$$

Initial points can be chosen anywhere in the interval of the amniotic fluid:

$$D_P(u, v) = P_{FO}(u, v) + q(P_{FI}(u, v) - P_{FO}(u, v)), \quad 0 \leq q \leq 1, \quad (4.6)$$

where q is a parameter that controls the position of points between the occluder and the fetus. It is reasonable to use the default value $q = 0.5$. In this way we define the initial points to be in the middle of the interval.

Initial points, detected based on the MIP image, contain also outliers. To remove the outliers we threshold the MIP image, so only very high intensity pixels are considered to have hit bone inside the fetus. This leaves only a few seed pixels where we know that we have hit the fetus. We apply a threshold on MIP values, to obtain a mask for samples of the MIP image that do not correspond to bones of the fetus.

$$M_B(u, v) = \begin{cases} true, & \text{if } I_M(u, v) \geq T_B \\ false, & \text{if } I_M(u, v) < T_B \end{cases} \quad (4.7)$$

where T_B is a threshold for bone tissue applied on a MIP value. The MIP image is filtered based on the heuristic measure which we define as a *confidence*. Confidence is a heuristic measure which corresponds to the quality criteria of the detected initial points, such as original intensity values or lengths of intervals in amniotic fluid. When the confidence measure corresponds to the intensity value in the MIP image the threshold T_B is given by:

$$T_B = \max(I_M(u, v)) - T_C \cdot \max(I_M(u, v)), \quad (4.8)$$

where T_C is a confidence threshold. The threshold is selected in an adaptive way, based on the most confident value $\max(I_M(u, v))$. The most confident value is the highest intensity of the MIP image. Confidence threshold is a heuristic parameter in the range $0 < T_C \leq 1$. If $T_C = 1$, all depth values would contribute to the clipping surface construction. However, due to the outliers, it is reasonable to start the clipping surface construction only with points that have a high confidence.

Initial points for the clipping surface construction are defined by their initial depth values:

$$D_0(u, v) = M_B(u, v).D_P(u, v), \quad (4.9)$$

and their initial confidence values:

$$B_0(u, v) = M_B(u, v).I_M(u, v). \quad (4.10)$$

It is reasonable to define confidence corresponding to the intensity value because high intensities indicate positions of fetal bones. However, the length of the interval of amniotic fluid could also be used as confidence in the calculation of the clipping surface.

After determining the initial points, due to the thresholding of the MIP image, only few pixels of the image are covered. The initial points are sparsely located only in regions where bones were detected. Starting rays from the location of the initial points would lead to an unobstructed view of the fetus. Therefore a clipping surface of starting rays has to be constructed.

Surface Construction

The result of the previous step is essentially a sparse two dimensional set of initial points $D_0(u, v)$ with confidence values $B_0(u, v)$. This type of data is also called *scattered data*. As mentioned in Section 4.2, many methods were developed for scattered data interpolation. A lot of methods are computationally expensive and cannot be applied in an interactive system such as an ultrasound machine. We decided to use a simple and effective method that fits well to our application. It is based on a *local filtering* approach [49] [162]. One advantage is that it does not need the connectivity information between the initial points. It also does not lead to a complex systems of linear equations like other methods based on radial basis functions or least squares approaches.

Local filtering can be used for construction of the clipping surface by applying a reconstruction filter kernel. If the filter kernel is the same for the reconstruction of all samples, the reconstruction process is called space invariant local filtering. The reconstructed 1D discrete signal can be expressed as a weighted average of the signal with the reconstruction kernel:

$$c(x) = \frac{\sum_{k=-K}^K f(x_k)h(x - x_k)}{\sum_{k=-K}^K h(x - x_k)}, \quad (4.11)$$

where samples of the signal $f(x_k)$ are weighted with the same kernel $h(x - x_k)$ of the size K . The sum of the filter kernel weights in the denominator guarantees that the result is *normalized*. This type of filtering is well suited for a uniformly sampled signal on a regular grid.

In case of scattered data, such as in our case, the sampling is nonuniform. In such a case *adaptive local filtering* can be used. It is based on the idea of analyzing the distribution of samples in a small neighborhood around each sample and to estimate the local filter kernel for reconstruction. Local filtering for a 1D signal based on *adaptive weighted average filtering* is

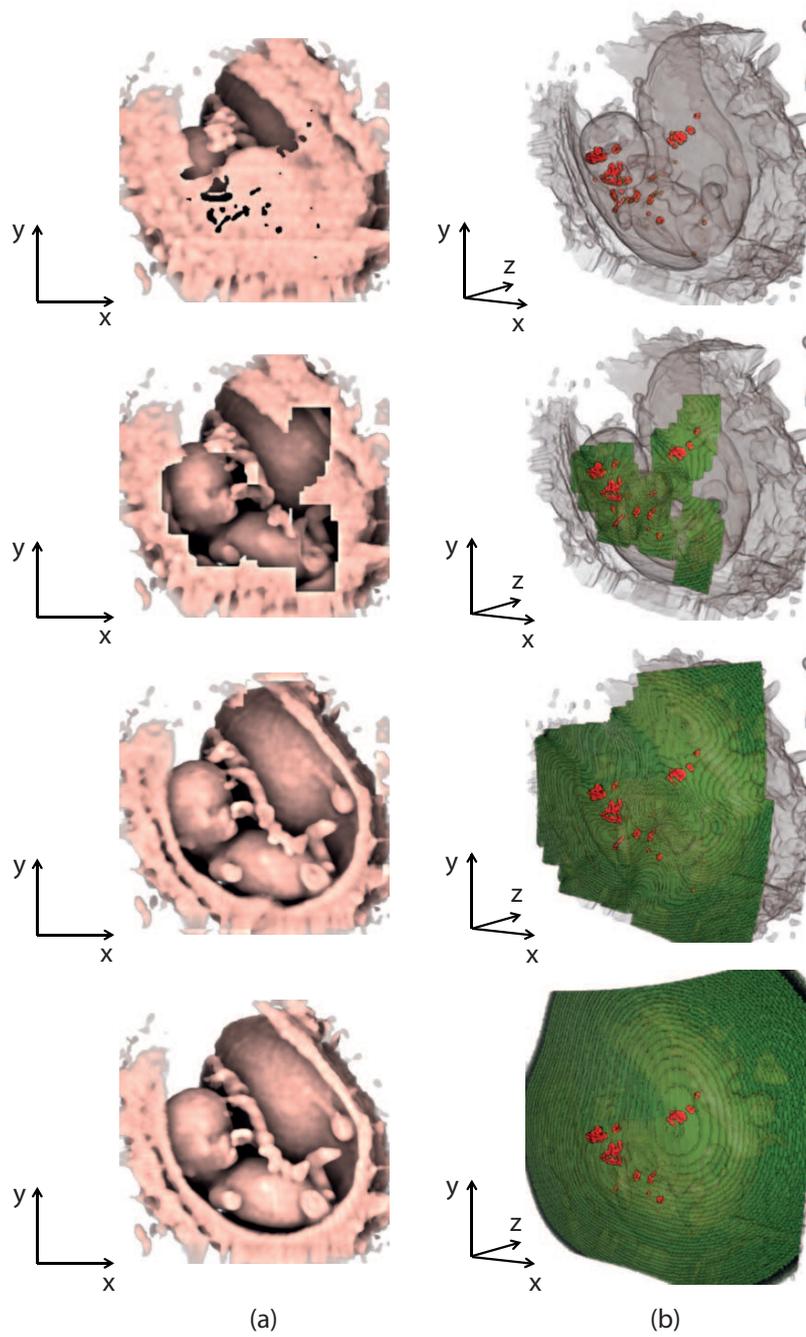


Figure 4.7: Clipping surface construction: (a) rendering with the constructed clipping surface, (b) initial points and successive construction of the surface after 3, 10 and 100 iterations are shown from top to bottom.

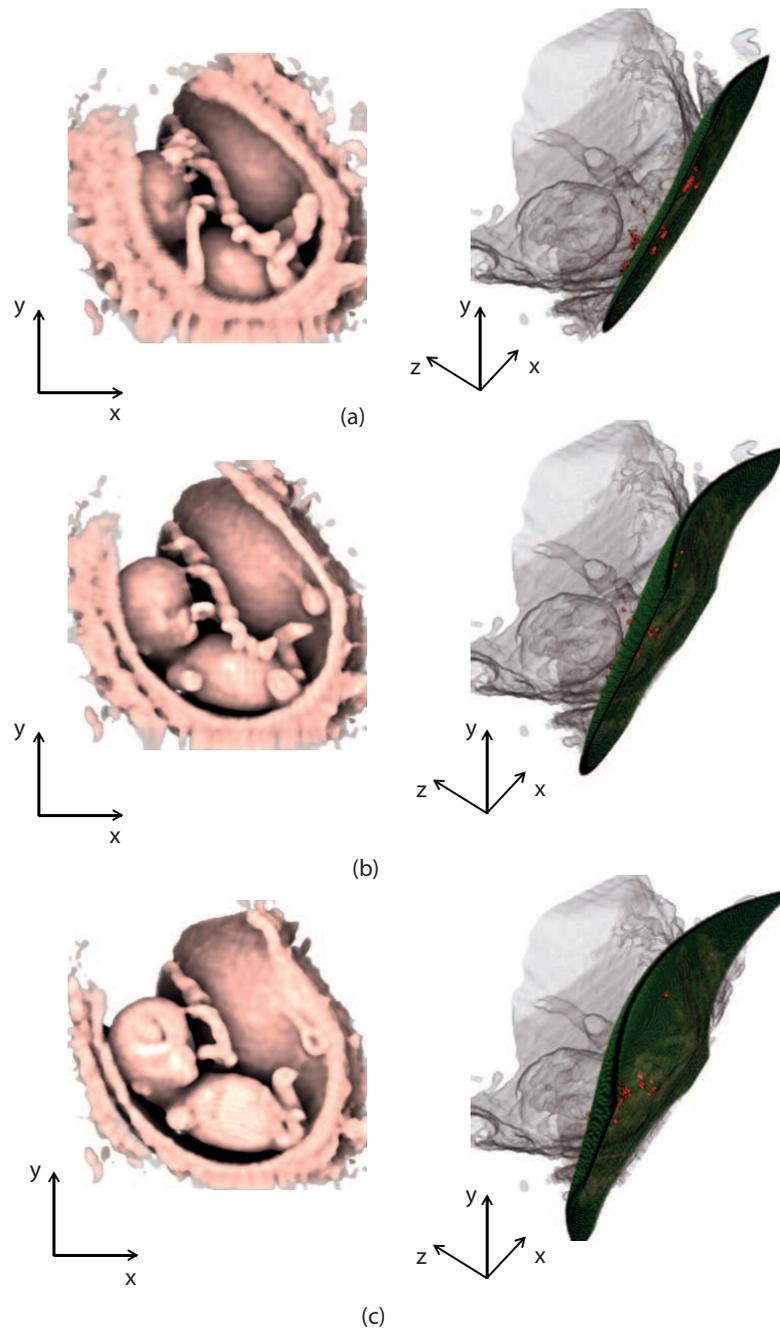


Figure 4.8: Constructed clipping surface with different positions of initial points in the interval of the amniotic fluid (see Equation 4.6): (a) $q = 0$ - far from the fetus, (b) $q = 0.5$ - in the middle of the amniotic fluid, and (c) $q = 1.0$ - close to the fetus.

given by:

$$c_A(x) = \frac{\sum_{k=-K}^K f(x_k)h_A(x - x_k)}{\sum_{k=-K}^K h_A(x - x_k)}, \quad (4.12)$$

where samples of the signal $f(x_k)$ are weighted with the different kernels $h_A(x - x_k)$. We apply a 2D version of this type of filtering in our scenario in an iterative fashion for the clipping surface construction. We propose a heuristic method that uses confidence values of local neighborhood samples as weights for the filter kernel. Confidence values are derived from the data by the algorithm for finding initial points.

After finding the initial points, for each point the depth value $D_0(u, v)$ is stored which corresponds to a distance to the viewer. Additionally to depth values, we store the confidence value $B_0(u, v)$ for each point. Initial confidence values at each point correspond to the original value of the MIP image. These confidence values serve as weights for the adaptive weighted average filtering of depth values.

The clipping surface is constructed by iterative local filtering of depth values. In this way, the missing pixels are successively reconstructed from samples in a local neighborhood. The filtering is computed as adaptive weighted average for each sample by:

$$D_i(u, v) = \frac{\sum_{k=-K}^K \sum_{l=-K}^K D_{i-1}(u_k, v_l) \cdot B_{i-1}(u - u_k, v - v_l)}{\sum_{k=-K}^K \sum_{l=-K}^K B_{i-1}(u - u_k, v - v_l)}, \quad (4.13)$$

where $D_i(u, v)$ is a *new* depth value computed by adaptive weighted average from the *previous* samples from $D_{i-1}(u_k, v_l)$ and $B_{i-1}(u - u_k, v - v_l)$ in a local neighborhood K around the sample:

$$-K \leq k, l \leq K. \quad (4.14)$$

The parameter K is called the kernel size. This averaging filter is applied *iteratively*, to spread the depth information across the image and fill the gaps of missing values. After several iterations, the extent of the whole rendered image is reached. In parallel to the construction of the clipping surface, the confidence map is constructed by iterative adaptive weighted average as well:

$$B_i(u, v) = \frac{\sum_{k=-K}^K \sum_{l=-K}^K B_{i-1}(u_k, v_l) \cdot B_{i-1}(u - u_k, v - v_l)}{\sum_{k=-K}^K \sum_{l=-K}^K B_{i-1}(u - u_k, v - v_l)}. \quad (4.15)$$

The size of the filter kernel corresponds to the number of samples that are taken from the local neighborhood around a sample. The kernel is different for each reconstructed sample point. Samples without any depth value have initially zero confidence. The number of iterations is chosen according to the size of the rendered image. Number of iterations and kernel size have

an influence on the smoothness of the clipping surface that is constructed. After the construction, a smooth clipping surface together with a constructed confidence map results.

Figure 4.6 illustrates the reconstruction of a 1D function with the adaptive weighted average filtering. The filter computes every new sample from three previous samples weighted by their confidence value. Depth values and confidence values are normalized. The function is reconstructed after four iterations.

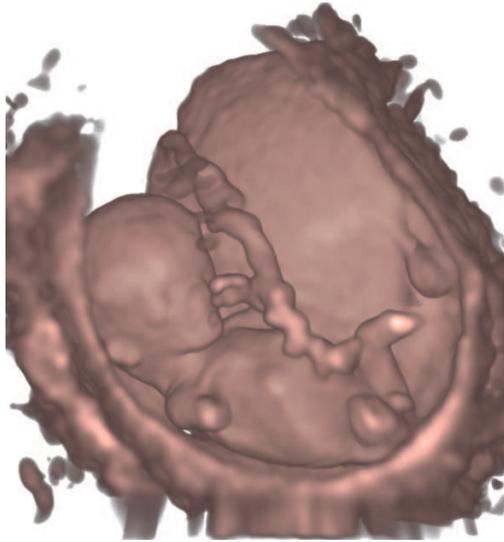
The process of construction of a clipping surface is illustrated in Figure 4.7(a). Initial points are depicted in yellow and the constructed clipping surface is shown in green. Iso-surface rendering with transparency is used to display the fetus in the context. Several iteration steps show the successive forming of the curved surface. The corresponding renderings in Figure 4.7(b) show the successive removal of the occluder in front of the fetus.

Clipping surface construction with different positions of initial points is shown in Figure 4.8. Images are rendered with local illumination. In Figure 4.8(a) points are defined to be far from the fetus. Rendering with the constructed clipping surface shows more of the view obstructing tissues. In Figure 4.8(b) points are positioned in the center of the amniotic fluid interval. The constructed clipping surface cuts away the limb and reveals the head of the fetus. In Figure 4.8(c) the initial points are located at a close distance to the fetus and the clipping surface is aggressively cutting into the tissue of the fetus.

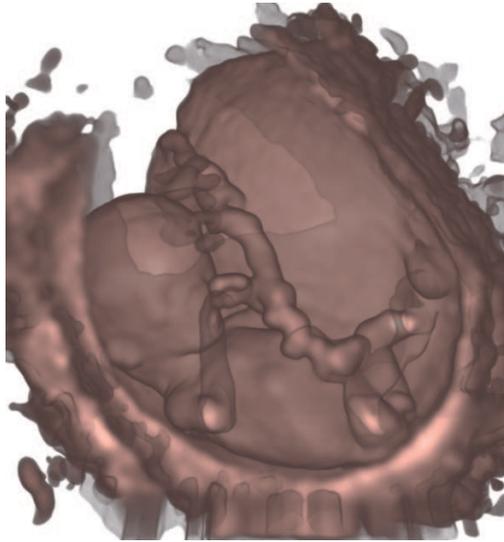
Silhouette Rendering with Opacity Modulation

The constructed clipping surface is provided to the DVR algorithm for rendering. It classifies the volume data into two classes, i.e., a visible part with interesting structures and an invisible part with occluding tissue. Rendering of the visible part of the volume might result into images of the fetus with editing artifacts. Editing artifacts can appear because the constructed clipping surface cuts through some parts of the tissue that would be otherwise important and should be displayed. This can happen for example at regions with extremities of the fetus that are not detected because the intensity of the ultrasound echo from the developing bone tissue is not very strong. Therefore bones of these fetal body parts can be missed in the detection step of the initial points which are in the following step used for the clipping surface reconstruction. The constructed clipping surface is then starting deeper inside the volume and cuts off part of the relevant tissue. This situation is shown in Figure 4.9(a) and in Figure 4.10(a), where the right hand of the fetus is not visible because it is not the part of the visible volume. In such case, as discussed in Section 4.2, ghosting techniques with rendering of ghost lines can be applied. In the SVPU algorithm, we apply ghost lines which we call *silhouettes*. Rendering of silhouettes is used in order to show outlines of missing structures in the context of the rendered image.

For rendering of silhouettes, we apply the iso-surface rendering as it was proposed in the original work of Levoy [85]. This method renders iso-surfaces with constant thickness using a DVR algorithm. Iso-surfaces are made transparent with the opacity modulation applied to the samples. Silhouettes appear implicitly because some rays attenuate more light than others. Rays that travel perpendicular to the transparent iso-surface normal attenuate more light than rays traveling tangential to the transparent iso-surface normal. Opacity modulation for iso-surface



(a)



(b)

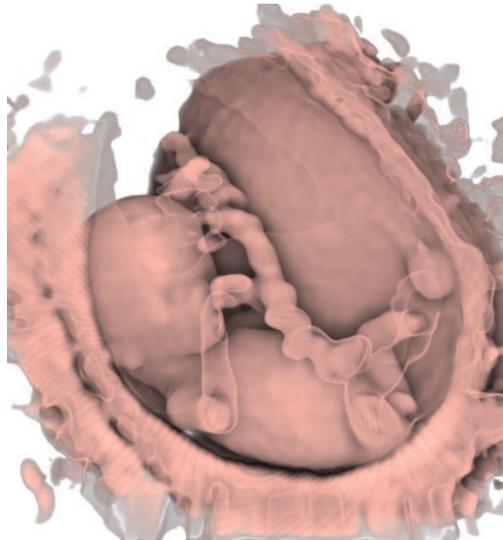
Figure 4.9: SVPU rendering modes. DVR with local illumination (a) without and (b) with silhouette rendering with opacity modulation.

rendering is defined as:

$$\alpha(s_i) = \alpha_v \begin{cases} 1 - \frac{1}{r} \frac{|T_I - I(s_i)|}{|\nabla s_i|}, & \text{if } |\nabla s_i| > 0 \\ & \text{and} \\ & I(s_i) - r|\nabla s_i| \leq T_I \leq I(s_i) + r|\nabla s_i| \\ 0, & \text{otherwise.} \end{cases} \quad (4.16)$$



(a)



(b)

Figure 4.10: SVPU rendering modes. DVR using DOS (a) without and (b) with silhouette rendering with opacity modulation.

The method of rendering iso-surfaces requires a specific iso-value T_I as a parameter. This parameter has the same value as defined in Equation 4.3. At each sample s_i a gradient magnitude value $|\nabla s_i|$ is computed that is used to control the opacity modulation. The transparency of the iso-surface is controlled with the parameter α_v . Lower quality iso-surfaces, which are present in ultrasound data because of a low signal-to-noise ratio, are enhanced with parameter r that modulates the *thickness* of the iso-surface. This parameter guarantees a constant thickness of the

transition region of the iso-surface by having opacity fall off if moving away from the iso-value. With this modulation we can achieve a semi-transparent display of iso-surfaces that enhances contours of objects. They create the effect of silhouettes of missing features on the final images.

The iso-value T_I corresponds to the fetal skin surface. A simple windowing function defines the iso-value $T_I = T_F$ as in Equation 4.3. The same windowing function is also used as a transfer function for mapping of opacities for the DVR of the soft tissues. The parameter T_F has the same value as the threshold that is used for the detection of the boundary between the fetal skin surface and the amniotic fluid defined in Equation 4.2. At the rendering stage, we combine the opacity modulation for rendering of iso-surfaces together with the DVR algorithm that is used for the rendering of the soft tissues (see Figure 4.9(b) and Figure 4.10(b)). We render the silhouettes only for parts of the volume data that were classified by clipping surface as occluding tissue. This provides an impression of outlines of missing anatomical structures with SVPU algorithm and makes the algorithm more robust to editing artifacts.

4.4 Implementation

We implement our smart visibility algorithm in C++ using CUDA. It is an extension of the standard direct volume rendering method which is nowadays the conventional approach for volume visualization. This section is discussing further details of our implementation.

In the beginning, one pass with ray casting through the volume is made, to obtain the initial points from the observer's perspective. Every ray is implemented as a kernel function that runs in parallel on the GPU. For each ray, the depth value of the initial point is identified together with the maximum intensity value which corresponds to the confidence. The depth value is computed according to the Equation 4.6 from the last entry and exit of the amniotic fluid before the maximum intensity value. The threshold for amniotic fluid T_F is defined by a simple transfer function for opacity mapping. Initial points are filtered out by applying a threshold to their confidence value. By default, we take only values that have their intensity 25% lower ($T_C = 0.25$) than the highest intensity value of the MIP image. This is a heuristics which generated stable results in all of the tested datasets. This parameter can be modified if necessary in order to include more initial points for the clipping surface construction.

We store the initial depth values and confidence values in two separate 2D buffers. Adaptive weighted average filtering of depth values is implemented on the GPU with the iterative multi-pass approach. For this purpose we allocate two buffers for depth values and confidence values that are used in each iteration. We initialize these buffers with the values that were identified after finding of initial points. In each iteration step one set of buffers is accessed for reading and another set is accessed for writing. Adaptive weighted average filtering is implemented as a kernel function that samples depth and confidence buffers accessed for reading and writes the new average value into the the buffers accessed for writing as defined in Equation 4.13 and Equation 4.15. After each iteration step, buffers are switched and the averaging operation is repeated until the whole image is covered. For all presented images, with a resolution around 512×512 pixels, we repeat 120 iterations with a filter kernel size $K = 7$. This number of iteration is required for the high resolution volume of $512 \times 512 \times 512$ voxels. SVPU is an image based algorithm that is dependent on the image resolution. In ultrasound rendering, parallel

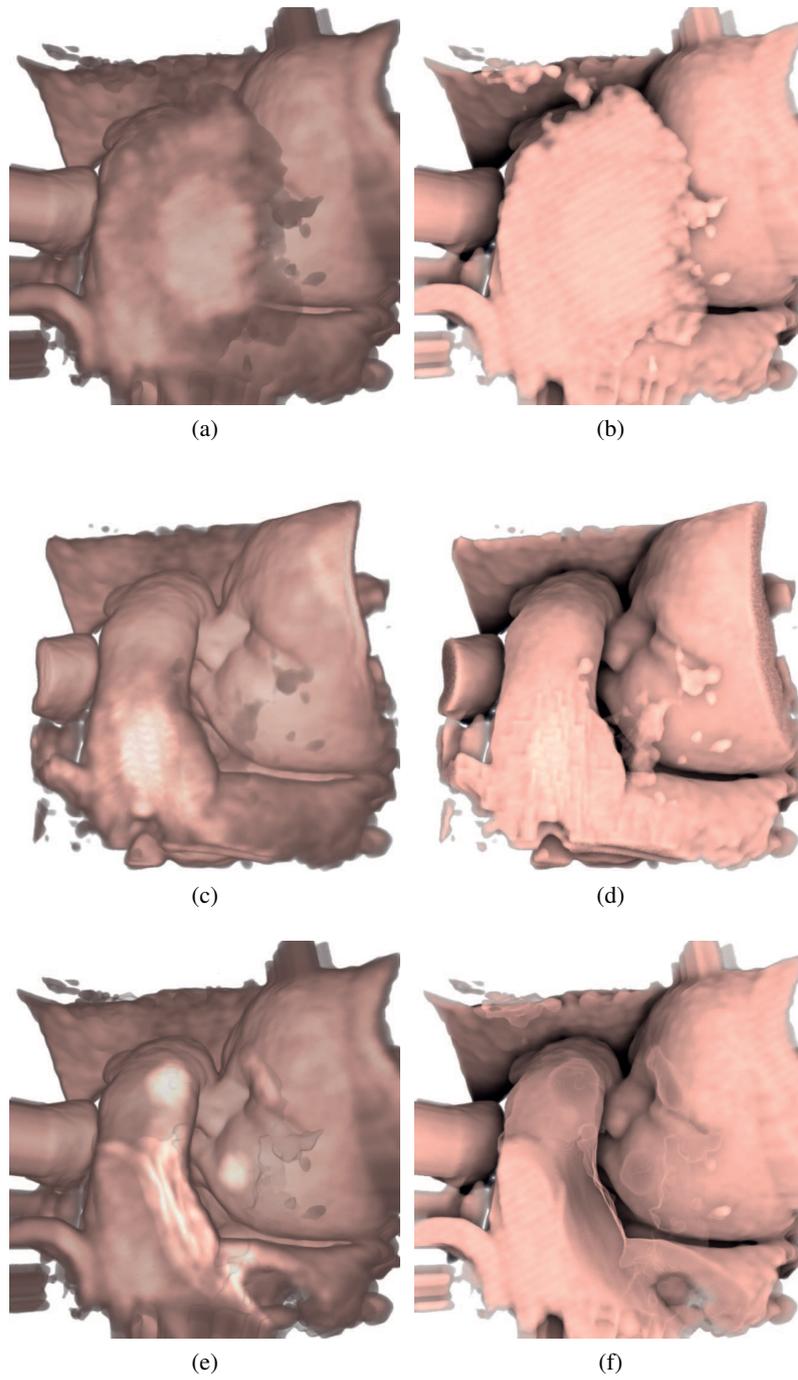


Figure 4.11: Comparison of results for 'Dataset 1' - easy category. DVR with local illumination and DOS. (a)(b) original data, (c)(d) manual ROI box, (e)(f) SVPU algorithm.

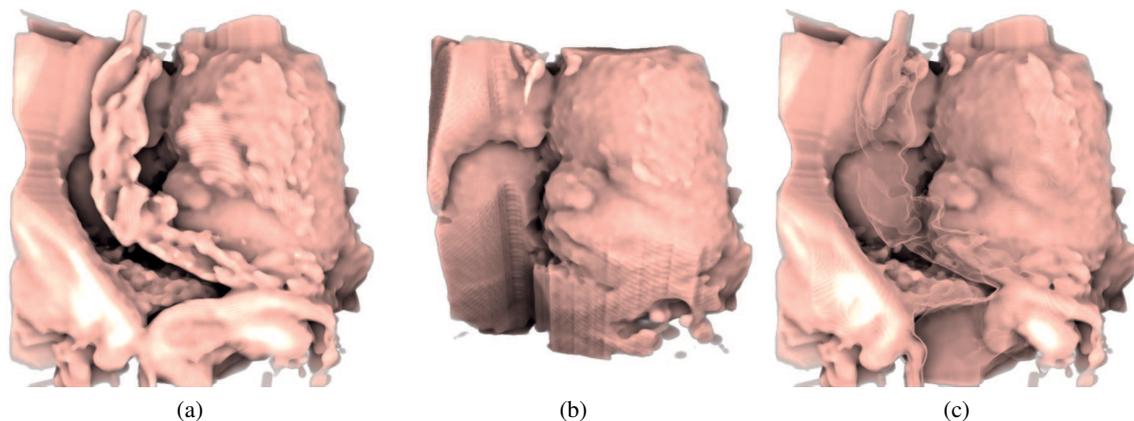


Figure 4.12: Comparison of results for 'Dataset 2' - easy category. (a) original data - DVR with DOS, (b) 'electric scalpel' - DVR with DOS, (c) SVPU algorithm - DVR with DOS.

projection is used for the DVR. This means that the traversed rays are parallel to each other and the image resolution depends on the resolution of the volume. Parallel projection is used in medical imaging in order to allow measurements on rendered images.

Silhouette rendering with opacity modulation is implemented directly in the rendering step. For comparison, we implemented two different rendering modes in our prototype. Simple direct volume rendering (DVR) with local illumination is implemented. Additionally to simple DVR, we implemented directional occlusion shading (DOS) which is a global illumination method. For comparison, we show the effect of the silhouette rendering with opacity modulation in Figure 4.9. Body parts that were accidentally cut away by the clipping surface are visible when the effect is applied. Silhouette rendering with opacity modulation is applied when opacity mapping for each sample along a ray is performed. We use front-to-back compositing of sample contributions to compute the final value of the rendered pixel. When the depth of the constructed clipping surface is reached, we employ the compositing of the samples for DVR of soft tissues mapped by the transfer function. By extending two different DVR algorithms we demonstrate that SVPU can easily be applied to existing DVR methods with only minor modifications to their implementation.

Our prototype was implemented on the GPU with interactive frame rates. The performance of the renderer with the SVPU algorithm is comparable to standard direct volume rendering. However, we can notice a performance drop caused by the additional rendering pass that is necessary for finding initial points and for the clipping surface construction. First, we implemented our prototype on the NVIDIA GeForce GTX 580 GPU with 3072 MB GDDR 5 dedicated video memory and Intel Core i7 CPU 3.07 GHz with 6 GB of RAM. Although the prototype of the algorithm could achieve real-time performance on the powerful desktop GPU, we had to consider the hardware components of the latest ultrasound machine and their performance.

In the next stage we integrated the prototype of the smart visibility algorithm into our pipeline that was discussed in the previous chapter. Our pipeline consists of several modules

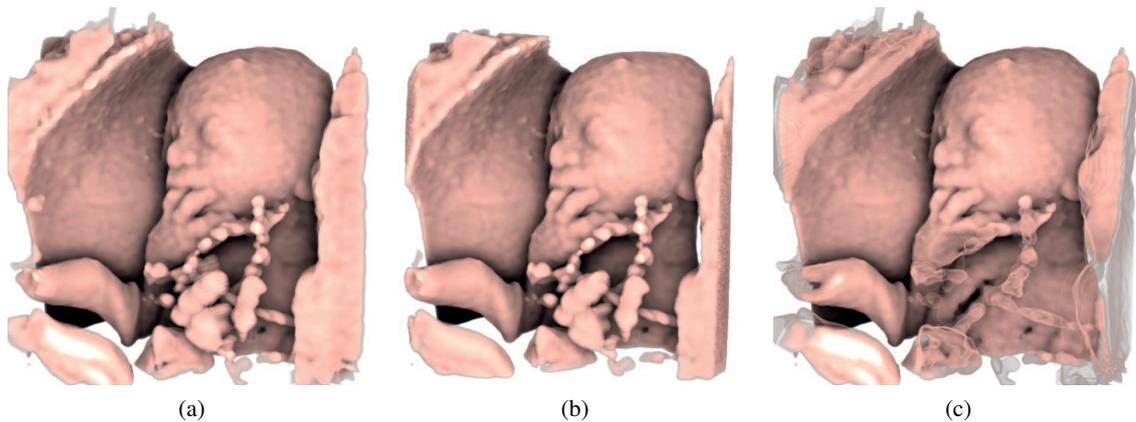


Figure 4.13: Comparison of results for 'Dataset 3' - easy category. (a) original data - DVR with DOS, (b) manual ROI box - DVR with DOS, (c) SVPU algorithm - DVR with DOS.

with independent functionality. The data flows through each module of the pipeline in slices which are organized in slabs. Thus, modules do not have access to the full data and when they are ready to process a portion of the data, they issue a request to the module connected to their input. For performance reasons we decided to use a lower resolution version of the volume for finding of initial points with ray analysis and clipping surface construction. The size of the volume is approximately $100 \times 100 \times 100$ voxels. The high resolution volume is used only for rendering. This is because of the computationally expensive scan-conversion algorithm that needs to be performed twice in our pipeline when the clipping surface is computed. First for the ray analysis, and second for the rendering of the data. It is assumed that the clipping surface will have low frequency. Therefore, we are able to perform the computation of the SVPU algorithm in our pipeline with the lower resolution data. It allows to decrease the computational load of the algorithm and the time of the scan-conversion computation. It also allows to perform the analysis in one pass through the volume. We are able to decrease the size of the clipping surface construction filter kernel and number of iterations that are required for the clipping surface construction. We notice that this modification has no significant impact on the quality of the achieved results when compared to the high resolution volume. In the following section, we present the results of our SVPU algorithm.

4.5 Results

The tested datasets were provided and organized into three category levels (easy, medium and difficult) by GE Healthcare. For all datasets, we had also regions of interest manually defined by domain experts. In comparison, we show the original data with occlusion, the rendering with manually defined ROI and the results with the smart visibility algorithm.

Easy category: Figure 4.11 shows the dataset with an occluder in front of the face of the fetus surrounded by a lot of amniotic fluid. The SVPU algorithm uncovers the face by cutting

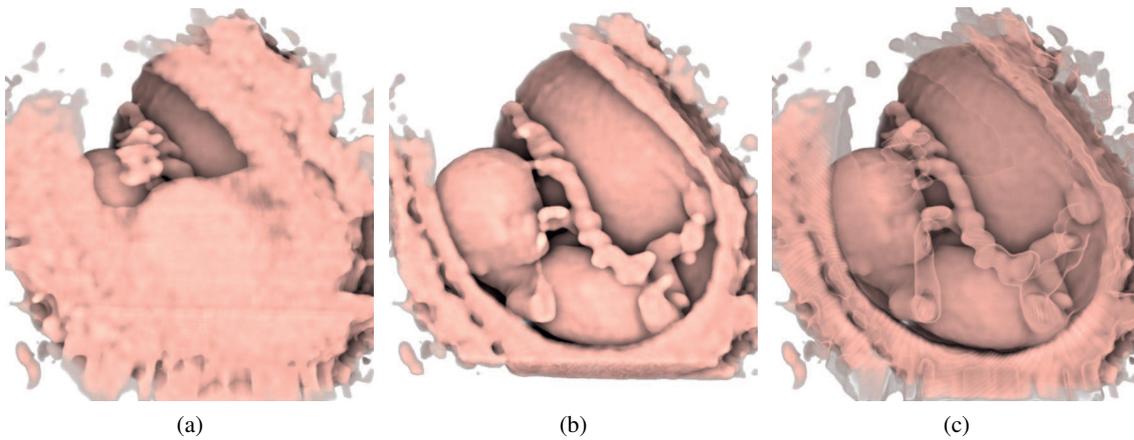


Figure 4.14: Comparison of results for 'Dataset 4' - easy category. (a) original data - DVR with DOS, (b) manual ROI box - DVR with DOS, (c) SVPU algorithm - DVR with DOS.

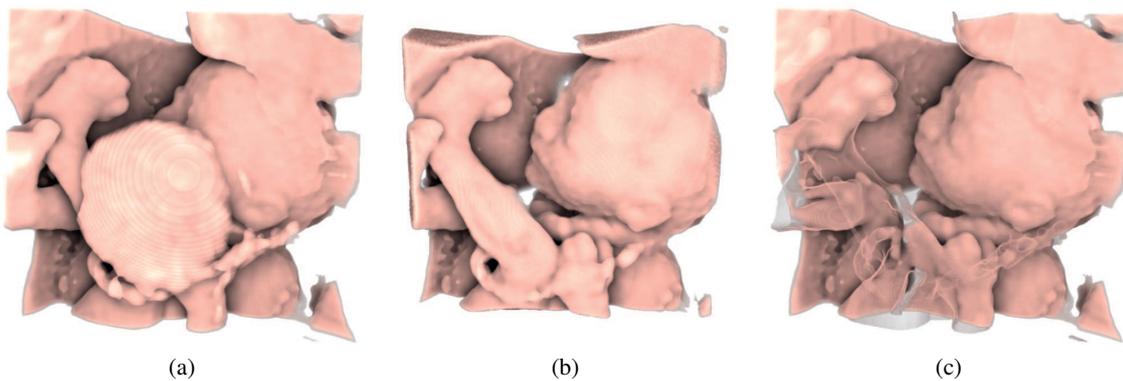


Figure 4.15: Comparison of results for 'Dataset 5' - medium category. (a) original data - DVR with DOS, (b) 'electric scalpel' - DVR with DOS, (c) SVPU algorithm - DVR with DOS.

away the occluding tissue. Images are rendered in both implemented rendering modes just for comparison. The DOS rendering mode provides better depth perception in comparison to local illumination that is used for the rendering of the images in Figure 4.11. Figure 4.12 shows a fetus that is occluded by the umbilical cord and some part of the other surrounding tissue. In order to show the face of the fetus, the operator had to use the 'electric scalpel'. Our method displays the face of the fetus without the umbilical cord. The contours of it are still visible because of the silhouette rendering with opacity modulation. A dataset without any occluder in front of the face is shown in Figure 4.13. The SVPU method is capable to recognize and to show the face without any difficulties also in this scenario. The fetus is fully occluded in the dataset of Figure 4.14(a). The rendering with the manually defined ROI (see Figure 4.14(b)) shows the whole body of the fetus. A similar quality of the rendering is achieved with the SVPU algorithm

(see Figure Figure 4.14). The limbs of the fetus that were cut away by the clipping surface is visible because of the silhouette rendering with opacity modulation.

Medium category: The fetus in the dataset shown in Figure 4.15 is quite well defined. Our algorithm shows the face without difficulties. However, together with the occluder it cuts away part of the limb that was not recognized in the step of finding initial points and clipping surface causes an editing artifact. The contours of the limb are still visible because of the silhouette rendering with opacity modulation (see Figure 4.15(c)). The face of the fetus shown in Figure 4.16 is occluded by the tissue in front of it. It is displayed together with the umbilical cord and the hand that is touching the face by SVPU algorithm. The images (see Figure 4.16(a)(c)(e)) for this dataset are rendered also with local illumination just for comparison. Notice the editing artifacts, which appear as a missing part of the hand, which are caused by excessive use of the 'electric scalpel' tool. It was difficult to apply the 'electric scalpel' tool in this case. Figure 4.17 shows a dataset with a bad contrast and a small gap between the fetus and the occluder. It is also difficult to define the ROI box in such a case. The results in this case demonstrates that of the SVPU algorithm works also for noisy data where the fetus is not very well defined. However, we can notice that the editing artifact has appeared in the SVPU result.

Hard category: An unusual rendering direction of the dataset depicted in Figure 4.18 makes the manual specification of the ROI cumbersome. Therefore, the 'electric scalpel' was used to manually remove some parts of the volume. The SVPU algorithm can handle also this scenario and provides an unobstructed view on the head of the fetus.

In addition to static volumes, we were provided with four sequences with 3D temporal volume data where the fetus was changing position (see Figure 4.19 and Figure 4.20). In this way it was possible to test the stability of our algorithm. Automatic classification of the temporal data with the application of our SVPU algorithm simulates live scans with the data streamed on-the-fly from the transducer. Each sequence had on average 30 frames in total. The face of the fetus was reliably detected in all sequences. The results achieved with our SVPU method demonstrate the potential of the method to minimize the user interaction with the ultrasound machine during examinations.

4.6 Conclusion

The human fetus scanned in the prenatal period is typically occluded by surrounding tissue. Therefore, in 3D renderings the fetus is often not visible. The current solution requires the manual definition of the ROI or manual processing with the 'electric scalpel' by the operator. This task becomes very difficult in live scans because of many factors. In this chapter we proposed a novel method that can automatically visualize the fetus without occluders. Smart visibility for prenatal ultrasound is an algorithm capable to visualize the face of the fetus from the volume data generated by ultrasound. It has been developed in order to work automatically, without additional parameter modification. This method provides a simple extension to the volume rendering pipeline of the ultrasound machine. We have tested it on a variety of real datasets during the development. The SVPU algorithm works reliably on the provided static volumes and streamed volume data as well. However, the algorithm has to be further tested on more data sets. In general, the method has the potential to decrease the amount of interaction by the operators while

performing the sonography. The decreased amount of time required for scanning could increase the comfort for the patients.

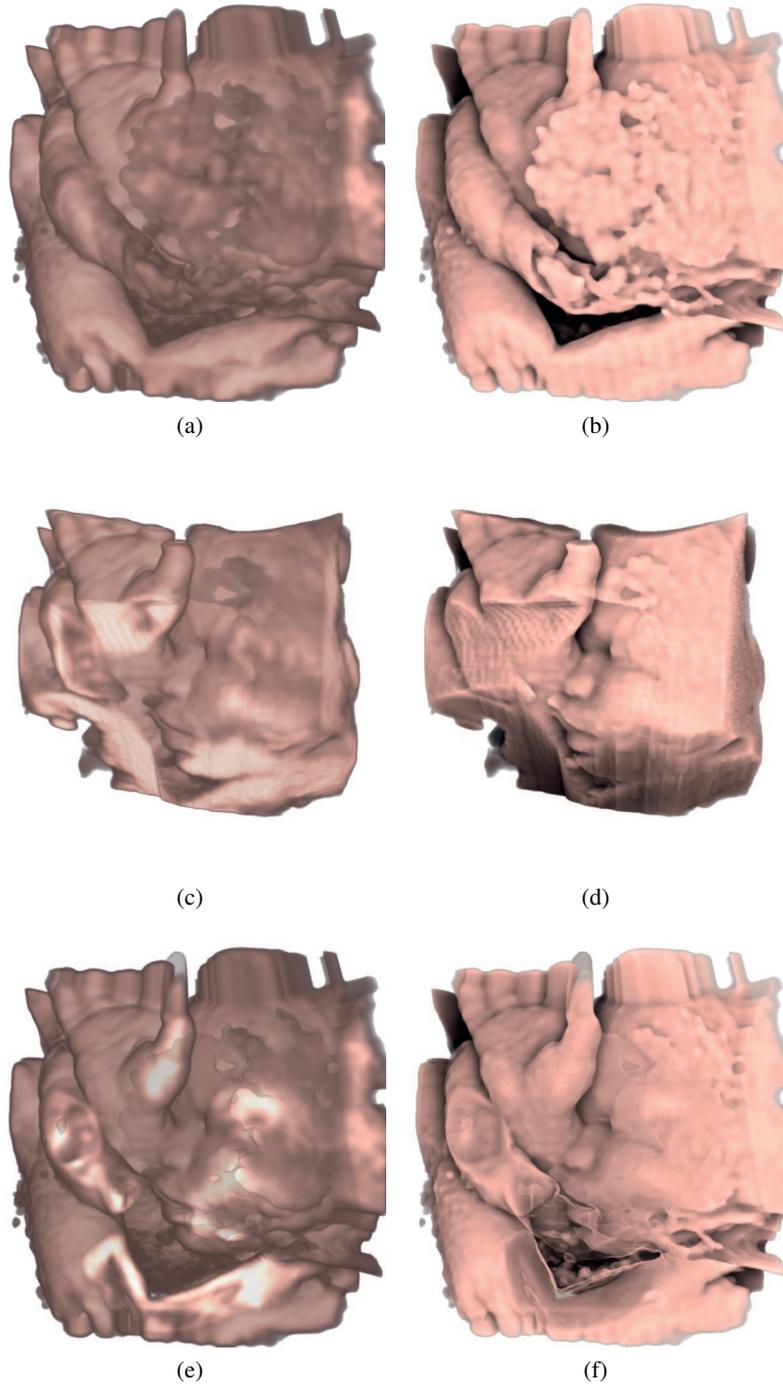


Figure 4.16: Comparison of results for 'Dataset 6' - medium category. DVR with local illumination and DOS. (a)(b) original data, (c)(d) manual ROI box, (e)(f) SVPU algorithm.

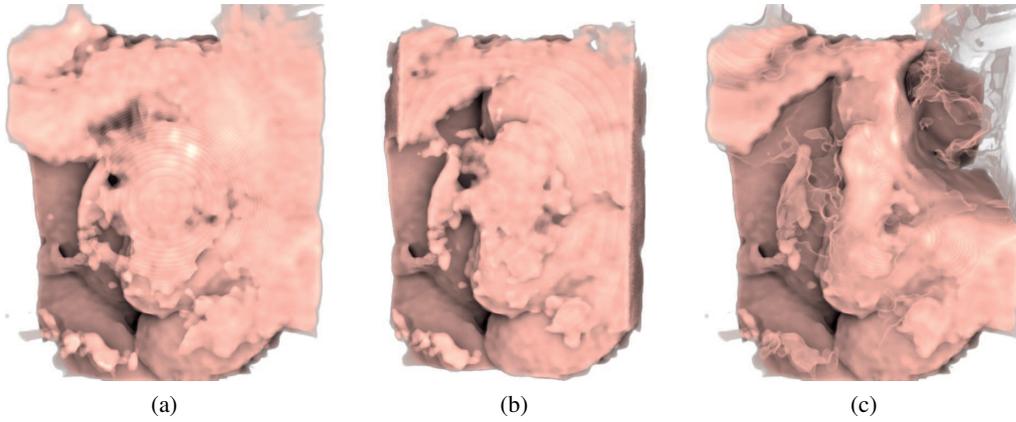


Figure 4.17: Comparison of results for 'Dataset 7' - medium category. (a) original data - DVR with DOS, (b) 'electric scalpel' - DVR with DOS, (c) SVPU algorithm - DVR with DOS.

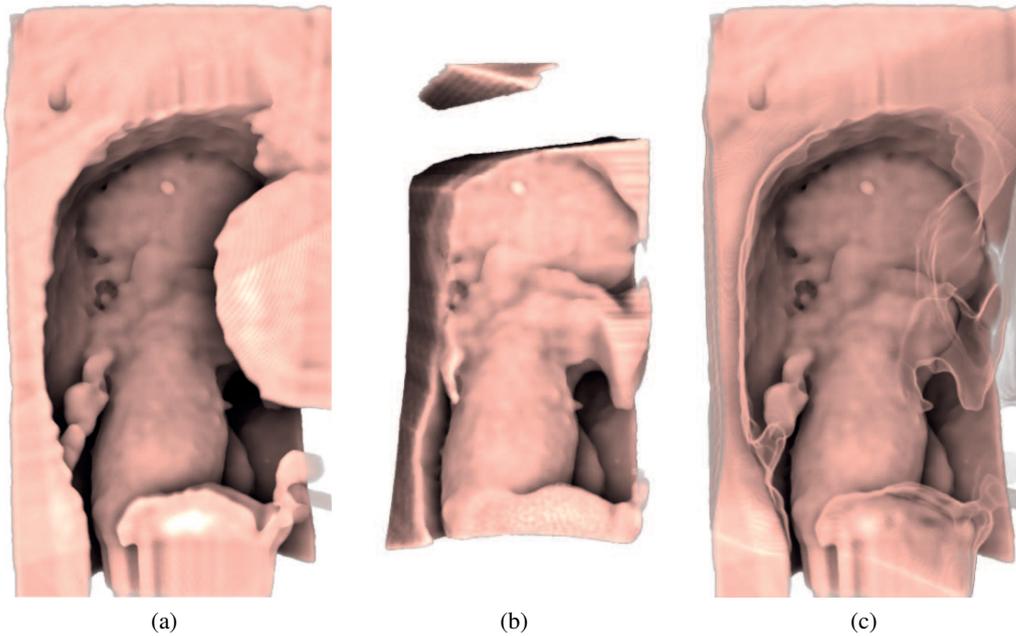


Figure 4.18: Comparison of results for 'Dataset 8' - hard category. (a) original data - DVR with DOS, (b) 'electric scalpel' - DVR with DOS, (c) SVPU algorithm - DVR with DOS.

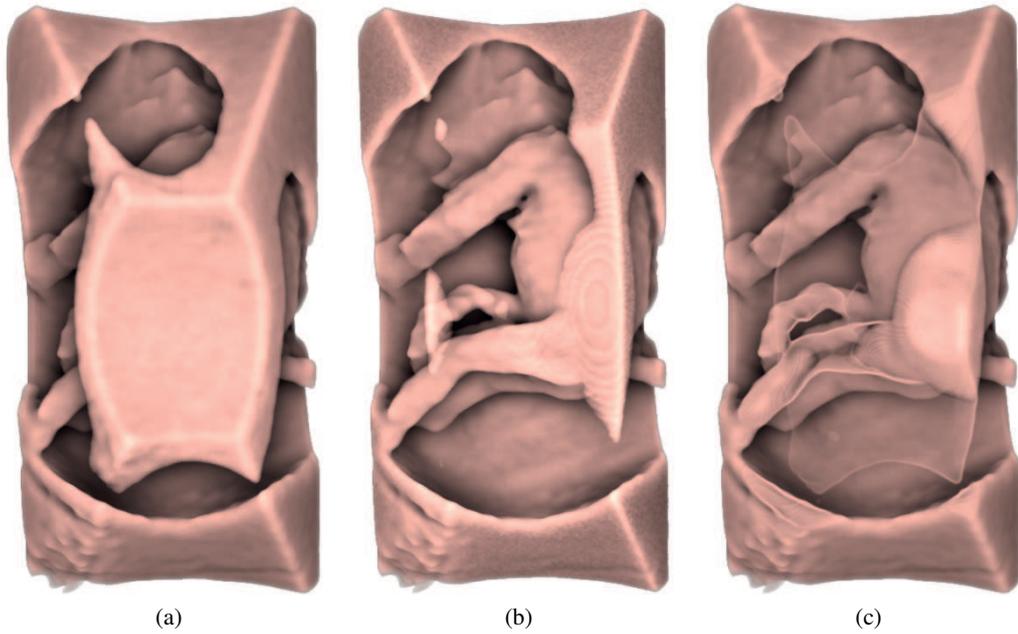


Figure 4.19: Comparison of results for 'Sequence 1'. (a) original data - DVR with DOS, (b) manual ROI box - DVR with DOS, (c) SVPU algorithm - DVR with DOS.

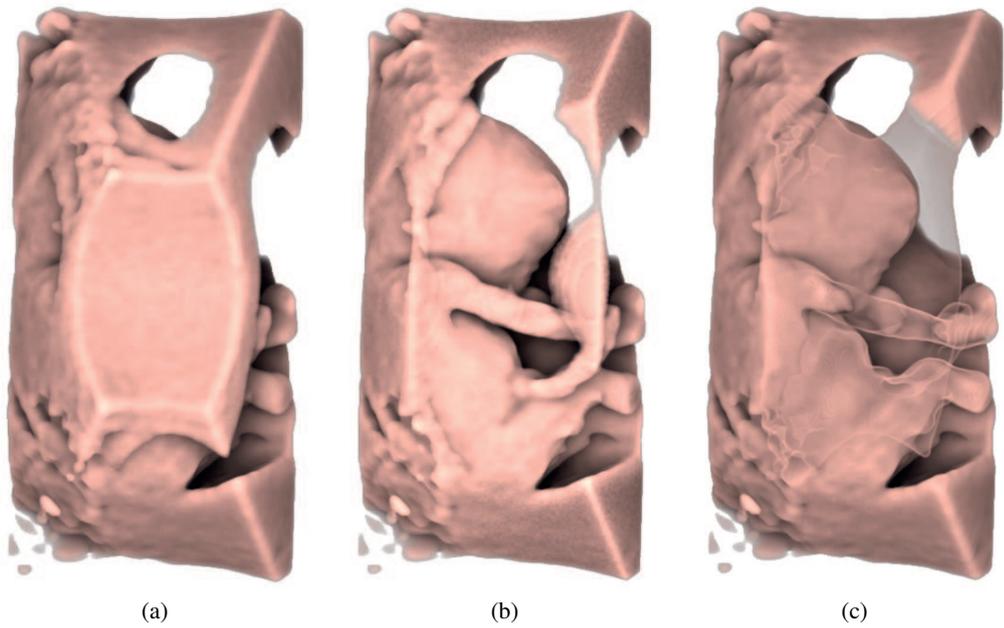


Figure 4.20: Comparison of results for 'Sequence 2'. (a) original data - DVR with DOS, (b) manual ROI box - DVR with DOS, (c) SVPU algorithm - DVR with DOS.

Fetoscopic Rendering

5.1 Introduction

Ultrasound imaging is widely used during pregnancy and it is very well recognized by the public. Viewing ultrasound images during and after an examination is important for parents because they want to understand the results of the examinations. Expressive imaging methods can lead to better discussions after the examination results are available and simplify the communication of findings. Patients usually also want to keep the images as a memory or to present them to their family. Furthermore, clinical confidence has also an important impact on the communication between clinicians and parents. Therefore, the visual quality of images is very important for patients, who undergo an ultrasound scan, and for clinicians, who perform the examination.

There are several different rendering modes in 3D/4D obstetric ultrasound. The most often used visualization algorithm is volume rendering for the display of surfaces. However, images rendered with this mode have a plastic-like and unrealistic look. In this chapter, we present a rendering method for realistic image synthesis from ultrasound data which aims to reproduce the visual quality of images coming from live fetoscopic examinations. Since the visual properties of real fetoscopic images are our goal, we develop an advanced illumination model that supports realistic skin rendering of the human fetus. We implemented our system as the *HDlive* imaging tool within Voluson E8 and Voluson E8 Expert which are the latest generation of GE Healthcare ultrasound imaging systems. As the visual quality of our images is an important aspect, we discussed our results with a group of ultrasound domain experts that helped us evaluate the results that we achieved with our system.

In the following section, we discuss related work to advanced volume rendering. In Section 5.3 we define the goals for live fetoscopic rendering of 3D/4D ultrasound and explain our visualization pipeline. In Section 5.4 we discuss our decisions during the development and cover the mathematical background of our model. Section 5.5 covers implementation details of our method. We discuss the parameter specification process for our model and evaluate results based on the feedback coming from clinical domain experts in Section 5.6. The final section of the chapter includes conclusions and a short discussion of possible directions for future work.



Figure 5.1: Examples of images produced by a real fetoscope. Images are credited to Professor Andrzej Skawina (Collegium Medicum Jagiellonian University, Krakow), Dr. Antoni Marsinek, MD (Czerwiakowski Gynecological and Obstetrics Hospital, Krakow) and to Zrodlo Foundation, Krakow.

5.2 Related Work

Volume rendering researchers have developed a large number of methods that can add various illumination effects to volumetric visualizations. Max [96] discusses various concepts of advanced optical models for volume rendering. In general, there are two main approaches to direct volume rendering, i.e., image-based methods and object-based methods. Illumination effects



(a)



(b)

Figure 5.2: Comparison of conventional and fetoscopic rendering. (a) Image produced with traditional DVR with local illumination. (b) Image rendered with our fetoscopic renderer.

which can be integrated with direct volume rendering strongly depend on the selected volume rendering approach.

Ray casting is an image based method that produces images by processing rays for individual pixels of the image. The rays propagate through the volume that is sampled and the optical properties are mapped to the samples with a transfer function. Typically a local illumination



(a)



(b)

Figure 5.3: Comparison of conventional and fetoscopic rendering. (a) Image produced with traditional DVR with local illumination. (b) Image rendered with our fetoscopic renderer.

model, such as Phong shading, is employed to provide shading for surface-like data regions. The final value of a pixel is computed by compositing the samples along an individual ray.

Levoy [85] proposed the ray casting method for direct volume rendering. Hadwiger et al. [52] give a practical course of possible illumination techniques based on ray-casting. Shadows can significantly increase the realism of an image and add important perceptual cues for the observer. Ropinski et al. [123] presented a survey with possible solutions for extending a ray-casting based system with shadows. There are several works that describe how to include shadow effects into a volume rendering system. Hadwiger et al. [51] proposed deep shadow maps for adding volumetric shadows based on an approximation of the occlusion profile in order to decrease the computational load. Rezk-Salama [119] achieved advanced shadowing and light scattering effects with a Monte-Carlo based ray-casting method. However, the algorithm is not suitable for real-time applications and it is restricted to iso-surface rendering. Ropinski

et al. [122] proposed an illumination model with light volumes, where the illumination cache is computed directly on the GPU. More complex approaches were proposed to simulate global illumination and complex light interactions with different materials using spherical harmonics in order to achieve a more realistic appearance. Ritschel [121] used spherical harmonics and hierarchical visibility computations for volume illumination. Lindemann and Ropinski [88] developed a heuristic approach for simulating advanced light-material interactions with spherical harmonics. Kronander et al. [72] optimized the computation of lighting with spherical harmonics for a better performance. Although the visual quality of images produced with spherical harmonics is very convincing, they still require precomputed lighting information. Ambient-occlusion shading methods were studied for volume visualization to increase the realism while avoiding costly global illumination methods. Ambient-occlusion methods are ray-casting based methods that consider the local neighborhood of a given sample to approximate global illumination. Vicinity shading [140] was proposed to enhance depth perception of volumetric data. Hernel et al. [55] proposed a local ambient occlusion technique that estimates local visibility of a sample in a spherical neighborhood. They use this information for shading of the samples in DVR. Hernel et al. [56] [57] improved this idea with piecewise integration and multiresolution volume rendering. These optimizations allow the computation of the global visibility with interactive framerates. Ropinski et al. [124] developed a method for dynamic ambient occlusion with color bleeding. This approach computes lighting information in the preprocessing step and allows interactive exploration of volumetric datasets.

Ray-casting approaches compute light propagation through the volume along individual rays and the integration for each pixel on the rendered image is determined in parallel (see Section 2.8). Although the ray-casting approach is friendly to the computational model of recent graphical processing units (GPUs), advanced illumination effects are usually difficult to achieve because of synchronization issues. Sundén et al. [142] recently proposed an image-plane sweeping for ray-casting based rendering to integrate advanced illumination effects such as light scattering which allows on-the-fly rendering without precomputation. The algorithm is utilizing the sweeping paradigm which is similar to slice-based rendering.

Slice-based volume rendering methods belong to the category of object-order approaches (see Section 2.8). These methods generate multiple slices of polygons that intersect the volume. The slices are aligned orthogonally to the view direction and are parallel to each other. Samples of the volumetric data are used to texture the slice polygons. This approach performs direct volume rendering efficiently by blending the slices together on the GPU. Local illumination effects can be added as well.

Cabral et al. [21] were the first to propose direct volume rendering with a slice-based approach. Brehens and Ratering [11] added shadows to slice-based volume rendering. Kniss et al. [69] introduced half-angle slicing. This method allows to add hard volumetric shadows to direct volume rendering without precomputation. In half-angle slicing the axis of the slice is oriented half way between the light and the view directions. The half-angle slice is swept through the volume. The integration of the rendered image is synchronized with the shadow map computation which stores the information of light propagating through the volume.

In further work, Kniss et al. [70] [71] extended half-angle slicing with soft shadows and light scattering effects. Light scattering is approximated using a diffusion operation. For every slice,

the algorithm blurs samples of the shadow map with a low-pass filter in order to approximate light scattering.

For the visualization of ultrasound data, Desgranes et al. [31] performed an additional dilation in the shadow map. Dilation is computed by taking the minimum value from samples in the conical neighborhood in each lock-step update of the shadow map. In this way, they achieve additional diffusion-like shading effects.

Schott et al. [128] achieved effects of soft shadows with view-aligned slice-based DVR. Their directional occlusion shading sweeps the shadow map through the volume. In each update to the shadow map, it blurs the samples taken from a conical neighborhood by a low pass filter. A soft shadow effect is achieved based on the convolution with a low pass filter. The light source has a fixed position in the directional occlusion shading model. Soltészová et al. [135] proposed an extension to directional occlusion shading. By modifying the convolution kernel, they can control the light direction. Due to the view-aligned slice orientation this model is restricted only to the light coming from the front hemisphere.

5.3 Goals of Live Fetoscopic Rendering

In this section we discuss the main goals of our system in more detail and describe its architecture. We analyze the current 4D ultrasound modality and the properties of the data acquired during fetal examination. Driven by the requirements of ultrasound domain experts, we discuss the desired visual properties of fetoscopic images from a visualization perspective. We explore several relevant rendering methods and consider their applicability in our scenario. Based on the character of the data and the requirements of the clinicians, we propose a method for live 4D ultrasound rendering of the human fetus.

We developed our system for the live 4D ultrasound visualization of a moving fetus in a clinical environment. We identify the main goals of our system based on the requirements from the clinicians and the constraints of the ultrasound modality.

Live performance: In the typical scenario of a live examination, the clinician guides the ultrasound transducer and tries to find a favorable position for the investigation of the fetus. The system has to provide a real-time visual feedback on the screen of the ultrasound machine during the live scan. A 4D transducer sends the data continuously to the ultrasound machine. Significant delays would hinder clinicians from a live examination.

Fetoscopic visual quality: Images of the fetus, produced by our system, should look realistic in order to increase the confidence of the clinician performing the examination of the fetus. A rendering algorithm should achieve visual characteristics similar to the images produced by a fetoscope (see Figure 5.1). Clinicians are trained based on fetoscopy images and they study the development of the fetus with ultrasound by comparing these images. Additionally, more realistic images can also improve the communication of the findings with the parents (see Figure 5.2 and Figure 5.3).

A fetoscope is a medical device that has a camera and a light source which is used for the illumination of the fetus. Illumination of the fetus with the light source makes it visible to the physician which is performing the procedure. The clinician changes the camera and the light position to study internal and external structures of fetal anatomy from different perspectives.

Changes of the light source position cause the appearance of shadows on the structures of the fetus or surrounding tissues. When the light falls on the fetus, it produces a range of tones on the surface of the skin. These subtle details appear because of the anatomy of the human skin. The skin of the fetus is a translucent material which has distinct optical properties. It absorbs the light and changes the color and the direction of the light underneath the skin surface. These effects are called *spectral* or *chromatic absorption* and *subsurface* or *multiple scattering*.

Interactive parameter control: Ultrasound data differs from standard modalities (CT, MRI,...) and cannot be classified by tissue intensities. Adaptation of the visualization parameters is usually required in order to adjust the displayed image. The fetus is typically occluded by other tissue. The region of interest and a clipping plane have to be specified interactively to remove the occluding parts. The clinician has to be able to interactively control the parameters of the visualization on the ultrasound machine.

Robustness to the characteristics of sonographic data: Data values that correspond to the intensity of the acoustic signal can vary due to reflections between tissues. Ultrasound data is usually also degraded by various types of noise that significantly decrease the signal-to-noise ratio. A typical type of artifacts, known as *speckle* noise, appears due to interference effects of the acoustic signal and different structures inside the human body. In order to improve the perceptual quality of the images, the system should be resistant to the noisy character of the ultrasound modality.

5.4 Fetoscopic Illumination Model

In real fetoscopy, the clinician can change the position of the camera and the light to study the anatomy of the fetus. Our volume rendering algorithm should employ an advanced illumination model that can visualize 4D ultrasound data similar to the navigation with the real fetoscope.

In the related work, we discussed the state-of-the-art of advanced illumination models for direct volume rendering. During the development of our system, we evaluated the advantages and disadvantages of these approaches in the context of the constraints of 4D ultrasound visualization in obstetrics. We compared features from several existing approaches. Table 5.1 shows a summary of the considered illumination models. We compared selected illumination models and their properties based on our goals and the requirements of our workflow. Our comparison included methods from both direct volume rendering approaches. We took the following issues related to our workflow into account:

Advanced effects: Realistic depiction of the fetus requires global illumination effects. We identified advanced effects that are possible to realize with the individual models. For the purpose of our model, we searched for effects including volumetric shadows, light scattering, color bleeding, ambient occlusion, and advanced light-material interactions.

Robust to noise: Ultrasound has a low signal-to-noise ratio that has an impact on the quality of the rendered image. Therefore we considered mainly advanced illumination models that are robust also to noisy data.

Streaming: Ultrasound data is sent continuously by the transducer to the ultrasound machine. The data flows as slabs through our workflow and only part of the whole volume is

accessible to the rendering stage at a time. Knowledge about the data access pattern was therefore important for our workflow. It was important to determine whether the illumination model requires access to the whole volume or can compute the effects in a streaming fashion.

Pre-computation: Instant visual feedback has to be given to the clinician without any substantial delay. The illumination method has to be able to render the illumination effects in real-time without any pre-computation. Some rendering algorithms rely on this step and allow real time visualization only when the pre-processing is finished.

Table 5.1: Comparison of illumination models

Method	Streaming	Robust to Noise	Pre-computation	Advanced Effects	Light Position
Directional occlusion shading [128]	yes	yes	no	shadows	-
Multi-directional occlusion shading [135]	yes	yes	no	shadows	viewer's hemisphere
Deep shadow maps [51]	no	yes	yes	shadows	free
Half-angle slicing [70]	yes	yes	no	shadows, multiple scattering, color bleeding	free
Gradient-free shading [31]	yes	yes	no	shadows, gradient-free specular highlights	free
Shading with Light Volume [122]	no	yes	yes	shadows, multiple-scattering, color bleeding	free
Dynamic ambient occlusion [124]	no	yes	yes	shadows, color bleeding	-
Local ambient occlusion [57]	yes	yes	no	ambient-occlusion	-

Our analysis showed that some of the techniques could be used and customized for our purpose. In our study, we analyzed more closely which of the established scientific methods can be integrated into a robust and reliable workflow on an ultrasound machine. We built the foundation of our model by combining features from several existing approaches.

The streaming character of our workflow was one of the main decisive criteria for the selection of the fetoscopic illumination model. A real-world clinical US workstation requires a stable performance in a clinical environment. Hardware components of the machine are constrained by several factors and they are carefully chosen for long-term functioning. The streaming architecture was designed in order to scale up with the next generations of ultrasound transducers and processing components of ultrasound machines. Several methods for volume illumination, that we compared, did not fit to this workflow because they assume global access to the volume [51] [122] or they require pre-computation [51] [124]. We noticed that these methods belong to ray-casting approaches. It is possible to apply local ambient occlusion [57], because it requires only a small neighborhood for the computation. Slice-based methods [70] [31] [128] [135] appeared to be applicable in our application scenario because of their low memory footprint

and the way how they compute the illumination. Slice-based methods that we compared use a specific computation order by sweeping a slice through the volume.

We considered also robustness to noise of the summarized illumination methods. In some cases, a potential applicability to noisy data is indicated on data examples coming from modalities like MRI [57] [122] or ultrasound [31] [135]. In our further steps we assessed the individual illumination effects based on the goals of live fetoscopic rendering.

Camera position: The camera is attached to the end of the fetoscope which allows to display images on a monitor for navigation. Our workflow has to visualize an ultrasound signal acquired on-the-fly by a 4D transducer on the monitor of the ultrasound machine. A fetoscopic illumination model has to support dynamically changing volume data. All of the methods that we compared can interactively change the camera position.

Light position: The fetoscope uses a light source for the illumination of the inner body structures. The shadows appear on the inner structures that are obscured from the perspective of the light source. We have to be able to imitate the fetoscope's light movement through the interface of the ultrasound machine. Our scenario required the interactive control of the light position with unrestricted positioning of the light around the scene. Several models [70] [31] [122] [51] fulfill this requirement. Directional occlusion models that we compared did not support the movement of the light source throughout the scene. They either assume a static light source position [128] or they have a limited degree of freedom [135]. Ambient occlusion methods [124] [57] compute an illumination effect which does not simulate a movable light source at all.

Light intensity: Fetoscopes typically contain optical fibers that transmit the light to the area of examination. Enough light has to be transmitted to the fetus for a clear view. The light intensity of the fetoscope can also be controlled by the clinician to get a brighter image, even though it is restricted because the increased heat of the light could burn the patient. It has to be possible to control the light intensity with our illumination model. It is possible to adjust the intensity of the shading effect with all of the compared models.

Shadow softness: A partially obscured light source of the fetoscope causes the appearance of soft shadows. The darkness of the shadow is varying on the illuminated surface depending on the size of the light emitting area that is obscured from the perspective of the illuminated surface. This creates the effect of soft shadows. Larger fetoscopes contain more fibers and can produce a brighter image. A real fetoscope is restricted to a spot light source because it is desirable to reduce the degree of invasiveness. Our illumination model has to have a possibility to control the appearance of shadows. Slice-based methods [70] [31] [128] [135] are more suitable for achieving soft shadow effects because of their inherent synchronization of the light front during light integration. It is possible to achieve soft shadows also in case of ray-casting methods when the shadow volume is pre-computed [122]. Ambient occlusion methods [124] [57] are also applicable for soft shadow effects and were considered in our scenario.

Skin rendering: The skin of the fetus is going through many changes during gestational development (see Section 2.9). In the early pregnancy the skin is more transparent than in later stages of pregnancy. When the fat starts to develop the fetus is beginning to have a look of a newborn child. It is necessary for our model to display various skin tones of the human fetus. The skin of the fetus is a translucent material and the light that interacts with it produces a range of color tones. Scattering of the light underneath the skin surface is responsible for the coloring

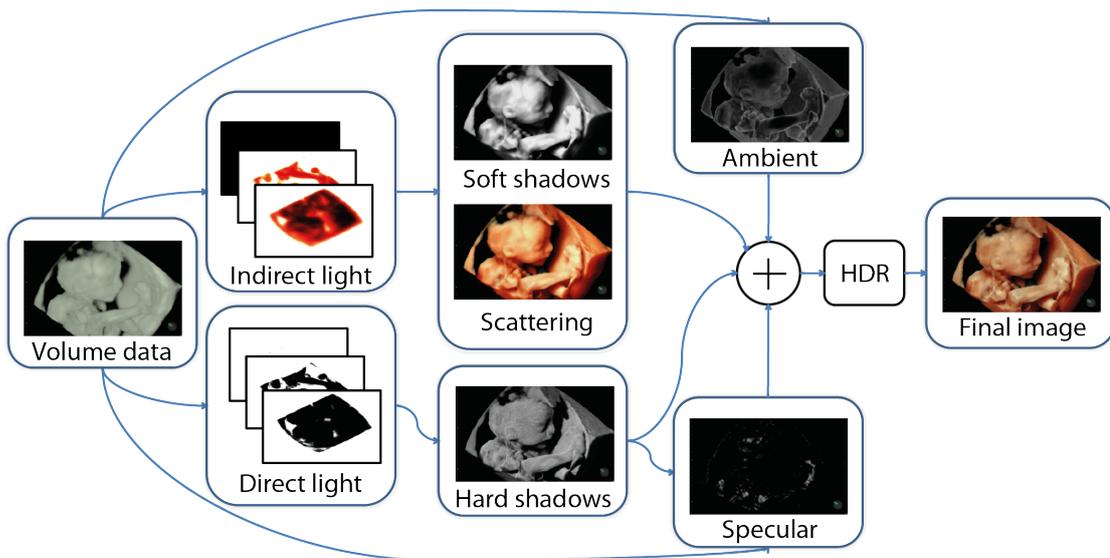


Figure 5.4: Individual components of illumination model for fetoscopic rendering. Direct light - hard shadows, indirect (achromatic) light - soft shadows, specular highlights, indirect (chromatic) light - multiple scattering, local ambient occlusion, and final image after HDR post processing.

of the skin. Our model has to realistically reproduce this color bleeding effect. Some of the illumination methods appeared to be suitable [70] [122] [124] to approximate this advanced effect for translucent materials.

We decided to build our illumination model based on the half-angle slicing approach [70], which fulfilled most of the required criteria for fetoscopic rendering. The main reason for choosing this approach is the list of advanced effects which could be achieved with a very low memory footprint. The possibility to integrate the algorithm into our streaming architecture and an unrestricted light source position were also considered as advantages of this algorithm. However, the selection of this algorithm also had implications on how we process the ultrasound data that is streamed through the pipeline.

We can achieve several advanced illumination effects with direct volume illumination based on half-angle slicing. These effects include hard shadows, soft shadows, specular highlights, local ambient occlusion, and light scattering effects. Figure 5.4 illustrates all individual components separately. We explain how we adapt the half-angle slicing method and how we extend it with other features for the purposes of live fetoscopic rendering.

Mathematical Background

We have introduced the basic emission-absorption optical model in Equation (2.4) which is described by the differential equation for transport of light. The fetoscopic illumination model belongs to the category of global illumination models. Global illumination models are described

by the differential equation for the transport of light:

$$\vec{\nabla}_s L(s, \vec{\omega}_V) = Q(s, \vec{\omega}_V)\tau(s) - L(s, \vec{\omega}_V)\tau(s), \quad (5.1)$$

where the term $\tau(s)$ corresponds to the extinction term. It corresponds to the attenuation of the light due to absorption and *scattering*. It is defined as:

$$\tau(s) = \sigma_A(s) + \sigma_S(s), \quad (5.2)$$

where $\sigma_A(s)$ is the absorption coefficient and the term $\sigma_S(s)$ is the scattering coefficient. The source term $Q(s, \vec{\omega}_V)$ corresponds to the light intensity that is added at the position s due to the self-emission of particles and in-scattering. The complete optical model for the transport of light with emission, absorption and scattering is defined as [23]:

$$\begin{aligned} \vec{\nabla}_s L(s, \vec{\omega}_V) = & \underbrace{\sigma_A(s)L_E(s, \vec{\omega}_V)}_{\text{emission}} + \underbrace{\sigma_S(s) \int_{\Omega} p(\vec{\omega}_V, \vec{\omega}_I)L(s, \vec{\omega}_I)d\vec{\omega}_I}_{\text{in-scattering}} \\ & - \underbrace{\sigma_A(s)L(s, \vec{\omega}_V)}_{\text{absorption}} - \underbrace{\sigma_S(s)L(s, \vec{\omega}_V)}_{\text{out-scattering}} \end{aligned} \quad (5.3)$$

The in-scattered light is coming to the position s from any direction $\vec{\omega}_I$ defined around the sphere Ω . The term $p(\vec{\omega}_V, \vec{\omega}_I)$ is the *phase function*. The light is scattered when it hits a particle with an index of refraction different from its environment. A phase function describes the probability that the light coming from a direction $\vec{\omega}_I$ is scattered into another direction $\vec{\omega}_V$. It is often dependent on the the angle between incoming and outgoing directions of the light, i.e., $\vec{\omega}_I$ and $\vec{\omega}_V$. The light is also attenuated as it travels along direction $\vec{\omega}_V$ due to the out-scattering. The solution of the differential equation along the view direction $\vec{\omega}_V$, between the initial position and the eye position V , is the volume rendering integral [23]:

$$L(V, \vec{\omega}_V) = L(0, \vec{\omega}_V)e^{-\int_0^V \tau(t)dt} + \int_0^V Q(s, \vec{\omega}_V)\tau(s)e^{-\int_s^V \tau(t)dt} ds. \quad (5.4)$$

The term $L(0, \vec{\omega}_V)$ corresponds to the light coming from the background. The source term $Q(s, \vec{\omega}_V)$ is defined as as [23]:

$$Q(s, \vec{\omega}_V) = (1 - \gamma(s))L_E(s, \vec{\omega}_V) + \gamma(s) \int_{\Omega} p(\vec{\omega}_V, \vec{\omega}_I)L(s, \vec{\omega}_I)d\vec{\omega}_I, \quad (5.5)$$

where $\gamma(s) = \frac{\sigma_S(s)}{\tau(s)}$ is the *albedo*. It is a dimensionless parameter defined as the ratio between the scattering coefficient and the extinction term.

The optical depth $T(s_a, s_b)$, also called *transparency*, of the interval is [53]:

$$T(s_a, s_b) = e^{-\int_{s_a}^{s_b} \tau(t)dt}. \quad (5.6)$$

It corresponds to the probability that the light ray travels a distance between the position s_a and the position s_b without being absorbed or scattered. For the numerical computation of the volume rendering integral by a Riemann sum (see Section 2.8), the transparency of the interval between the samples s_i $s_i + \Delta s$ is approximated as:

$$T(s_i, s_i + \Delta s) = e^{-\int_{s_i}^{s_i + \Delta s} \tau(t) dt} \approx e^{-\tau(s_i)\Delta s}. \quad (5.7)$$

Light propagation in a tissue that is assumed to be a random medium, such as human skin, is simulated with *albedo*, *transparency* (optical depth) and *phase function* [10]. Human skin is a tissue with a high albedo. If the light is coming from a certain direction $\vec{\omega}_I$, it is scattered in the tissue predominantly in a forward direction $\vec{\omega}_O$ within a certain range of diffusion. We can describe this mathematically with the concept of a phase function. Phase functions can be also expressed as functions of an angle θ between incoming and outgoing light $\cos \theta = \vec{\omega}_O \cdot \vec{\omega}_I$. We also assume that the phase function is normalized such that the integral over 4π steradians is unity:

$$\int_{\Omega} p(\vec{\omega}_O \cdot \vec{\omega}_I) = 1 \quad (5.8)$$

Since we want to approximate light propagation in the human skin tissue, our phase function needs to be predominantly *forward scattering*. The mean cosine μ of the scattering angle between $\vec{\omega}_I$ and $\vec{\omega}_O$ is:

$$\mu = \int_{\Omega} (\vec{\omega}_O \cdot \vec{\omega}_I) p(\vec{\omega}_O \cdot \vec{\omega}_I) d\vec{\omega}_I \quad (5.9)$$

If the mean cosine μ of the scattering angle is positive $\mu > 0$, the phase function is predominantly forward scattering.

The Henyey-Greenstein (H-G) phase function [54] well approximates the light propagation in human skin [157]. The H-G function is an empirical phase function that by variation of one parameter $-1 \leq g \leq 1$, called anisotropy coefficient, produces back scattering, isotropic scattering, and forward scattering.

$$p(\theta) = \frac{1}{4\pi} \frac{1 - g^2}{(1 + g^2 - 2g \cos(\theta))^{3/2}}. \quad (5.10)$$

For $g > 0$, forward scattering is dominant. For human skin tissues, the anisotropy parameter of the H-G function was estimated to be $g \in [0.85, 0.91]$ [157]. The range of this phase function has predominantly a conical shape (see Figure 5.5) with the cone apex at the scattering position s of light coming from the direction $\vec{\omega}_I$. In fetoscopic rendering, we compute the light propagation with an algorithm which also assumes a forward scattering phase function in the human skin. We restrict the evaluation of the scattered light propagation from all directions around the unit sphere only to the conical range.

Unlike the volume rendering integral for the basic optical model which yields a pure integral (see Equation 2.7), the model with light scattering yields an integro-differential equation [53]. We adapt the light transport equation to the fetoscopic rendering with several assumptions. We omit the emission term $L_E(s, \vec{\omega}_V) = 0$ in Equation 5.5 since our model is designed for rendering

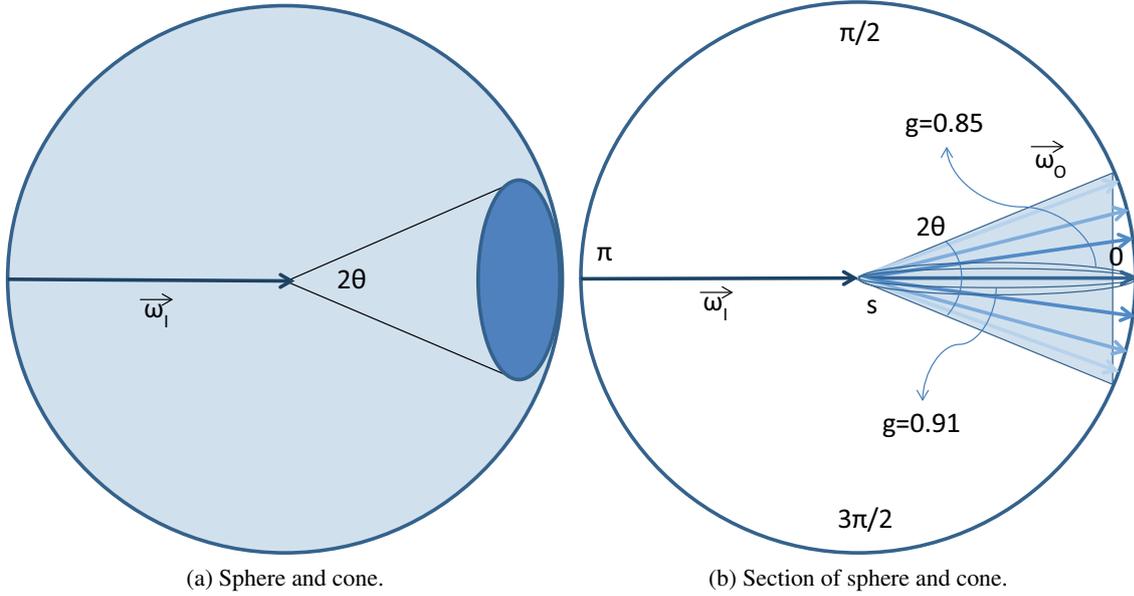


Figure 5.5: (a) A range of light scattering can be restricted within the cone with an apex angle 2θ . (b) Polar plot of the normalized Heyney-Greenstein phase function which can approximate the forward scattering in the human skin with $g \in [0.85, 0.91]$. The function returns high values only for very skew angles of scattered light directions.

of human tissues which do not emit light in a visible range. We also assume that the virtual light source is the only source of light in the scene and we set the light coming from the background in Equation 5.4 to zero $L(0, \vec{\omega}_V) = 0$. We define the volume rendering integral for the fetoscopic rendering as:

$$L(V, \vec{\omega}_V) = \int_0^V \underbrace{\overbrace{(\gamma(s))}_{\text{albedo}} \int_{\Omega} \overbrace{p(\vec{\omega}_V, \vec{\omega}_I)}^{\text{phase function}} L(s, \vec{\omega}_I) d\vec{\omega}_I}_{\text{source term}} \tau(s) \overbrace{T(s, V)}^{\text{transparency}} ds. \quad (5.11)$$

If we represent the source term of our model as:

$$Q_F(s, \vec{\omega}_V) = \gamma(s) \int_{\Omega} p(\vec{\omega}_V, \vec{\omega}_I) L(s, \vec{\omega}_I) d\vec{\omega}_I, \quad (5.12)$$

we can rewrite Equation 5.11 more compactly as:

$$L(V, \vec{\omega}_V) = \int_0^V Q_F(s, \vec{\omega}_V) \tau(s) T(s, V) ds. \quad (5.13)$$

In accordance to the approximation of the volume rendering integral described in Section 2.8, Equation 5.13 can be approximated with a Riemann sum as:

$$L(V, \vec{\omega}_V) \approx \sum_{i=0}^{V/\Delta s} Q_F(s_i, \vec{\omega}_V) \tau(s_i) \Delta s \prod_{j=i+1}^{V/\Delta s} e^{-\tau(s_j) \Delta s}. \quad (5.14)$$

We extend the computation of the volume rendering integral from an achromatic light $L(V, \vec{\omega}_V)$ to a chromatic light $L(V, \vec{\omega}_V, \lambda)$. After substitution of the source term Q_F with the chromatic global illumination term L_F and the opacity term $\alpha(s_i)$ we obtain [127]:

$$L(V, \vec{\omega}_V, \lambda) \approx \sum_{i=0}^N L_F(s_i, \vec{\omega}_V, \vec{\omega}_L, \lambda) \alpha(s_i) \prod_{j=i+1}^N (1 - \alpha(s_j)). \quad (5.15)$$

$\alpha(s_i)$ corresponds to the opacity of the sample i , assigned by a transfer function. $\lambda = \{R, G, B\}$ corresponds to the evaluation of the final light contribution for a color image, i.e., R red, G green, and B blue channel. The transfer function for opacities is a simple windowing function. The global illumination term in fetoscopic rendering is given by:

$$L_F(s_i, \vec{\omega}_V, \vec{\omega}_L, \lambda) = C(s_i) L_G(s_i, \vec{\omega}_V, \vec{\omega}_L, \lambda), \quad (5.16)$$

where $C(s_i)$ is the reflective color opacity of the sample s_i , assigned by a transfer function. $L_G(s_i, \vec{\omega}_V, \vec{\omega}_L, \lambda)$ is a global light contribution scattered in direction $\vec{\omega}_V$ after interactions with participating media. In fetoscopic rendering the scene is illuminated with one *virtual light source* which is defined only with the orientation $\vec{\omega}_L$. $N = V/\Delta s$ discrete samples are taken with constant intervals. Equation 5.15 can be iteratively computed for every pixel of the rendered image with the over compositing operator given in Equations 2.13 2.14 or the under compositing operator given in Equations 2.15 2.16.

In contrast to the local illumination model, given in Equation 2.20, we need to consider the in-scattered contribution of the light to each sample from the whole scene after many previous interactions with the participating media. The amount of light that arrives at the location s_i and is reflected to the view direction is evaluated based on the absorption and multiple scattering in the tissue. The absorption of the light in the medium produces shadows. Shadows correspond to the single scattering of the light. Because of multiple scattering in the skin tissue, the mathematical definitions for the global illumination model are substantially more complex than in the local illumination model. Multiple scattering effects are usually computed with Monte Carlo methods for computing the light transport equation, such as photon-tracing. However, these methods are computationally expensive and they are not suitable for real-time visualization. If we could evaluate the contribution of the scattered light in every iteration, synchronized with the standard DVR compositing of the rendered image as with the local illumination, it would be possible to approximate advanced light-skin interaction effects in real-time. Therefore we approximate scattering effects based on the half-angle slicing.

Half-angle slicing computes the light propagation in a lock step with the DVR compositing of the rendered image. We extend this algorithm to compute several additional illumination effects and combine them into a practical model that can be used for real-time visualization in

fetoscopic rendering. A similar approach is also used with other models that compute global illumination effects [64] [71] [142]. In our fetoscopic illumination model we combine single scattering effects, multiple scattering effects, and ambient light effects. Global light contribution to every composited sample s_i in fetoscopic rendering is given by:

$$L_G(s_i, \vec{\omega}_V, \vec{\omega}_L, \lambda) = \underbrace{\overbrace{\gamma(s_i)}^{\text{albedo}} I(s_i, \vec{\omega}_V, \vec{\omega}_L, \lambda)}_{\text{chromatic indirect light}} + \underbrace{(1 - \gamma(s_i)) U(s_i, \vec{\omega}_V, \vec{\omega}_L)}_{\text{direct light}} + \underbrace{S(s_i, \vec{\omega}_V, \vec{\omega}_L)}_{\text{specular highlights}} + \underbrace{A(s_i, \vec{\omega}_V)}_{\text{ambient occlusion}} \quad (5.17)$$

The term $\gamma(s_i)$ corresponds to the *albedo*. The term $I(s_i, \vec{\omega}_V, \vec{\omega}_L, \lambda)$ corresponds to the *indirect light* in-scattered after multiple scattering events. The light contribution in-scattered after a single scattering event is represented by the *direct light* term $U(s_i, \vec{\omega}_V, \vec{\omega}_L)$. *Specular* highlights correspond to the single scattering event from the skin surface and they are represented with the term $S(s_i, \vec{\omega}_V, \vec{\omega}_L)$. And finally, ambient light in-scattering is in our model considered by the *ambient occlusion* term $A(s_i, \vec{\omega}_V)$. Figure 5.4 illustrates all individual components of the fetoscopic illumination model. We perform several simplifications in order to compute all individual components.

Indirect Light: Multiple Scattering and Soft Shadows

We use only one virtual light source in the fetoscopic illumination model. We can simplify the computation of the multiple-scattering term (see Equation 5.12) based on this assumption:

$$Q_F(s, \vec{\omega}_I) \approx \gamma(s) \int_{\Omega: \vec{\omega}_I \cdot \vec{\omega}_L > 0} p(\vec{\omega}_V, \vec{\omega}_I) \underbrace{L(s, \vec{\omega}_I)}_{\text{reduced intensity}} d\vec{\omega}_I, \quad (5.18)$$

where $\Omega : \vec{\omega}_I \cdot \vec{\omega}_L > 0$ corresponds to contributions of the scattered light that are coming only from the hemisphere of the light source. The term $L(s, \vec{\omega}_I)$ is the intensity of the scattered light. The light intensity is decreased exponentially with the distance from the previous scattering position according to the transparency of the participating media (see Equation 5.6). Reduced light intensity can be written as:

$$\underbrace{L(s, \vec{\omega}_I)}_{\text{reduced intensity}} = \underbrace{L(s - d(s, x)\vec{\omega}_I, \vec{\omega}_I)}_{\text{reduced intensity}} \underbrace{T(s, x)}_{\text{intensity at previous position}}, \quad (5.19)$$

where $L(s - d(s, x)\vec{\omega}_I, \vec{\omega}_I)$ is light intensity at the previous position x of the light. The distance between the previous position x of light and the current position s corresponds to the term $d(s, x)$.

We compute the propagation of the light through the participating media in a lock step i with the compositing of the rendered image based on the half-angle slicing algorithm. Half-angle slicing uses a *shadow map* for the computation of the scattered light contribution. The shadow

map propagates through the volume in the direction of the light orientation $\vec{\omega}_L$ in a lock step Δs with compositing. A position on the shadow map corresponds to x . The exponential extinction of the light can be approximated with opacity at the sample s_i if the distance $d(s_i, x)$ is small (see Equation 5.7 and Equation 2.11). The reduced light intensity, scattered from direction $\vec{\omega}_I$ in each step i , can be computed as:

$$L(s_i, \vec{\omega}_I) \approx \overbrace{L_i(s_i, \vec{\omega}_V, \vec{\omega}_L)}^{\text{intensity at current position}} = \underbrace{L_{i-1}(x\vec{\omega}_I, \vec{\omega}_V, \vec{\omega}_L)}_{\text{reduced intensity}} (1 - \alpha(s_i)). \quad (5.20)$$

Human skin is a highly scattering tissue where light scatters predominantly in forward direction. Forward scattering of the light in direction $\vec{\omega}_I$, which travels predominantly from the direction $\vec{\omega}_L$, can be restricted only to directions within a certain angle $\vec{\omega}_I \cdot \vec{\omega}_L < \theta$. Furthermore, the light propagation tends to become isotropic after multiple scatterings. Each scattering event blurs the light distribution and the distribution becomes uniform. It means that the contribution of the multiple-scattered light in a highly scattering media, such as human skin, can be estimated by an *average light intensity*. This is also a basic assumption for an approximate solution to the volume rendering equation with a multiple scattering term that is based on a diffusion theory [147] [137] [64]. The multiple-scattering term in Equation 5.18 can be approximated as:

$$Q_F(s, \vec{\omega}_I) \approx \gamma(s) \cdot \underbrace{I(s, \vec{\omega}_V, \vec{\omega}_L)}_{\text{average achromatic indirect light intensity}}, \quad (5.21)$$

where $I(s, \vec{\omega}_V, \vec{\omega}_L)$ corresponds to the average light intensity of the multiple-scattered light and we call it *achromatic indirect light*.

Propagation of the achromatic indirect light is computed in every step i with the local filtering operation (see Equation 2.2 in Section 2.6). This is similar to the approximation of the multiple scattering with a convolution using a low-pass filter kernel [71], such as filtering with a Gaussian filter (see Equation 2.3). In fetoscopic rendering we use a weighted average filtering of the shadow map for the approximation of an average light intensity. The computation of the achromatic indirect light is defined as:

$$\underbrace{I_i(x + \Delta s\vec{\omega}_L, \vec{\omega}_V, \vec{\omega}_L)}_{\text{new shadow map}} = \underbrace{(I_{i-1}(x, \vec{\omega}_V, \vec{\omega}_L) \otimes h_I(x))}_{\text{average achromatic indirect light intensity}} \cdot \underbrace{(1 - \alpha(x + \Delta s\vec{\omega}_L))}_{\text{achromatic attenuation}}, \quad (5.22)$$

where $h_I(x)$ is a normalized filter kernel. Weights of the normalized averaging filter kernel $h_I(x)$ correspond to reduced intensities of the light scattered from positions with larger distances. The filtering of the indirect light essentially approximates the light diffusion process, which is typical for the light interaction with human skin [97] [64]. The filtering in Equation 5.22 is performed as a 2D convolution on the shadow map as:

$$\begin{aligned} \underbrace{W(I_{i-1}(x, \vec{\omega}_V, \vec{\omega}_L), \sigma_I)}_{\text{average achromatic indirect light intensity}} &= \underbrace{I_{i-1}(x(u, v), \vec{\omega}_V, \vec{\omega}_L) \otimes h_I(x(u, v))}_{\text{2D convolution}} = \\ &= \sum_{k=-K}^K \sum_{l=-K}^K I_{i-1}(u_k, v_l) \cdot h_I(u - u_k, v - v_l), \end{aligned} \quad (5.23)$$

where $-K < k, l < K$. The size of the filter kernel K , i.e. the number of samples, corresponds to the scale of the filter σ_I . The scale of the filter σ_I is computed from the angle at the apex of the cone-shaped phase function:

$$\sigma_I = \tan(\theta)\Delta s, \quad (5.24)$$

where Δs is the distance between two sampling points s_i and s_{i-1} , corresponding to the sampling distance of the integration. $\vec{\omega}_I \cdot \vec{\omega}_L < \theta$ is the range of the scattered light which is restricted to angles within the cone defined by the angle θ . This defines an implicit forward scattering phase function which describes the light propagation in the fetoscopic illumination model. Hence, forward scattering is approximated with iterative filtering of indirect light.

Besides multiple light scattering of the light under the transparent surface of the skin, the light is also partially absorbed by the human skin in a chromatic way (see Figure 5.6). This effects are responsible for the pink coloration of the skin and the shadow color bleeding effect. The red light is penetrating deeper than blue and green light. This means that the light changes the color with the attenuation. This phenomena produces a perceived color of the human skin which is illuminated with the light, and it also gives rise to a color bleeding effect (see Figure 5.7). A simple opacity-based transfer function is used for classification (see Figure 5.9). Colors are assigned to samples from original intensities of data values that are multiplied by a constant color. The original reflective color of the skin is greenish-grey (see Figure 5.8). We add multiple scattering effect with chromatic absorption to the fetoscopic illumination model by including color channels to the shadow map of the indirect light. Chromatic indirect light can be computed as:

$$I_i(x + \Delta s \vec{\omega}_L, \vec{\omega}_V, \vec{\omega}_L, \underbrace{\lambda}_{\{R,G,B\}}) = \underbrace{\hspace{10em}}_{\text{new chromatic shadow map}} \underbrace{W(I_{i-1}(x, \vec{\omega}_V, \vec{\omega}_L, \underbrace{\lambda}_{\{R,G,B\}}), \sigma_I)}_{\text{average chromatic indirect light intensity}} \underbrace{(1 - \underbrace{\alpha_\lambda}_{\{R,G,B\}} \cdot \alpha(x + \Delta s \vec{\omega}_L))}_{\text{chromatic attenuation}}. \quad (5.25)$$

$\underbrace{\hspace{10em}}_{\text{previous chromatic shadow map}}$

We extend the standard transfer function, which maps colors and opacity to each sample, with a *transport color* [70] (see Figure 5.9). Chromatic light attenuation is simulated with the *indirect alpha* α_λ parameter which is defined by the transport color:

$$T_\lambda = 1 - \alpha_\lambda. \quad (5.26)$$

Indirect alpha corresponds to the evaluation of the light attenuation for each color channel $\lambda = \{R, G, B\}$ separately, i.e., R red, G green, and B blue channel. The transport color T_λ controls the chromatic attenuation of the light when it propagates through translucent material. This effect is responsible for the coloring of the skin within our model. The original color of the skin, which is assigned by the transfer function, is changed under illumination because of the light scattering below the surface. We illustrate the process of the color bleeding effect with our model in a simple experiment (see Figure 5.9). The profile of individual color intensities shows

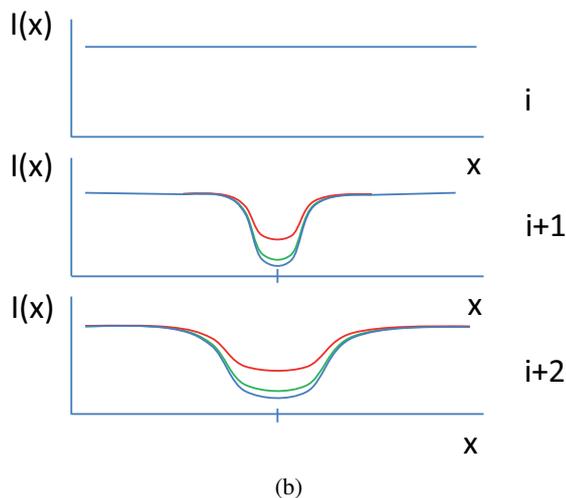
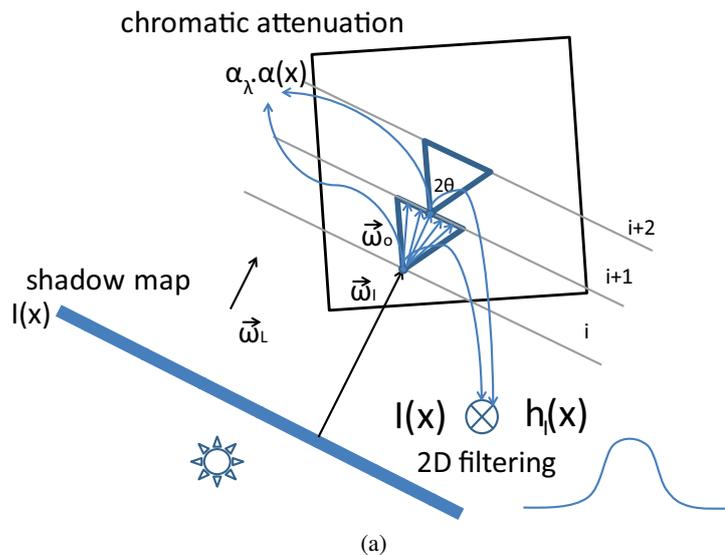


Figure 5.6: Multiple scattering of light is approximated with a weighted average filtering. The amount of the light scattering corresponds to the size of the filter kernel. (a) Iterative front-to-back evaluation of the indirect light propagation corresponds to forward scattering of the light. (b) Indirect light has a chromatic attenuation (see Equation 5.28).

the color bleeding effect. The blue light and the green light are attenuated faster than the red light which propagates deeper into the translucent material.

Additionally to multiple scattering, the filtering of indirect light allows us also to render soft shadows. In global illumination, soft shadows are caused by external area light sources. A partially occluded light source creates a smooth transition of the shadow border on the surface. In our model, we do not evaluate the contribution of the light hitting the surface by tracing of the rays from samples towards the light source. Instead, we illuminate samples by probing the

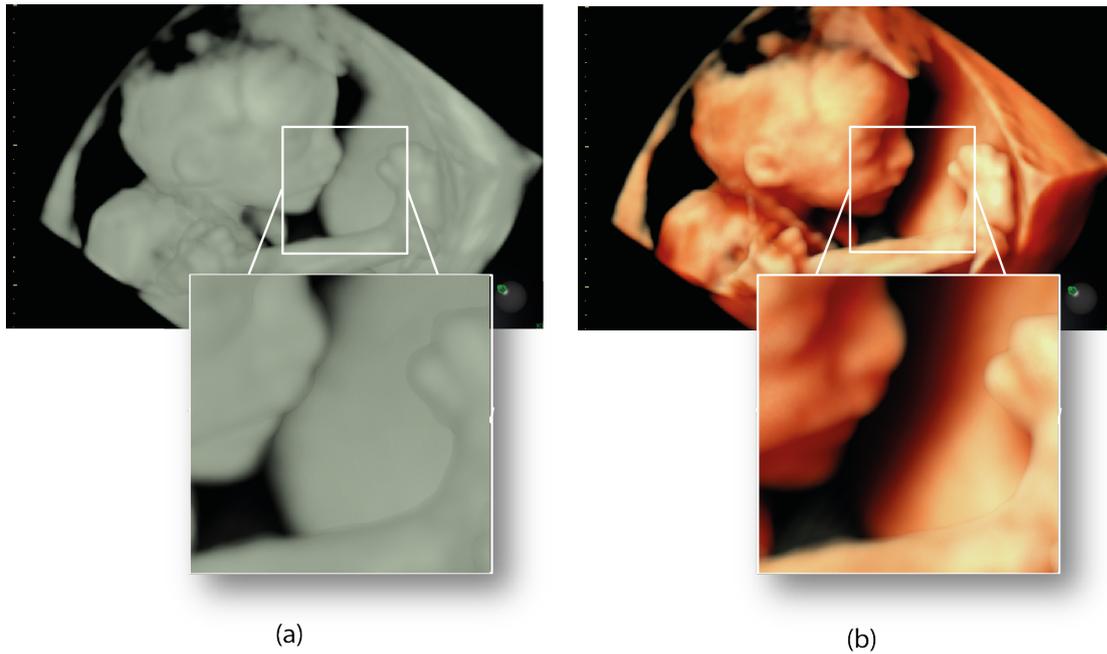


Figure 5.7: Comparison of DVR with basic optical model and DVR with approximation of multiple light scattering. (a) basic DVR without illumination (see Equation 2.12), (b) illumination with chromatic indirect light (see Equation 5.28).

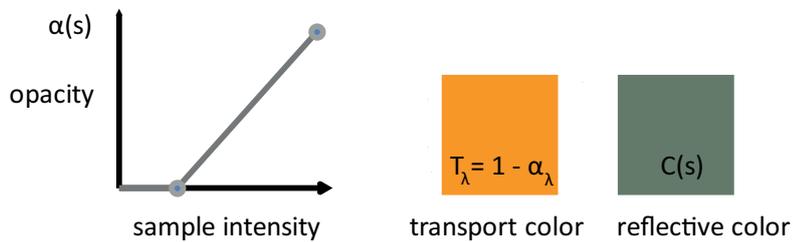


Figure 5.8: Transfer function which defines optical properties for volume samples. Opacity based transfer function is a simple windowing function. Skin reflective color and transport color are constant.

shadow map that is traversed through the volume. The shadow map stores the attenuation of the light intensity according to the opacity assigned by the transfer function. In every iteration step of the compositing we blur the shadow map with the weighted average filter. This blurring is a diffusion process that spreads in all directions across the shadow map. The size of the low-pass filter controls the amount of blurring. We call this effect *shadow softness*. This approach provides a good control over the effect of soft shadows. Variations of this idea were proposed in several other works that were discussed in the section with related work [70] [128] [135]. The

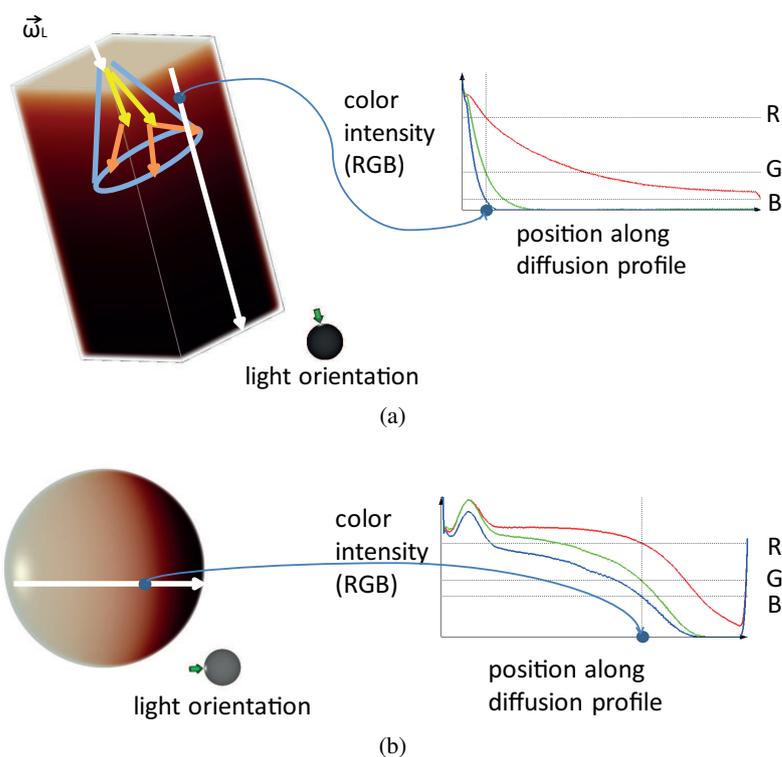


Figure 5.9: Approximation of the chromatic light attenuation in the skin tissue with our illumination model. Skin tissue is colored through multiple scattering with chromatic light attenuation. Diffusion profiles show chromatic attenuation of light for individual color channels.

softness of the shadow can be controlled with the filter kernel size. Filtering with larger kernels increases the softness of the shadows (see Figure 5.10). In addition, as we made the absorption of the light chromatic, we achieve also realistic coloring of the skin and color bleeding effect which appears on the edges of shadows. This appearance is typical for shadows on the human skin (see Figure 5.11).

We noticed that a headlight illumination with the chromatic indirect light, i.e. colored shadows, requires more perceptual cues for the depth perception (see Figure 5.11(a)). This is mainly because humans are more sensitive to perceive small changes in the luminance but not in the hue [89]. Desgranes et al. [31] use dilation on the shadow map of the achromatic direct light for a better shadowing effect. This dilation helps to improve the perception of spatial structures when the headlight is used for illumination in their case. To achieve a more realistic skin appearance, through the color bleeding effect, we apply a similar idea to the computation of our chromatic indirect light. For this purpose, we add an additional alpha channel $I(x, \vec{\omega}_V, \vec{\omega}_L, \alpha)$ to the shadow map of indirect light that stores the chromatic component of attenuation. We use this channel to filter the chromatic component of attenuation that is used for the light color computation. The filtering of the chromatic component of attenuation is performed by weighted

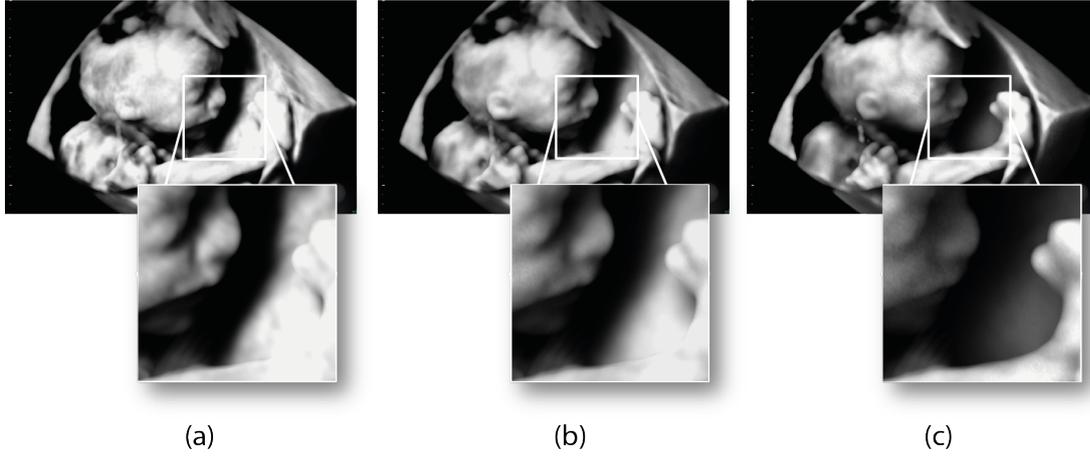


Figure 5.10: Shadow softness corresponds to the filter kernel size. The filter kernel size is computed from the apex angel of the phase function 2θ . (a) $\theta = 20^\circ$, (b) $\theta = 45^\circ$, (c) $\theta = 70^\circ$.

average filtering according to the equation:

$$\begin{aligned}
 & \underbrace{I_i(x + \Delta s \vec{\omega}_L, \vec{\omega}_V, \vec{\omega}_L, \alpha)}_{\text{new opacity of shadow map}} = \alpha(x + \Delta s \vec{\omega}_L) + \\
 & \quad \underbrace{\text{filtered shadow map opacity}} \\
 & + \underbrace{W(I_{i-1}(x, \vec{\omega}_V, \vec{\omega}_L, \alpha), \sigma_I)}_{\text{previous opacity of shadow map}} - \underbrace{W(I_{i-1}(x, \vec{\omega}_V, \vec{\omega}_L, \alpha), \sigma_I)}_{\text{previous opacity of shadow map}}^2. \quad (5.27)
 \end{aligned}$$

Filtering of the opacity channel $I(x, \vec{\omega}_V, \vec{\omega}_L, \alpha)$ is *decoupled* from the filtering of the color channels $I(x, \vec{\omega}_V, \vec{\omega}_L, \lambda)$ of the shadow map which is used for the computation of the indirect light. Indirect light with a chromatic component of attenuation is computed by:

$$\begin{aligned}
 & \underbrace{I_i(x + \Delta s \vec{\omega}_L, \vec{\omega}_V, \vec{\omega}_L, \underbrace{\lambda}_{\{R,G,B\}})}_{\text{new chromatic shadow map}} = \\
 & \quad \underbrace{\text{average chromatic indirect light intensity}} \quad \underbrace{\text{filtered chromatic attenuation}} \\
 & = \underbrace{W(I_{i-1}(x, \vec{\omega}_V, \vec{\omega}_L, \underbrace{\lambda}_{\{R,G,B\}}), \sigma_I)}_{\text{previous chromatic shadow map}} \underbrace{(1 - \underbrace{\alpha_\lambda}_{\{R,G,B\}} \cdot \underbrace{I_{i-1}(x, \vec{\omega}_V, \vec{\omega}_L, \alpha)}_{\text{shadow map opacity}})}_{\text{shadow map opacity}}. \quad (5.28)
 \end{aligned}$$

Although this approach is not physically correct, it provides enhanced color bleeding effect in the shadow areas (see Figure 5.11(b)). Equation 5.27 and Equation 5.28 correspond to the compositing operators of indirect light in the fetoscopic illumination model.

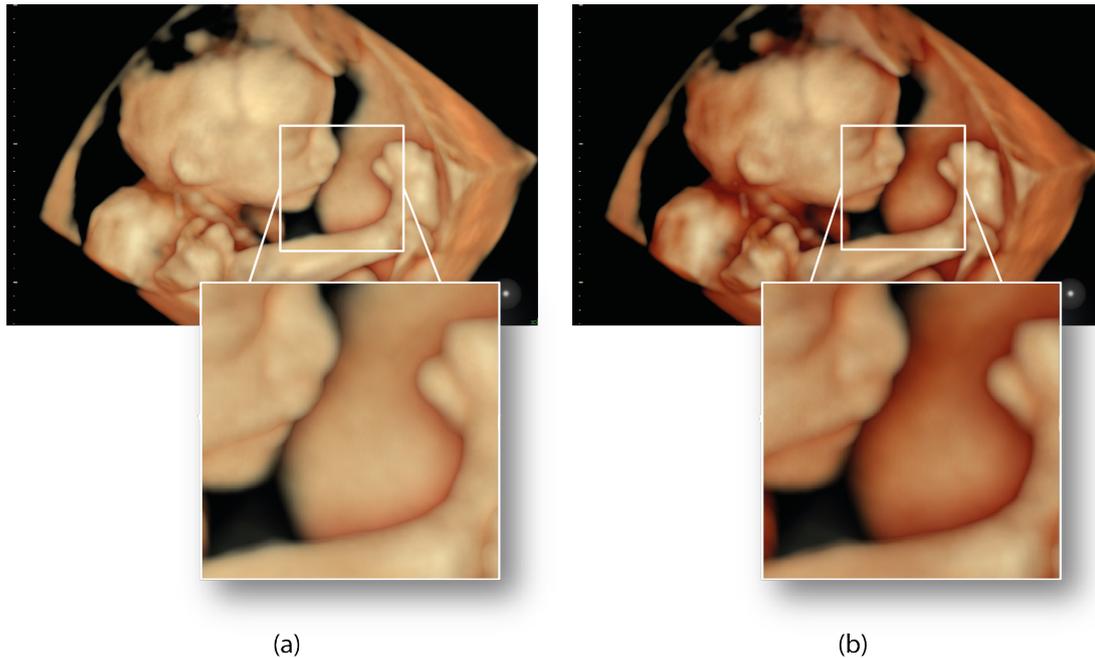


Figure 5.11: Illumination with the light source in the position of the viewer. Illumination (a) without filtering (see Equation 5.25) and (b) with filtering (see Equation 5.28) of the decoupled chromatic alpha component.

Direct Light: Hard Shadows and Specular Highlights

In the next step, we incorporate into our model the global illumination effect of hard shadows. Hard shadows are caused by a single scattering of the partially attenuated light coming from an external directional light source in this case. The light contribution that is responsible for this effect in our model is called direct light. The shadows appear on surfaces of structures when the light source rotates around the scene. This effect adds more depth perception cues to the image. Hard shadows are important because they improve the perception of spatial structures [152]. Figure 5.12 shows a comparison between hard shadows and soft shadows. In our model, we achieve the hard shadows with the additional direct light component. Ropinski et al. [122] use a similar idea of decoupling of the hard shadow computation from color bleeding in their illumination model with light volumes. In their work they use a separate blur for the computation of colored light. They omit the blurring of their light intensity part in the computation of their light volume. This is done in order to preserve the hard shadows in the illumination model. In our model, we do not use light volumes because they would require a full access to the whole volume. We use an additional shadow map that is used for the iterative computation of hard shadows with the direct light. Computation of the hard shadows is synchronized with the computation of multiple scattering through the indirect light.

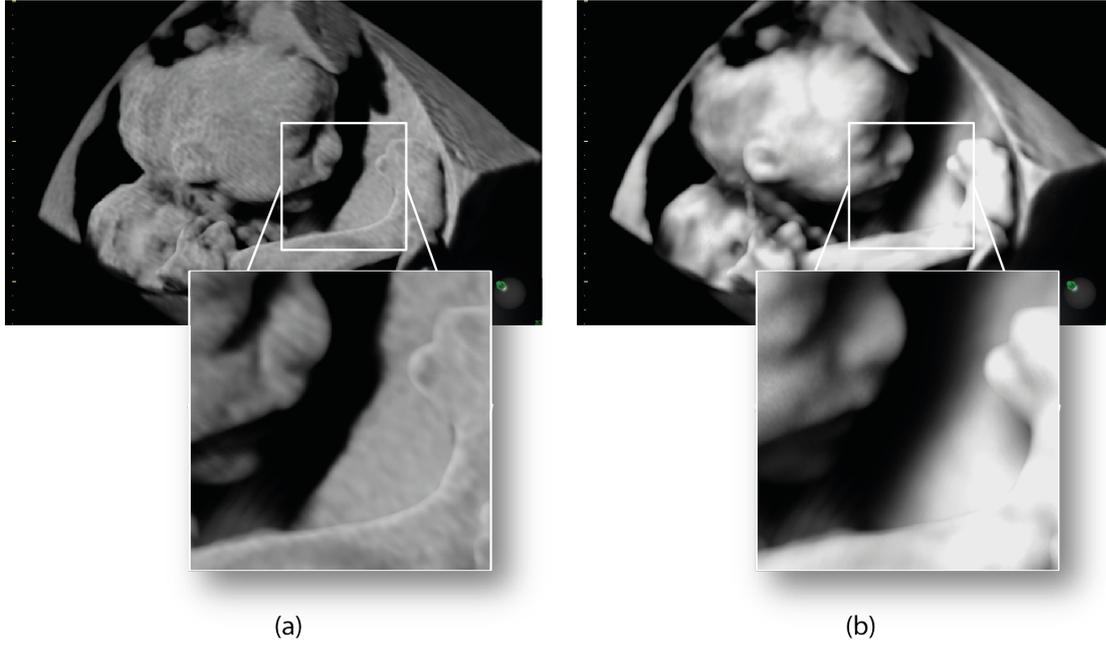


Figure 5.12: Comparison of hard and soft shadows. (a) hard shadows - single scattering of direct light, (b) soft shadows - filtering of indirect light.

Single scattering is a special case of multiple scattering. For approximating single scattering we assume additional simplifications to Equation 5.18 for multiple scattering. The direct light travels from the initial position of the light source X_L which is defined only with the initial intensity $U(X_L, \vec{\omega}_L)$ and orientation. It travels and attenuates in the participating media only along parallel light rays with the light direction $\vec{\omega}_L$. The single scattering contribution into direction $\vec{\omega}_V$ at the position s is given by:

$$U(s, \vec{\omega}_L) = U(X_L, \vec{\omega}_L) e^{-\int_0^{X_L} \tau(s-x\vec{\omega}_L) dx}. \quad (5.29)$$

It corresponds to the reduced intensity of the direct light traveling through the participating media. The direct light at the position s corresponds to the light traveling along the light direction $\vec{\omega}_L$ from the position x . The light attenuation in the interval $(x, x + \Delta s\vec{\omega}_L)$ is computed with:

$$\underbrace{U_i(x + \Delta s\vec{\omega}_L, \vec{\omega}_L)}_{\text{new shadow map}} = \underbrace{U_{i-1}(x, \vec{\omega}_L)}_{\text{previous shadow map}} \overbrace{(1 - \alpha(x + \Delta s\vec{\omega}_L))}^{\text{attenuation}}. \quad (5.30)$$

where $U_{i-1}(x, \vec{\omega}_L)$ is the direct light at the previous position x and $\alpha(x + \Delta s\vec{\omega}_L)$ is the opacity at the new position $(x + \Delta s\vec{\omega}_L)$. The computation of the lighting for the direct light corresponds to single scattering and in terms of ray tracing it can be understood as tracing a secondary ray from every position s into the light direction $\vec{\omega}_L$. Equation 5.30 corresponds to the compositing operator for the direct light computation.

Part of the light that is hitting the surface of the skin is directly reflected in the view direction. This type of reflection is called *specular* reflection. It creates the wet appearance of the oily skin surface. This type of light interaction belongs also to the single scattering. This component has an important impact on the surface perception. Specular reflection is a single scattering of the light. However we can notice that illumination with direct light only, which corresponds to the computation of single scattering, does not approximate this effect (see Figure 5.12(a)). Therefore we add to our model the additional effect of specular highlights. Specular highlights correspond to the light that directly reflects from the surface into the view direction. The specular reflection can be computed based on the direct light. It can be approximated by the dot product between the surface normal $\vec{N}(s_i)$ and the halfway vector $\vec{H}(\vec{\omega}_V, \vec{\omega}_L)$ between the viewer and the light (see Equation 2.20 in Section 2.8). The surface normal is usually approximated by the gradient vector. With global illumination we have to consider the specular component only in the areas that are illuminated with direct light. Therefore we compute the specular component as:

$$S(s_i, \vec{\omega}_V, \vec{\omega}_L) = \underbrace{U(s_i, \vec{\omega}_L)}_{\text{shadow map sample}} \cdot (\vec{H}(\vec{\omega}_V, \vec{\omega}_L) \cdot \vec{N}(s_i))^p, \quad (5.31)$$

where p is a specular coefficient controlling the specularity of the material.

The quality of the computed gradients is crucial for the visual quality of the specular highlights. The noisy character of the original ultrasound data degrades the quality of the gradients. Figure 5.13 compares the computation of specular highlights from the original and the filtered version of data. Therefore, we use the filtered version of the data for the gradient computation in the fetoscopic illumination model. This provides us with a better quality of the specular reflection effect from the skin surface of the fetus.

Ambient Occlusion

An ambient term is typically used in local illumination models to account for ambient light and to illuminate samples regardless of the light source orientation. The ambient term is used as a coarse approximation of the multiple scattering. Traditionally the ambient light contribution is represented by a constant factor and it is used to add more light to the illumination of the object. However, a simple application of a constant ambient term for illumination does not enhance the illumination of the object and rather degrades the quality of the rendered images. We want to enhance the illumination with the effect that enhances the shape of objects only when the external light source does not illuminate them directly and images appear too dark. Silhouettes of objects can provide additional cues for visual perception of their dark shape in such scenarios. Therefore, instead of a constant ambient term, we use a term corresponding to the local ambient occlusion (LAO) as proposed by Hernell et al. [57]. The LAO is robust to noise and it can add an appearance of silhouettes. This effect becomes more prominent when the light is positioned behind the scene (see Figure 5.14 and Figure 5.15). In contrast to the computation of multiple scattering with indirect light, the advantage of the LAO is that it can be computed only from the local neighborhood around the sample.

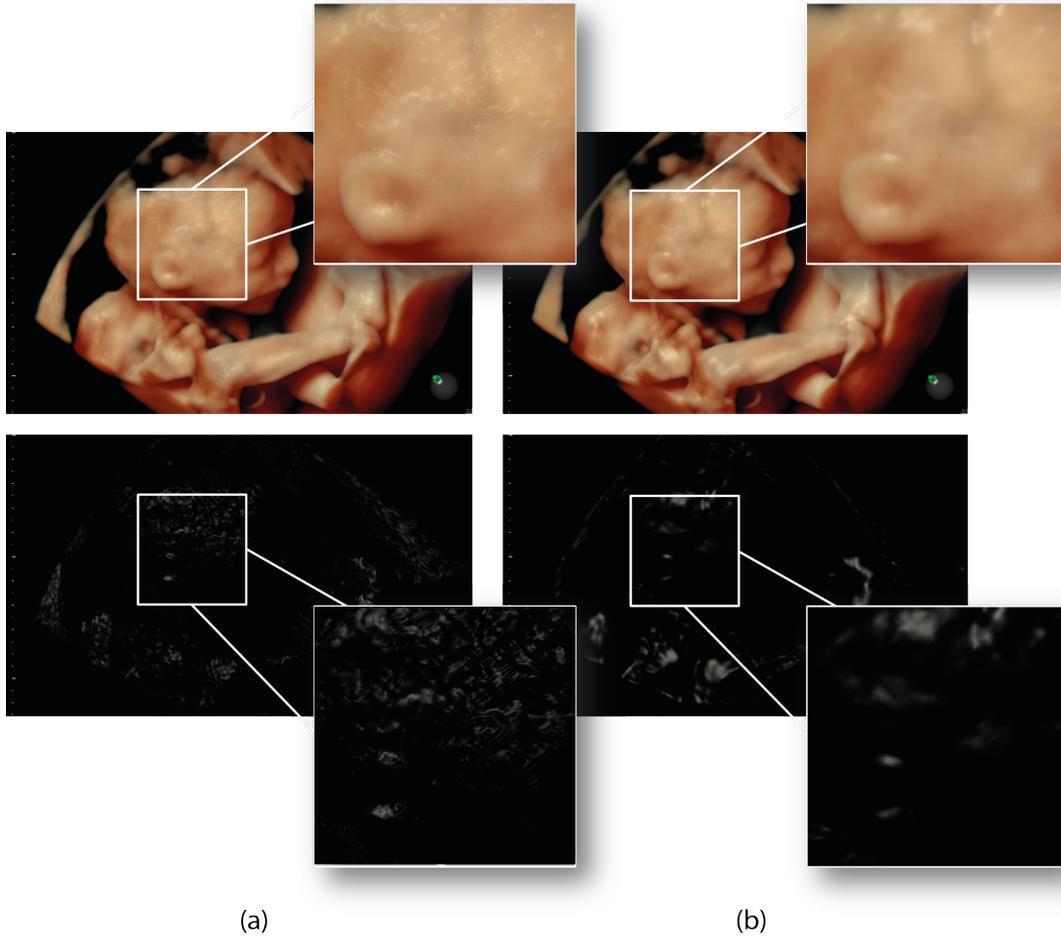


Figure 5.13: Comparison of the quality of gradients for the computation of specular highlights. (a) gradients computed from original data. (b) gradients computed from filtered data.

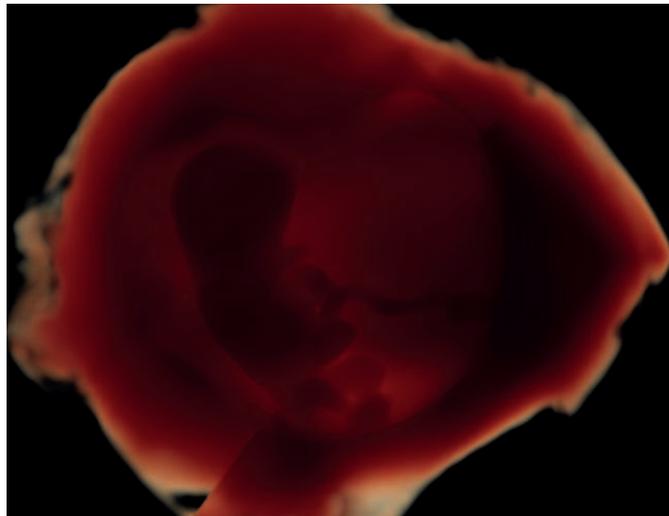
The LAO computes the incident light coming from all directions $\vec{\omega}_m$ in a local neighborhood that is in-scattered into the view direction. The local ambient occlusion can be written as:

$$L_A(s, \vec{\omega}_V) = \int_{\Omega} L_A(d(s, X_A), \vec{\omega}_m) e^{-\int_0^{X_A} \tau(s-x\vec{\omega}_m) dx} d\vec{\omega}_m \quad (5.32)$$

We compute this illumination component based on the attenuated incident light coming from the local neighborhood within a small distance $d(s, X_A)$ around the position s . Initial intensity of the ambient light is $L_A(d(s, X_A), \vec{\omega}_m)$. In our model we consider only light attenuation, without an emissive contribution around a sample. The LAO is numerically approximated as an average of contributions $L_m(s_i, \vec{\omega}_m)$ of in-scattered light from M directions $\vec{\omega}_m$ around the sample s_i



(a)



(b)

Figure 5.14: (a) Illumination of the fetus with the light positioned in (a) front and (b) behind the scene. Image features become too dark when the light is coming from the back.

as:

$$A(s_i, \omega_V) = \frac{1}{M} \underbrace{\sum_{m=1}^M L_m(s_i, \vec{\omega}_m)}_{\text{average light contribution}}, \quad (5.33)$$

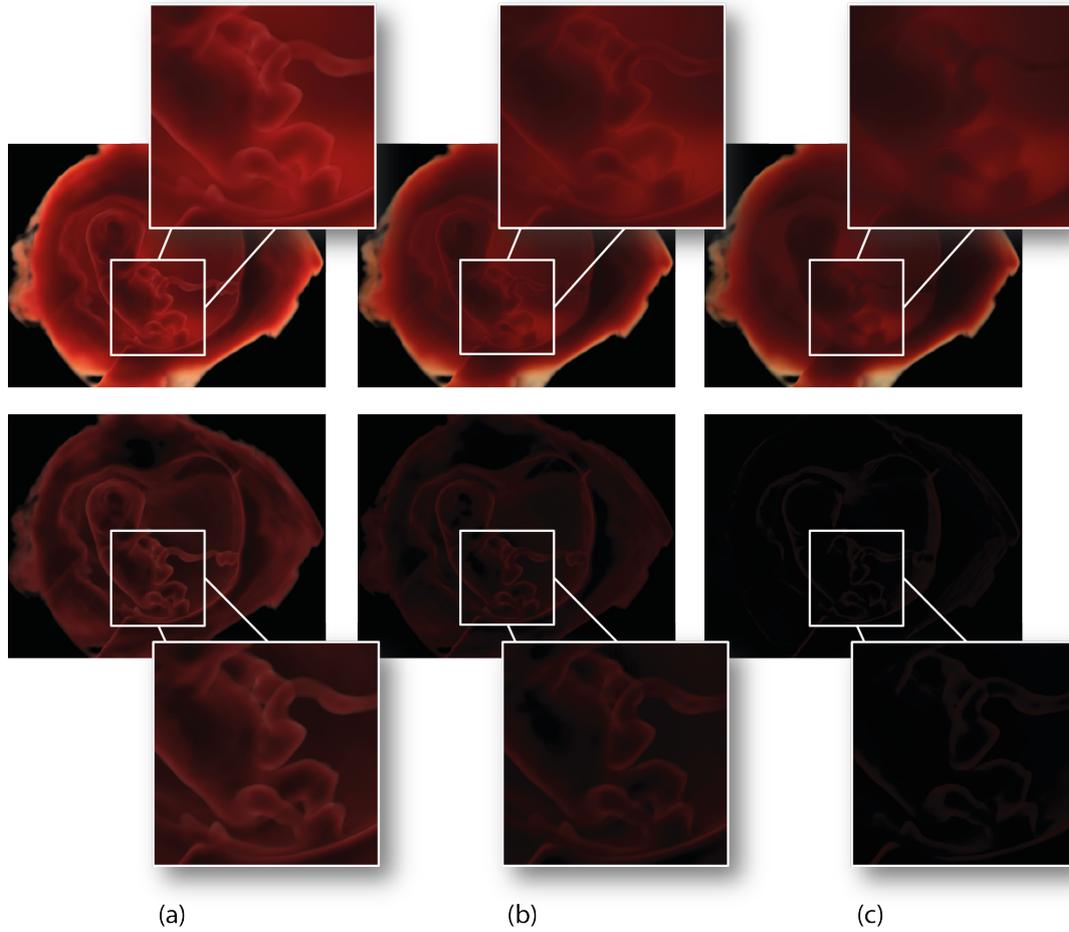


Figure 5.15: Local ambient occlusion with different levels of opacity modulation (see Equation 5.35) creates the appearance of silhouettes of the fetus. The bottom row illustrates net contribution of the effect. (a) opacity of the samples is the same as in original TF used for DVR $r = 1$, (b) increased opacity with $r = 0.6$, (c) increased opacity with $r = 0.3$.

where M is a number of rays with light directions and $\vec{\omega}_m$ are in-scattered light directions.

$$L_m(s_i, \vec{\omega}_m) = \underbrace{L_m(d(s_i, X_A), \vec{\omega}_m)}_{\text{initial LAO intensity}} \prod_{j=1}^{X_A/\Delta s} \underbrace{(1 - \alpha_A(s_i - j\Delta s\vec{\omega}_m))}_{\text{attenuation along ray}}, \quad (5.34)$$

where $L_m(d(s_i, X_A), \vec{\omega}_m)$ is the initial light coming from the distance $d(s_i, X_A)$ along the direction of the ray $\vec{\omega}_m$ and it is initialized to a constant value. Light is attenuated at each sample according to the opacity mapping $\alpha_A(s_i - j\Delta s\vec{\omega}_m)$ that is computed according to the opacity

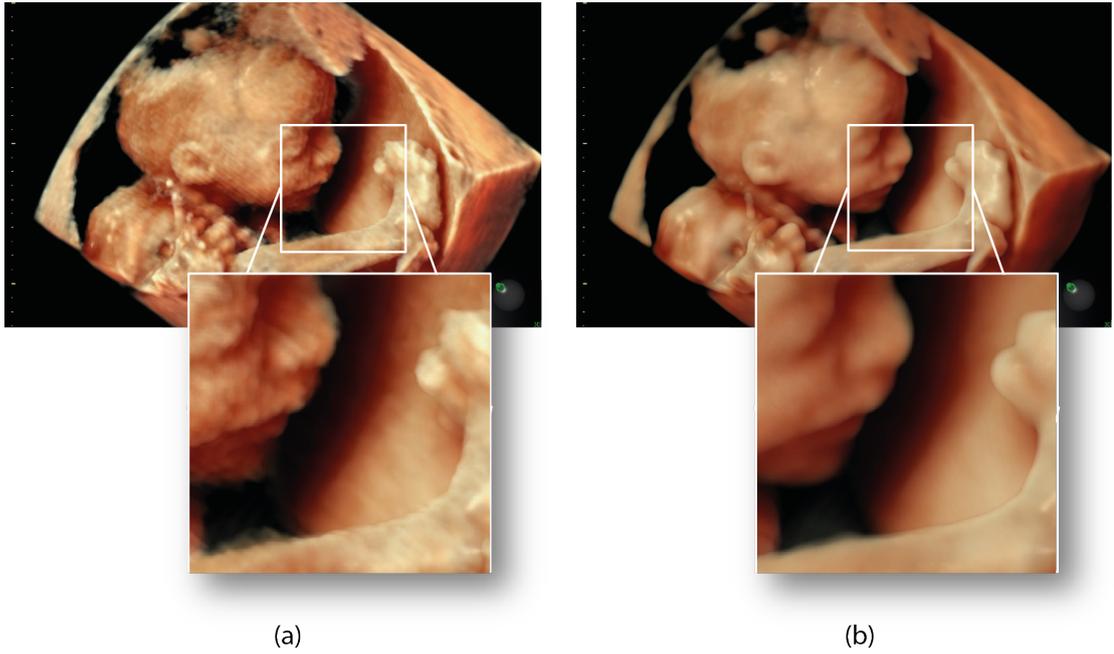


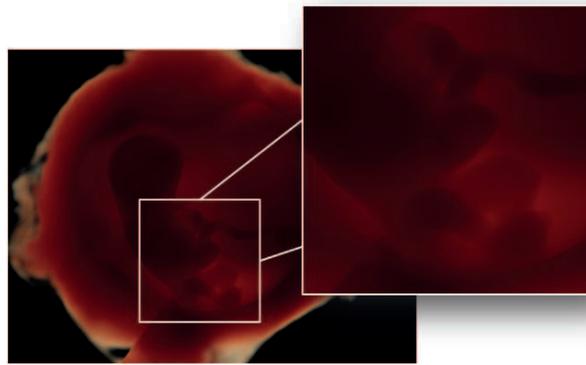
Figure 5.16: Fetoscopic rendering with (a) original and (b) filtered data.

modulation. A silhouette effect can be achieved if the opacity $\alpha_A(s_i - j\Delta s\vec{\omega}_m)$, assigned to samples for computation of the ambient occlusion term, is higher than the opacity of samples used for the evaluation of the volume rendering integral $\alpha(s_i - j\Delta s\vec{\omega}_m)$. Therefore, we modify the opacity for the LAO computation in our model according to:

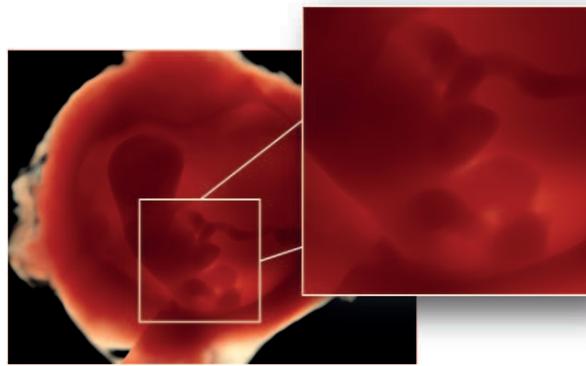
$$\alpha_A(s_i - j\Delta s\vec{\omega}_m) = \alpha(s_i - j\Delta s\vec{\omega}_m)^r, \quad (5.35)$$

where r is a coefficient between zero and one that modulates the opacity $\alpha(s_i - j\Delta s\vec{\omega}_m)$ assigned by the original transfer function. Figure 5.15 illustrates the impact of this parameter in rendered images.

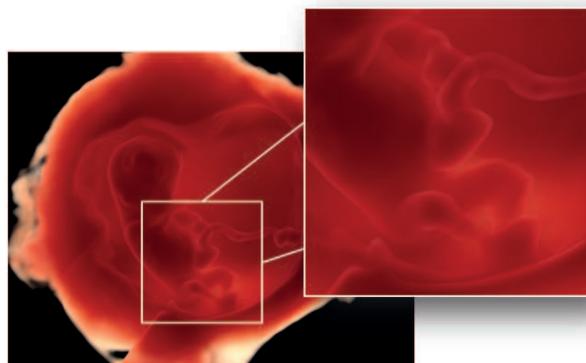
The rendering stage of the pipeline renders the images from the original and filtered version of the data. The clinician can view images rendered from both versions of the data (see Figure 5.16). According to the level of noise that is present in the data, the clinician applies a blending between the filtered and original version. The blending factor is chosen usually in the way that it enhances important features of the data while it suppresses the noise. The next stage of the pipeline applies final corrections to the output image that appears on the screen of the ultrasound machine. The corrections are applied for improving the image quality and to ensure that the lighting and shading are displayed correctly.



(a)



(b)



(c)

Figure 5.17: Illumination of the fetus from the back with HDR post-processing. (a) illumination from the back without additional effects, (b) tone-mapped HDR image of DVR rendering produced with the Reinhard tone-mapping operator, (c) HDR image with the LAO effect.

Post-processing with High Dynamic Range Imaging

Images displayed with our rendering method contain a wide range of light intensities that result from different lighting conditions. Some results lose details because of the light attenuation and would require a manual brightness and contrast adjustment by clinicians. The implementation of the external light source allows to interact with the position of the light source. For example, it is possible to position the light source behind the scene to enhance the effect of translucency. If the light is coming from behind, sometimes features of the image are too dark and difficult to distinguish (see Figure 5.17). It is difficult to adjust image properties manually, during the live scan examination with ultrasound. The limited range of the ultrasound display can also be inadequate to present results of high precision volume rendering. High dynamic range (HDR) methods are routinely used in digital photography as a post-processing step to adjust images with a wide range of intensities. HDR methods allow to work with images with greater dynamic range of the luminance than the standard, low-dynamic-range (LDR) digital imaging techniques. Tone-mapping techniques map the high range of luminance of HDR images to standard devices with lower dynamic range, in a way that the local contrast between features is preserved. Yuan et al. [160] show that tone-mapping algorithms can be leveraged also for automatic adjustments of volume rendering results. In the fetoscopic rendering, we enhance features for the final display automatically by means of the tone-mapping algorithm that was proposed by Reinhard et al. [118].

This tone-mapping algorithm belongs to the category of global operators. Global operators are non-linear functions based on the luminance or other global variables and therefore are spatially uniform. This means that every pixel in the image is tone-mapped in the same way, disregarding the values of surrounding pixels. The algorithm was chosen because it can provide good results and it can also generate images at interactive frame rates.

5.5 Implementation

We integrated our system into the latest generation of GE Healthcare ultrasound imaging systems. It is available as the *HDlive* imaging tool within the Voluson E8 and the Voluson E8 Expert systems. The specification of the ultrasound machine and the hardware components are designed for a reliable application in a medical environment. The operating system running on the machine is developed in parallel with the hardware components. After the development cycle, all software and hardware components undergo a thorough testing. An institute responsible for accreditation has to approve the ultrasound machine before its application in a medical environment. This whole process defines the constraints for the hardware components and technologies that are available for utilization.

We developed our workflow within the ultrasound machine framework in C++ using DirectX 9.0. The workflow was optimized for live 4D ultrasound rendering in obstetrics. In order to achieve an optimal performance of the system we had to maximize the utilization of all resources. All stages of our workflow run on the GPU that is exploited for the rendering and also for the acceleration of general purpose computations. We achieve an interactive performance

between 15-20 FPS on the ultrasound machine. This performance on the current machine is limited by the acquisition speed of the transducer and not the rendering algorithm.

The implementation of the rendering algorithm, with the illumination model based on half-angle slicing, required special considerations for the ultrasound data flow through the pipeline. First, the scan conversion process of the original data was modified to accommodate the half-angle slicing algorithm. The algorithm assumes a virtual light source that can be rotated around the scene. Light scattering effects are computed by means of shadow maps which are used for the evaluation of the light propagation through the volume.

Half-angle slicing requires the orientation of the slices to be orthogonal to the half-way vector between the light source and the viewer (see Section 2.8). Therefore, we perform the scan conversion with respect to the orientation of the half-way vector (see Section 3.4). Besides the orientation of the slices, the integration order, i.e., front-to-back or back-to-front, of the slices for volume rendering also depends on the light position. For a position of the light in the viewer's hemisphere, the slices are integrated from front-to-back, using the under operator given in Equation 2.15 and Equation 2.16. When the light is positioned behind the scene the integration order changes to back-to-front, using the over operator given in Equation 2.13 and Equation 2.14. In general there is no need to iteratively compute the accumulated opacity for the back-to-front compositing and the color contribution can be evaluated without it. However, the accumulated opacity is later used for the purpose of distinguishing the object from the background. This information is used in order to achieve a correct average luminance estimation for the purpose of post-processing with HDR imaging that is applied to the image after rendering. We modify the order of the scan conversion when the virtual light source switches from the front to the back hemisphere and vice versa. We store the result of the scan conversion from the curvilinear grid per slab in a texture. The filtering stage of our workflow performs filtering of the original scan-converted data. The filtering is efficiently computed on all slices of the slab in parallel. We store the filtered result from the original scan-converted data in a second texture. The resulting textures are streamed to the rendering stage.

Both compositing operators are two different ways how to numerically compute the volume rendering integral, defined in Equation 5.4, in discretized form. The volume rendering integral is evaluated along the viewer's direction $\vec{\omega}_V$ and it corresponds to computing light contributions along each ray discretely with a fixed sampling distance. The image is composited in the image buffers for the viewer using the aforementioned blending operators. We allocate two output buffers for the resulting image integration. One output rendering buffer holds the result from rendering the original scan-converted data and the other one holds the results from rendering the filtered version of the data.

Samples are illuminated with the light contribution arriving to the sample location computed using Equation 5.16. Light contribution has several components (see Figure 5.4). They are computed according to Equation 5.17. Final light contribution is obtained as a weighted sum of the individual light components. Indirect light comes from multiple directions and is in-scattered towards the view direction $\vec{\omega}_V$. The average intensity of the indirect light is computed by the compositing operators given in Equation 5.27 Equation 5.28. Direct light arrives at the sample s_i from the light direction $\vec{\omega}_L$, which is defined by the orientation of the light source. Direct light is computed according to the compositing operator given in Equation 5.30. Separate shadow

maps for the indirect and the direct light accumulation are required. Two sets of additional off-screen render buffers are used as shadow maps. They accumulate direct and indirect light that propagates through the volume. We initialize the light buffer for direct light before the initial rendering pass with the desired light intensity value. Color channels of the shadow map for indirect light are also initialized with a uniform light intensity value (see Figure 5.20(a)). Non-uniform initialization of the direct and indirect light shadow maps with a distance based function (see Equation 2.22) creates effects of spotlights (see Figure 5.20(b)). We initialize the alpha channel of the shadow map for indirect light with zero. This channel accumulates the filtered chromatic component and it is used for the computation of the chromatic attenuation of the light. The light propagation is computed in a lock step with the integration into two output rendering buffers for the observer. The light propagates in the volume along the preferential light direction $\vec{\omega}_L$ from the external light source. Filtering of the shadow map for indirect light produces light diffusion in one direction according to the forward scattering phase function for scattered indirect light. We filter the shadow map for indirect light by a weighted average of the samples in a neighborhood around the current sample (see Equation 5.28). The weighted average filter is applied in order to account for a difference in the scattered light intensity that is reduced with the distance as the light travels in the participating medium. Chromatic attenuation of the light in the human skin is approximated using the samples that are taken from the filtered alpha channel of the shadow map (see Equation 5.27). This channel stores the chromatic component. Samples taken from this channel are multiplied with the indirect alpha (see Equation 5.26) in order to chromatically attenuate the indirect light propagation. Filtering of the color channels of the shadow map for indirect light is decoupled from the filtering of the alpha channel. This means that in our model, the phase function is evaluated implicitly by filtering values from the light buffers.

In the iterative multi-pass algorithm, in one pass we render the slice into the output buffers. In the next pass we compute the further propagation of the direct and indirect light in shadow maps (see Figure 5.18). This approach is also sometimes called ping-pong rendering and it belongs to the category of slice-sweeping algorithms. It is compatible with our streaming visualization pipeline.

Specular highlights can be computed according to the direct light intensity (see Equation 5.31). In addition to the direct light intensity, we need to estimate the surface normal. For this purpose we use the gradient estimated by central differences.

The final computed light component is LAO. Similar to the gradient estimation, LAO is evaluated also from the local neighborhood around the sample s_i . We cast a few rays ($M = 9$) around the sample and evaluate the final contribution by averaging each individual ray (see Equation (5.33)). We take several samples ($K = 4$) along each ray from the local neighborhood. The rays are cast around the current sample only in the plane of the half-angle slice and not in the spherical neighborhood. This allows to compute the LAO effect also with the slice-sweeping algorithm, where neighboring slices are not always available in the memory.

After the slices from the current slab are rendered to the output buffer, the new slab is scan converted and filtered. After all slabs are integrated, we pass the two resulting image buffers to the post-processing step for the final adjustments. The blending between the two images rendered from the original and the filtered version of the data is performed, to improve the

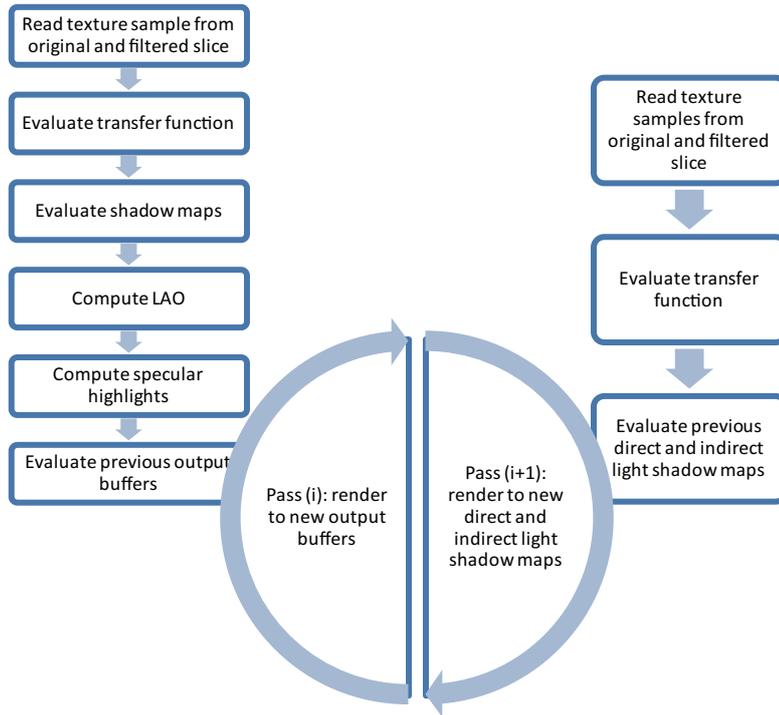


Figure 5.18: Multi-pass rendering algorithm renders the volume iteratively into the output buffers, with an incremental computation of light propagation in the shadow maps.

visual quality of the final rendering. In the next step, we perform the HDR tone mapping in order to enhance the contrast of the output images.

Numerical precision has an impact on the results obtained from the numerical computation of the volume rendering integral. If the volume rendering method is computed with high precision numbers, the rendered images can preserve more details between the regions with different attenuation. In essence, it is producing HDR images with a high range of luminance. By using HDR tone-mapping, the contrast of the images can be automatically adjusted to enhance features that are otherwise not visible. The current volume rendering method is using floating point buffers to increase the precision of the rendered images.

Initially, the log-average luminance is computed in order to estimate the global variable for the tone mapping. For the purpose of volume rendering, the average luminance has to be computed only for pixels that belong to the object, rejecting pixels of the background. Log-average luminance \bar{L}_W is computed with a parallel reduction operation, reducing image dimension by two in every step until a single value is achieved. It is given by:

$$\bar{L}_W = \frac{1}{N} \exp \left(\frac{\sum_{u,v}^N \alpha(u,v) \log(L_W(u,v) + \delta)}{\sum_{u,v}^N \alpha(u,v)} \right) \quad (5.36)$$

where N is the total number of pixels, δ is a small value to avoid an invalid input to the log function that occurs for black pixels. Information about the background is stored in the alpha channel $\alpha(u, v)$ accumulated during integration and it is used to weight pixels. $L_W(u, v)$ is the 'world' luminance for pixel (u, v) and it is obtained from the red I_R , green I_G and blue I_B color triplets of the image $I_{RGB}(u, v)$ with:

$$L_W(u, v) = 0.2125I_R(u, v) + 0.71254I_G(u, v) + 0.0721I_B(u, v) \quad (5.37)$$

The luminance scaled image $I_S(u, v)$ is computed by mapping of the image pixels $I_{RGB}(u, v)$. The image $I_{RGB}(u, v)$ is an image obtained by blending of the two images rendered with the aforementioned DVR algorithm from the original and the filtered data. The blended image $I_{RGB}(u, v)$ is mapped with log-average luminance \bar{L}_W through:

$$I_S(u, v) = I_{RGB}(u, v) \frac{a}{\bar{L}_W} \quad (5.38)$$

where $a = 0.5$ is a user defined scaling factor on a scale from zero to one.

Finally a tone-mapping operator, as proposed by Reinhard et al. [118], is applied uniformly to each pixel:

$$I_D(u, v) = \frac{I_S(u, v) \left(1 + \frac{I_S(u, v)}{L_{white}^2}\right)}{1 + I_S(u, v)} \quad (5.39)$$

where L_{white} is the smallest luminance that is mapped to pure white. This tone mapping operator maps all luminances of the luminance scaled DVR image $I_S(u, v)$ image into the range of the display $I_D(u, v)$. After the post-processing step the final image is displayed on the screen.

5.6 Results and Discussion

Live fetoscopic rendering is a novel medical imaging method for the visualization of the human fetus in obstetrics. During the design and development phase, we tested our method on a wide variety of real datasets coming from different stages of pregnancy. The method was integrated and approved for clinical application with the current generation of ultrasound machines. It is currently available and used in practice in several prenatal imaging centers [91] [115] [126]. Our system achieves an interactive performance between 15-20 fps on live 4D ultrasound examinations with current machines. This performance is mainly limited by the acquisition speed of the ultrasound data with the transducer. It allows a live realistic visualization of the moving human fetus in a clinical environment from ultrasound data.

Ultrasound is performed in various weeks of pregnancy in order to diagnose the evolution and the health of the fetus. There are various guidelines for examinations of fetal anatomy that are different in different countries. We investigated our new method in a discussion with a small group of domain experts. Based on their experience with the new method, they provided us with feedback and gave us their opinions.

Although our model has a potential to display the images of the fetus with more realism, we had to customize it with ultrasound domain experts in order to render convincing images with

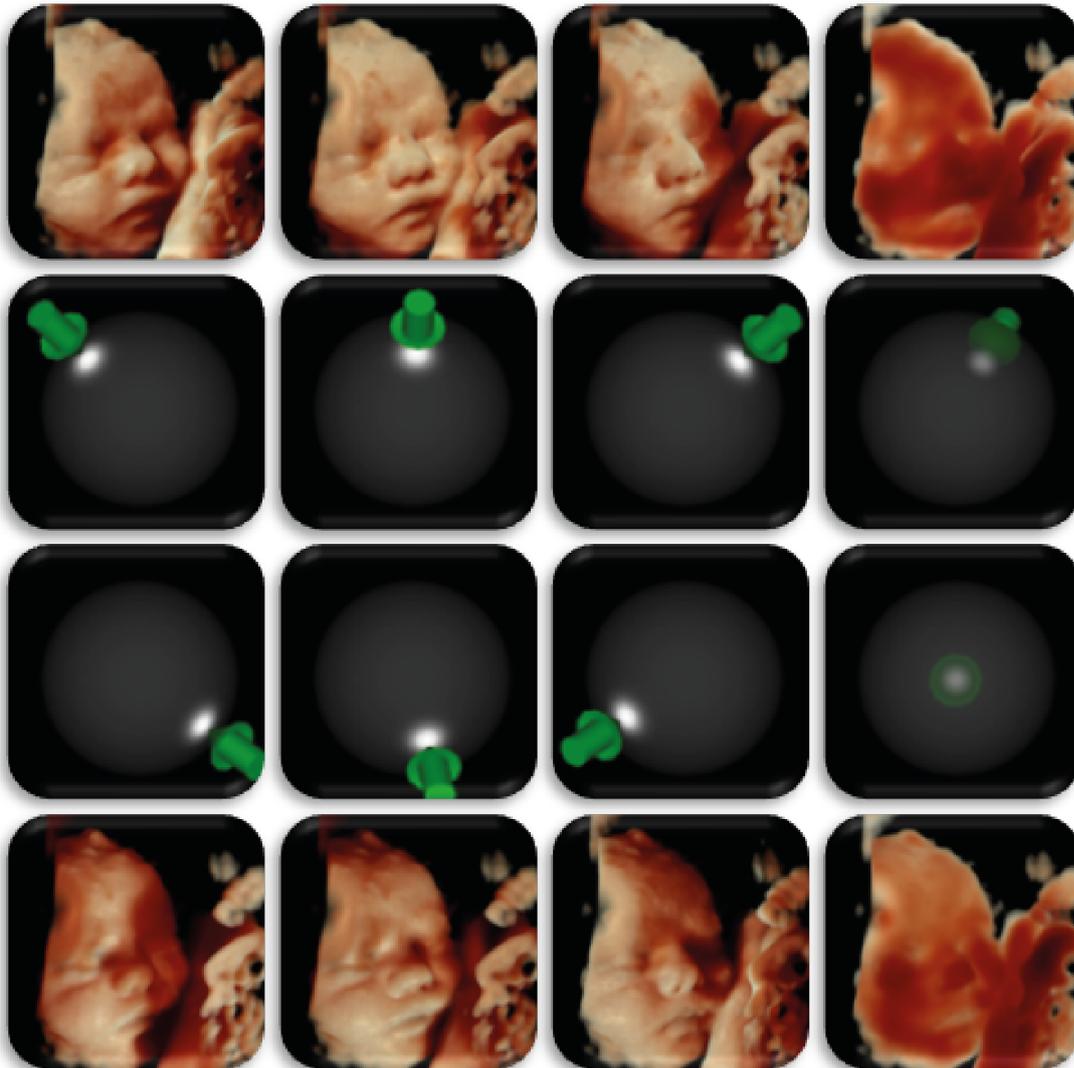


Figure 5.19: Different light positions in the scene. The current light position is always indicated with a sphere glyph to make the navigation with the virtual light source easier for the user.

ultrasound. The number of parameters of our model defines a parameter space which is non-intuitive for users to explore without assistance. We worked closely with a group of ultrasound domain experts which guided us through the customization of the life fetoscopic renderer. In the parameter study, we tried to find optimal values of parameters for our illumination model. The main parameters of our model include:

- position of the light source



(a)



(b)

Figure 5.20: Comparison between (a) the area light source and (b) the spotlight illumination with the fetoscopic rendering.

- intensity of the light
- shadow softness
- balance between hard shadows and soft shadows
- optimal color of the skin

Our illumination model allows to change the position of the virtual light source. However, it was difficult to navigate the position of the light in 3D virtual space especially during the live examination of the fetus. We implemented an arc ball controller [132] that allows to rotate the light source in the scene with convenient interaction operations. Navigation of the light source with the arc ball can also be mapped to the trackball of the ultrasound machine. To further minimize the amount of interaction we predefined several presets of light positions that are directly available to the user. The users found it important to know the actual position of the light

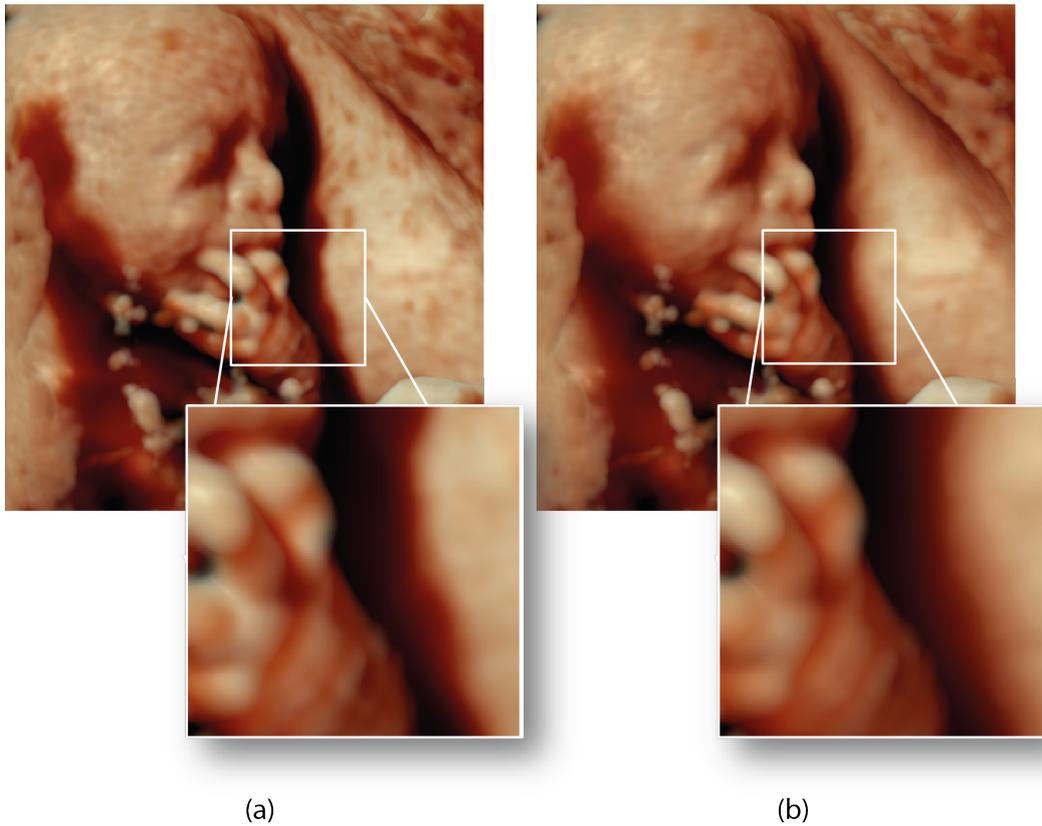


Figure 5.21: Comparison of images rendered with different shadow softness value, with (a) harder shadow borders, (b) softer shadow borders.

source in the scene in order to have a confident control of it. For that reason we implemented a sphere glyph that indicates the actual position and makes the navigation easier. Figure 5.19 illustrates different light positions in the scene. The current light position is always indicated as a bright spot on a shaded sphere with an arrow which helps the user to easily perceive the current light orientation.

We also tried to approximate the illumination of the scene similar to employing the spotlight of the real fetoscope. We can simulate a spotlight illumination effect in our model by non-uniform initialization of the light intensity. Figure 5.20 shows the effect of the spotlight light source with our model. To achieve this effect, we use a non-uniform initialization of the shadow maps as described in the previous section. However, our discussion with clinicians indicated that they prefer an uniform illumination. The possibility to control the overall intensity of the light source was accepted as useful option.

We tried to understand what appearance shadows should have in live fetoscopic rendering. Our model has a parameter that can control the effect of soft shadows. Shadows appear in the image because of the two types of light that we use for the illumination of the scene. Direct light

is responsible for the appearance of hard shadows. Soft shadows appear due to the application of indirect light. We control the diffusion of indirect light by the size of the filter kernel that is used for blurring the indirect-light shadow-map. Larger filters are responsible for softer shadows. Figure 5.21 shows a comparison between shadows with various softness levels. We introduced the control of the shadow softness to our domain experts. The possibility to control the shadow softness was accepted as useful option as well.

The transfer-function specification for our illumination model was also non-intuitive for the clinicians. In the customization cycle of our model, we adjusted the transfer-function parameters according to the desired changes by the cooperation partners to find the optimal configurations for the color tones of the fetal skin. In our discussions with the domain experts, we provided the clinicians successively with images rendered with various configurations of parameter values. To converge our exploration of the parameter space to an optimal parameter configuration, we asked clinicians for a short assessment of the visual quality and indications concerning clinical relevance. In our discussions the clinicians were provided only with the images produced by the live fetoscopic renderer.

Figure 5.22 (a) displays the fetus in early pregnancy with a strong emphasis on soft shadows and scattering. We presented this visualization to the clinicians and asked them about their opinion on the appearance of the image. Three of our clinicians did not like the level of detail and it was difficult for them to assess the depth relations because the image appears too dark. Two of them liked the enhanced detail of the umbilical cord, the early ear buds and the limbs. Only one of the experts considered the image to be very similar to a real fetoscopic image.

We decreased the amount of soft shadows in the parameter settings of our model. Figure 5.22 (b) shows the detailed image of the ultrasound scan of a foot which is visualized with a decreased level of shadow smoothness. One of the clinicians did not like the amount of soft shadows. Two clinicians considered the perception of the toes with this visualization to be enhanced. One clinician commented that the angle between the foot and the leg can be easily recognized. The reduction of shadow smoothness was appreciated because it led to the enhancement of borders and a better perception of details. However, the image seems to be too bright for the clinicians and they wanted to change the color tone. A conventional visualization with local illumination is shown just for illustration in Figure 5.26(a). Clinicians were not provided with this visualization in the comparison.

To change the color tone, we modified the transfer function. We presented the visualizations with modifications of the color tone of the skin. Figure 5.23(a) visualizes the embryo in the first quarter of gestation with the modified skin-color transfer-function. The image did not appear realistic for two of our clinicians because the skin color-tone is overall too pink. One clinician considered the shadow softness still to be too much. The others could identify anatomical structures of the fetus, e.g., the yolk sac, the early limb buds and the umbilical cord, with confidence and better understanding of depth cues. They also pointed out that the configuration with the light illuminating the scene from the back (see Figure 5.23(b)) is important as it enhances translucent tissue structures of the fetus in this stage of pregnancy. Conventional visualization with local illumination is shown just for illustration in Figure 5.26(b).

In Figure 5.24(a), which depicts the 3D face of a fetus, we tried to present a more realistic tone of the skin. The modification to the transfer function of the skin color was well accepted



(a)



(b)

Figure 5.22: (a) Live fetoscopic rendering of the fetus in early pregnancy resembles visual features of images coming from fetoscopy. The umbilical cord, the early ear buds and the limbs can be studied on this image, (b) live fetoscopic rendering of a foot of the fetus. The angle between the foot and the leg can be easily recognized. Soft shadows enhance toes on the foot.

by all clinicians. One of the experts wanted to decrease the shadow softness for the image of the fetus. A conventional visualization with local illumination is shown just for illustration in Figure 5.26(c).

We presented in Figure 5.24(b) another face of a fetus with the same skin color as in the previous case but with a decreased level of soft shadows. The experts welcomed the change to the parameter of shadow softness because all important facial features were presented with a sufficient amount of detail. The clinicians commented on the clear perception of the profile of the face and easy recognition of lips, chin, and fingers of the fetus.

After finding the optimal transfer function for the skin and a balanced shadow smoothness we tried to identify the optimal positions for the light source. Figure 5.25(b) shows the light source



(a)



(b)

Figure 5.23: Live fetoscopic rendering of a fetus in the first quarter of gestation, (a) light in front of the scene, (b) light in the back of the scene creates the effect of translucency.

positioned at the top-left in front of the embryo scanned in the first trimester. This corresponds to the standard configuration for scene illumination used by medical illustrators. The clinicians appreciated this light source position because of the enhanced depth perception of the shape of the head and limbs of the fetus. They also asked an option to quickly switch to this light configuration with our model. Therefore we predefined a set of favorite light positions for the quick navigation with the ultrasound machine's interface.

Clinicians also asked for skin presets of different skin types. In order to achieve rendering with different skin types, we had to modify the transfer function. We provided several presets that allow to modify the skin color for Caucasian, African and Asian skin type. Figure 5.27 shows the comparison between Caucasian and African skin type.

We summarize the findings from the discussions with the group of domain experts and identify the possibilities for fetal examinations with live fetoscopic rendering in different stages of



(a)



(b)

Figure 5.24: Live fetoscopic rendering of the fetal face. (a) Live fetoscopic rendering with light in front of the face, (b) live fetoscopic rendering of the fetal profile with realistic rendering of the skin. Nose profile, lips, chin and fingers are clearly depicted.

pregnancy. The strongest potential and advantages of our method were considered in the visualization of the fetus during the *second trimester*. The images clearly depict the facial profiles and enhance perception of the nose and ears. Our method is expected to visualize the limbs and the extremities with realistic details. An improved visual perception of toes and finger details could be achieved. The chin of the fetus could also be better studied with our method as well as its longitudinal section of the spine. In the *first trimester* the new method has a potential to



(a)



(b)

Figure 5.25: Fetus in the first quarter of gestation. (a) conventional rendering method, (b) live fetoscopic rendering.

visualize the plant and the presence of the gestational sac, the umbilical cord insertion and buds of the limbs. In this stage of pregnancy, the profile of the fetus can be enhanced and the length of the fetus can be measured. The yolk sack could also be expressively visualized in early stages of pregnancy (4-8/10 weeks).

The clinicians who tested and worked with our system also identified the advantage of the illumination with a movable virtual light source in comparison to the conventional method. This method implements only the local illumination model. The position of the light source is static

and the fetus appears unrealistic on the images rendered with this method. The possibility to adjust lighting helps to reveal fine details. Better visual cues could also help to reduce examination time and enrich clinician-patient communication.

In general, the clinicians identified a lot of advantages of the new imaging tool. They appreciated the depiction of the fetus with anatomical realism and increased depth perception. The method can help them to achieve a deeper understanding of anatomic relationships. This has a potential to enhance diagnostic confidence and could furthermore facilitate communication with patients. Clinicians also liked the possibility to adjust the position of the light source because it allowed them to better study the structures of the fetus.

Examinations with conventional rendering do not always reveal abnormalities of the fetus in sufficient detail. The method has the potential to help with the diagnosis of malformations of the fetal anatomy. This includes head proportions, malformations of nose and face profile, cleft lip, club foot, presence of toes and fingers.

The novel rendering provides a new way for the analysis of the health of the fetus. *HDlive* was for the first time publicly showcased at the International Society of Ultrasound in Obstetrics and Gynecology (ISUOG) congress 2011 in Los Angeles [116]. The new rendering method was reportedly presented also at the Radiological Society of North America (RSNA) conference 2011 in Chicago [34]. Kagan et al. [66] demonstrated the diagnostic potential of the new imaging method and showed early results of studies performed during the first trimester of gestation. Early results with the method were also published by Merz [101]. Currently the method is publicly available and it is used in several prenatal imaging centers [91] [115] [126] and other facilities. According to the reported feedback, the method was positively accepted by patients who appreciate the visual quality of the images rendered with the new technology.

5.7 Conclusion

In this chapter, we presented live fetoscopic rendering for ultrasound data, a novel medical imaging method for the visualization of the human fetus. Our method supports interactive high-quality rendering during examinations of the moving fetus. We developed an advanced illumination model that supports shadows, a movable virtual light source, and realistic rendering of the skin. We integrated our system as the *HDlive* imaging tool within Voluson E8 and Voluson E8 Expert which are the latest generation of GE Healthcare ultrasound imaging systems. The results of our work were discussed with domain experts who provided a very positive feedback from patients. They appreciated the images produced with our system and preferred them to the images rendered with the previous method. Discussions with clinicians also indicated that the new imaging method has a potential to improve examinations with ultrasound during pregnancy. The visualizations from our imaging system could help increase the clinical confidence and achieve a better understanding of anatomical relationships in the fetus. However, this has to be further evaluated with clinical surveys.

In the future, we would like to improve further the quality of the live fetoscopic rendering with more virtual light sources. The specification of the transfer function is also a challenging task in case of scans with a low signal-to-noise ratio. The position of the light source has a strong

influence on the quality of the resulting images. Improving the robustness of the parameters of our model opens challenging questions for further research as well.



(a)



(b)



(c)

Figure 5.26: Conventional rendering method with local illumination provided for illustration. Depth shading with blue color is used for better depth perception with this method. (a) rendering of a foot, (b) fetus in the first quarter of gestation, (c) 3D fetal face



(a)



(b)

Figure 5.27: Fetoscopic rendering with different skin types. (a) Caucasian skin type, (b) African skin type.

Summary

In this thesis, we presented the results of our research in the visualization of ultrasound data coming from prenatal sonography. The work was done in cooperation with our industrial partner GE Healthcare and its domain experts. Their experience and knowledge significantly helped us to specify relevant problems and to define goals for our work. During three years of intensive collaboration, we achieved several goals that improve existing solutions and provide additional tools for the visualization of ultrasound data. Our work also revealed possibilities and limits of modern ultrasound technologies. We addressed several challenges related to the visualization pipeline of modern ultrasound machines.

The large amount of data is a challenge and requires a visualization pipeline with an adequate throughput. In Chapter 3 we proposed slice-based streaming of volume data that is the basis for the architecture of our pipeline. We decided to use streaming of slices in our data flow. This representation is a natural extension of tomographic modalities, such as US modality, and it can handle the amount of data produced by modern scanners. The way of processing the data has strong implications on algorithms that are employed in all stages of the pipeline and it had to be considered also in the selection of the rendering algorithm.

Our pipeline implements several concepts of the proposed streaming approach. It allows processing of the generated data with our algorithms at interactive frame rates. However, currently it does not support the parallel execution of all modules. Parallel computing is used only on the level of each module. All modules are executed one-by-one as the data flows through the pipeline. This restriction comes from the synchronization mechanism that is currently implemented. Our pipeline is demand driven and it triggers the execution of each module when the request for a new portion of the data happens. We rely on the parallelism inside each module. Modules are implemented as parallel algorithms for modern GPUs. In the future, it would be possible to improve the performance of the visualization pipeline by parallelizing the execution of all modules.

In Chapter 4, we described the smart visibility algorithm for prenatal sonography that can automatically detect and visualize the fetus from 3D/4D US data. It provides an alternative to the adjustable ROI box and the 'electric scalpel', which are difficult to use during scanning

because they require manual interaction. Our method can improve visibility and can decrease the amount of work required from sonographers. If the method fails, it is important to provide other tools to fix the problem. Therefore it would be required in the future, to develop quality criteria which could estimate the possibility to successfully detect the face of the fetus. More testing and evaluation can also lead to minor modifications to the basic algorithm.

In Chapter 5, we described fetoscopic rendering of ultrasound data, obtained by scanning of fetuses. Our main goal was to improve the visual quality of human prenatal anatomy from US data. We implemented a model that can render expressive images with the realistic appearance of human skin under illumination. The model has several components that approximate the desired effects. It can approximate the appearance of the skin and can be customized with a reasonable number of parameters.

Our model introduces a virtual light source that can be controlled interactively. This provides a new tool to clinicians that can help to study anatomical relationships in the fetus on the rendered images with more visual cues which support perception. The possibility to control the illumination was well accepted by the clinicians. The method is currently restricted to one external light source. In the future it would be desirable to extend the model with additional light sources for more illumination effects. Introducing further illumination effects, such as texturing and depth of field, can also improve the visual quality of the images.

Our fetoscopic rendering method provides an alternative to existing standard DVR methods for the display of surfaces with local illumination. The method was integrated as *HDlive* imaging tool in the latest generation of ultrasound machines, i.e., Voluson E8 and Voluson E8 Expert. It was positively accepted by clinicians and parents alike. The daily use in several prenatal imaging centers is further evidence in this respect. The novel method was mainly motivated by improving the visual quality of images for parents and it serves this purpose well. The clinical relevance of the method is still not fully clear. This requires extensive user evaluation with clinical experts in the future.

Obstetric ultrasound is a real-time imaging modality and it is characterized by many factors, e.g., process of data acquisition, data characteristics and data resolution. Furthermore, the comfort of patients, the required time for the examination, the skills of the sonographers and a confident communication of findings are very important aspects for the successful application of ultrasound scanning. This thesis provided various insights into the real-world application possibilities of novel visualization methods in clinical environments. Our experience with real-world applications indicates that in many cases simple and effective solutions are far more appreciated by domain experts than more complex algorithms. For the stable application in a clinical environment, it is necessary that the developed system is robust and generates predictable results. If the algorithm requires specification of parameters it is important to optimize their settings beforehand.

Bibliography

- [1] J. G. Abbott and F. Thurstone. Acoustic speckle: Theory and experimental analysis. *Ultrasonic Imaging*, 1(4):303 – 324, 1979.
- [2] K. Abd-Elmoniem, A.-B. Youssef, and Y. Kadah. Real-time speckle reduction and coherence enhancement in ultrasound imaging via nonlinear anisotropic diffusion. *IEEE Transactions on Biomedical Engineering*, 49(9):997–1014, 2002.
- [3] J. Ahrens, K. Brislawn, K. Martin, B. Geveci, C. C. Law, and M. Papka. Large-scale data visualization using parallel data streaming. *IEEE Comput. Graph. Appl.*, 21(4):34–41, 2001.
- [4] T. Arbel, X. Morandi, R. Comeau, and D. Louis Collins. Automatic Non-linear MRI-Ultrasound Registration for the Correction of Intra-operative Brain Deformations. In *Medical Image Computing and Computer-Assisted Intervention*, volume 2208, pages 913–922. Springer Berlin / Heidelberg, 2001.
- [5] K. Baba, K. Satoh, S. Sakamoto, T. Okai, and S. Ishii. Development of an ultrasonic system for the three-dimensional reconstruction of the fetus. *Journal of Perinatal Medicine*, 17:19–24, 1989.
- [6] A. Babinszki, T. Nyari, S. Jordan, A. Nasser, T. Mukherjee, and A. B. Copperman. Three-dimensional measurement of gestational and yolk sac volumes as predictors of pregnancy outcome in the first trimester. *Amer J Perinatol*, 18(04):203–212, 2001.
- [7] J. A. Bærentzen and N. J. Christensen. A technique for volumetric CSG based on morphology. In *International Workshop on Volume Graphics 2001*, pages 71–79, 2001.
- [8] C. L. Bajaj. *Surface fitting using implicit algebraic surface patches*. Society for Industrial and Applied Mathematics, 1992.
- [9] J. Bamber and C. Daft. Adaptive filtering for reduction of speckle in ultrasonic pulse-echo images. *Ultrasonics*, 24(1):41–44, 1986.
- [10] G. Baranoski and A. Krishnaswamy. *Light and Skin Interactions: Simulations for Computer Graphics Applications*. Morgan Kaufmann, 2010.

- [11] U. Behrens and R. Ratering. Adding shadows to a texture-based volume renderer. In *Proceedings of the 1998 IEEE symposium on Volume visualization*, pages 39–46. ACM, 1998.
- [12] B. Benoit and R. Chaoui. Three-dimensional ultrasound with maximal mode rendering: a novel technique for the diagnosis of bilateral or unilateral absence or hypoplasia of nasal bones in second-trimester screening for down syndrome. *Ultrasound in Obstetrics and Gynecology*, 25(1):19–24, 2005.
- [13] M. Bertram, X. Tricoche, and H. Hagen. Adaptive smooth scattered-data approximation for large-scale terrain visualization. In *Proceedings of the symposium on Data visualization 2003*, pages 177–184, 2003.
- [14] I. Bitter, R. Van Uitert, I. Wolf, L. Ibanez, and J.-M. Kuhnigk. Comparison of four freely available frameworks for image processing and visualization that use itk. *IEEE Transactions on Visualization and Computer Graphics*, 13(3):483–493, 2007.
- [15] J. F. Blinn. Models of light reflection for computer synthesized pictures. *SIGGRAPH Comput. Graph.*, 11(2):192–198, July 1977.
- [16] B. Brendel, S. W. A. Rick, M. Stockheim, and H. Ermert. Registration of 3D CT and Ultrasound Datasets of the Spine using Bone Structures. *Computer Aided Surgery*, 7(3):146–155, 2002.
- [17] S. Bruckner and M. E. Gröller. Exploded views for volume data. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1077–84, 2006.
- [18] S. Bruckner and M. E. Gröller. Instant volume visualization using maximum intensity difference accumulation. *Computer Graphics Forum*, 28(3):775–782, 2009.
- [19] M. Burns and A. Finkelstein. Adaptive cutaways for comprehensible rendering of polygonal scenes. *ACM Transactions on Graphics*, 27(5):1–9, 2008.
- [20] M. Burns, M. Haidacher, W. Wein, I. Viola, and E. Groeller. Feature emphasis and contextual cutaways for multimodal medical visualization. In *Eurographics / IEEE VGTC Symposium on Visualization 2007*, pages 275–282, 2007.
- [21] B. Cabral, N. Cam, and J. Foran. Accelerated volume rendering and tomographic reconstruction using texture mapping hardware. In *Proceedings of the 1994 IEEE symposium on Volume visualization, VVS '94*, pages 91–98, New York, NY, USA, 1994. ACM.
- [22] C. J. C. Cash, L. H. Berman, G. M. Treece, A. H. Gee, and R. W. Prager. Two- and three-dimensional ultrasound in the development of a needle-free injection system. *British Journal of Radiology*, 77(915):236–242, 2004.
- [23] E. Cerezo, F. Perez, X. Pueyo, F. J. Seron, and F. X. Sillion. A survey on participating media rendering techniques. *The Visual Computer*, 21:303–328, 2005.

- [24] J. H. Chang, J. T. Yen, and K. K. Shung. High-speed digital scan converter for high-frequency ultrasound sector scanners. *Ultrasonics*, 48(5):444–452, 2008.
- [25] Y. Chen, R. Yin, P. Flynn, and S. Broschat. Aggressive region growing for speckle reduction in ultrasound images. *Pattern Recogn. Lett.*, 24(4-5):677–691, 2003.
- [26] C. D. Correa and K.-L. Ma. The occlusion spectrum for volume classification and visualization. *IEEE Transactions on Visualization and Computer Graphics*, 15(6):1465–72, 2009.
- [27] C. D. Correa and K.-L. Ma. Visibility-driven transfer functions. In *IEEE Pacific Visualization Symposium*, pages 177–184, 2009.
- [28] C. D. Correa and K.-L. Ma. Visibility histograms and visibility-driven transfer functions. *IEEE Transactions on Visualization and Computer Graphics*, 17(2):192–204, 2011.
- [29] C. D. Correa, D. Silver, and M. Chen. Feature aligned volume manipulation for illustration and visualization. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1069–76, 2006.
- [30] F. Cunningham, K. Leveno, S. Bloom, J. Hauth, D. Rouse, and C. Spong. *Williams Obstetrics: 23rd Edition*. Williams Obstetrics. McGraw-Hill, 2009.
- [31] P. Desgranes, K. Engel, and G. Paladini. Gradient-free shading: A new method for realistic interactive volume rendering. In *Proceedings of Vision, Modeling, and Visualization 2005*, pages 209–216, 2005.
- [32] J. Diepstraten, D. Weiskopf, and T. Ertl. Interactive cutaway illustrations. *Computer Graphics Forum*, 22(3):523–532, 2003.
- [33] C. M. Dikkeboom, N. M. Roelfsema, L. N. A. van Adrichem, and J. W. Wladimiroff. The role of three-dimensional ultrasound in visualizing the fetal cranial sutures and fontanels during the second half of pregnancy. *Ultrasound in Obstetrics and Gynecology*, 24(4):412–416, 2004.
- [34] DOTmed. RSNA 2011: GE Healthcare brings ultrasound to the next level, available from <http://www.dotmed.com/news/story/17574>. Accessed: 2012-02-08.
- [35] Q. Duan, S. Homma, and A. Laine. Analysis of 4D Ultrasound for Dynamic Measures of Cardiac Function. In *IEEE Ultrasonics Symposium, 2007*, pages 1492–1495, 2007.
- [36] K. T. Dussik. Über die Möglichkeit, hochfrequente mechanische Schwingungen als diagnostisches Hilfsmittel zu verwerten. *Zeitschrift für die gesamte Neurologie und Psychiatrie*, 174:153–168, 1942.
- [37] V. Dutt and J. Greenleaf. Adaptive speckle reduction filter for log-compressed b-scan images. *IEEE Transactions on Medical Imaging*, 15(6):802–813, 1996.

- [38] N. Dyn. Subdivision schemes in cagd. In *Advances in Numerical Analysis*, pages 36–104. Univ. Press, 1992.
- [39] K. Engel, M. Hadwiger, J. M. Kniss, C. Rezk-Salama, and D. Weiskopf. *Real-Time Volume Graphics*. A K Peters, Limited, 2006.
- [40] R. Farias and C. T. Silva. Out-of-core rendering of large, unstructured grids. *IEEE Comput. Graph. Appl.*, 21(4):42–50, 2001.
- [41] R. Fattal and D. Lischinski. Variational Classification for Visualization of 3D Ultrasound Data. In *IEEE Visualization*, pages 403–410, 2001.
- [42] S. Feiner and D. Seligmann. Cutaways and ghosting: satisfying visibility constraints in dynamic 3D illustrations. *The Visual Computer*, 8(5):292–302, 1992.
- [43] M. K. Feldman, S. Katyal, and M. S. Blackwood. US Artifacts. *Radiographics*, 29(4):1179–1189, 2009.
- [44] R. Franke and G. Nielson. Scattered data interpolation and applications—a tutorial and survey. *Geometric Modelling: Methods and Their Application*, 16:131–160, 1991.
- [45] S. F. Frisken, R. N. Perry, A. P. Rockwood, and T. R. Jones. Adaptively sampled distance fields: a general representation of shape for computer graphics. In *SIGGRAPH Comput. Graph.*, pages 249–254, 2000.
- [46] GE Healthcare. Voluson E8 Expert, available from <http://www3.gehealthcare.com/en/Products/Categories/Ultrasound/Voluson/>. Accessed: 2012-08-27.
- [47] GE Healthcare. Voluson Women’s Health, available from <http://www3.gehealthcare.com/en/Products/Categories/Ultrasound>. Accessed: 2012-07-27.
- [48] A. Gee, R. Prager, G. Treece, and L. Berman. Engineering a freehand 3D ultrasound system. *Pattern Recognition Letters*, 24:757–777, 2003.
- [49] A. Glassner. *Principles of Digital Image Synthesis*. Morgan Kaufmann, 1995.
- [50] J. Haber, F. Zeilfelder, O. Davydov, and H. P. Seidel. Smooth approximation and rendering of large scattered data sets. In *IEEE Visualization*, pages 341–348, 2001.
- [51] M. Hadwiger, A. Kratz, C. Sigg, and K. Bühler. GPU-accelerated deep shadow maps for direct volume rendering. In *GH ’06: Proceedings of the 21st ACM SIGGRAPH/EUROGRAPHICS symposium on Graphics hardware*, pages 49–52, 2006.
- [52] M. Hadwiger, P. Ljung, C. R. Salama, and T. Ropinski. Advanced illumination techniques for GPU-based volume raycasting. In *ACM SIGGRAPH 2009 Courses*, pages 2:1–2:166. ACM, 2009.

- [53] H.-C. Hege, T. Höllerer, and D. Stalling. Volume rendering - mathematical models and algorithmic aspects. Technical report, ZIB, 1993.
- [54] L. Henyey and J. Greenstein. Diffuse radiation in the Galaxy. *Annales d'Astrophysique*, 3:117, 1940.
- [55] F. Hernell, P. Ljung, and A. Ynnerman. Efficient Ambient and Emissive Tissue Illumination using Local Occlusion in Multiresolution Volume Rendering. In *IEEE/EG Volume Graphics*, pages 1–8, 2007.
- [56] F. Hernell, P. Ljung, and A. Ynnerman. Interactive Global Light Propagation in Direct Volume Rendering using Local Piecewise Integration. In *Volume Graphics*, pages 105–112, 2008.
- [57] F. Hernell, P. Ljung, and A. Ynnerman. Local ambient occlusion in direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 16:548–559, 2010.
- [58] D. Hönigmann, J. Ruisz, and C. Haider. Adaptive design of a global opacity transfer function for direct volume rendering of ultrasound data. In *IEEE Visualization*, pages 489–496, 2003.
- [59] D. H. Howry, D. A. Stott, and W. R. Bliss. The ultrasonic visualization of carcinoma of the breast and other soft-tissue structures. *Cancer*, 7(2):354–358, 1954.
- [60] L. Ibanez, W. Schroeder, L. Ng, and J. Cates. *The ITK Software Guide: The Insight Segmentation and Registration Toolkit (version 1.4)*. Kitware, 2003.
- [61] L. Ibarria, P. Lindstrom, J. Rossignac, and A. Szymczak. Out-of-core compression and decompression of large n-dimensional scalar fields. *Computer Graphics Forum*, 22(3):343–348, 2003.
- [62] T. Igarashi, K. Nishino, and S. K. Nayar. The appearance of human skin: A survey. *Foundations and Trends in Computer Graphics and Vision*, 3:1–95, 2007.
- [63] M. Isenburg, Y. Liu, J. Shewchuk, and J. Snoeyink. Streaming computation of delaunay triangulations. *ACM Trans. Graph.*, 25(3):1049–1056, 2006.
- [64] H. W. Jensen, S. R. Marschner, M. Levoy, and P. Hanrahan. A practical model for sub-surface light transport. In *SIGGRAPH Comput. Graph.*, pages 511–518, 2001.
- [65] M. Jones, J. Baerentzen, and M. Sramek. 3d distance fields: a survey of techniques and applications. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):581–599, 2006.
- [66] K. O. Kagan, K. Pintoffl, and M. Hoopmann. First-trimester ultrasound images using HDlive. *Ultrasound in Obstetrics and Gynecology*, 38(5):607–607, 2011.
- [67] A. E. Kaufman, D. Cohen-Or, and R. Yagel. Volume graphics. *IEEE Computer*, 26(7):51–64, 1993.

- [68] J. Kniss, G. Kindlmann, and C. Hansen. Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets. In *IEEE Visualization*, pages 255–262, 2001.
- [69] J. Kniss, G. Kindlmann, and C. Hansen. Multi-dimensional transfer functions for interactive volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 8(3):270–285, 2002.
- [70] J. Kniss, S. Premoze, C. Hansen, and D. Ebert. Interactive translucent volume rendering and procedural modeling. In *IEEE Visualization*, pages 109–116, 2002.
- [71] J. Kniss, S. Premoze, C. Hansen, P. Shirley, and A. McPherson. A model for volume lighting and modeling. *IEEE Transactions on Visualization and Computer Graphics*, 9:150–162, 2003.
- [72] J. Kronander, D. Jonsson, J. Low, P. Ljung, A. Ynnerman, and J. Unger. Efficient visibility encoding for dynamic illumination in direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 18(3):447–462, 2012.
- [73] A. Krüger, C. Tietjen, J. Hintze, B. Preim, I. Hertel, and G. Strauß. Interactive visualization for neck dissection planning. In *IEEE/Eurographics Symposium on Visualization (EuroVis)*, pages 295–302, 2005.
- [74] J. Krüger, J. Schneider, and R. Westermann. ClearView: An interactive context preserving hotspot visualization technique. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):941–8, 2006.
- [75] C. Kubisch, C. Tietjen, and B. Preim. GPU-based smart visibility techniques for tumor surgery planning. *International Journal of Computer Assisted Radiology and Surgery*, 5(6):667–78, 2010.
- [76] J. Kuo, G. Bredthauer, J. Castellucci, and O. von Ramm. Interactive volume rendering of real-time three-dimensional ultrasound images. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 54(2):313–318, 2007.
- [77] A. Kurjak and G. Azumendi. *The Fetus in Three Dimensions: Imaging, Embryology and Fetoscopy*. Taylor & Francis, 2007.
- [78] A. Kurjak, T. Hafner, M. Kos, S. Kupesic, and M. Stanojevic. Three-dimensional sonography in prenatal diagnosis: a luxury or a necessity? *Journal of Perinatal Medicine*, 28:194–109, 2005.
- [79] A. Kurjak, B. Miskovic, W. Andonotopo, M. Stanojevic, G. Azumendi, and H. Vrcic. How useful is 3D and 4D ultrasound in perinatal medicine? *Journal of Perinatal Medicine*, 35(1):10–27, 2007.

- [80] A. Kurjak, M. Stanojevic, G. Azumendi, and M. Carrera. The potential of four-dimensional (4d) ultrasonography in the assessment of fetal awareness. *Journal of Perinatal Medicine*, 33:46–53, 2005.
- [81] C. C. Law, W. J. Schroeder, K. M. Martin, and J. Temkin. A multi-threaded streaming pipeline architecture for large structured data sets. In *IEEE Visualization*, pages 225–232, 1999.
- [82] S. C. Leavitt, B. F. Hunt, and H. G. Larsen. A scan conversion algorithm for displaying ultrasound images. *Hewlett-Packard Journal*, 34:30–34, 1983.
- [83] A. Lee. Four-dimensional ultrasound in prenatal diagnosis: leading edge in imaging technology. *The Ultrasound Review of Obstetrics and Gynecology*, 1(2):144–148, 2001.
- [84] S. Lee, G. Wolberg, and S. Shin. Scattered data interpolation with multilevel b-splines. *IEEE Transactions on Visualization and Computer Graphics*, 3(3):228–244, 1997.
- [85] M. Levoy. Display of surfaces from volume data. *IEEE Comput. Graph. Appl.*, 8:29–37, 1988.
- [86] E. Light, J. Angle, and S. Smith. Real-time 3-d ultrasound guidance of interventional devices. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 55(9):2066–2078, 2008.
- [87] E. Light, R. Davidsen, J. Fiering, T. Hruschka, and S. Smith. Progress in two-dimensional arrays for real-time volumetric imaging. *Ultrasonic Imaging*, 20(1):1–15, 1998.
- [88] F. Lindemann and T. Ropinski. Advanced light material interaction for direct volume rendering. In *Volume Graphics*, pages 101–108, 2010.
- [89] M. Livingstone. *Vision and art: the biology of seeing*. Harry N. Abrams, New York, 2002.
- [90] S. K. Lodha and R. Franke. Scattered data techniques for surfaces. In *IEEE Conference on Scientific Visualization*, page 181, 1997.
- [91] London Ultrasound Centre. available from <http://www.thelondonultrasoundcentre.co.uk/>. Accessed: 2012-08-21.
- [92] T. Loupas, W. McDicken, and P. Allan. An adaptive weighted median filter for speckle suppression in medical ultrasonic images. *IEEE Transactions on Circuits and Systems*, 36(1):129–135, 1989.
- [93] G. Ludwig and F. W. Struthers. *Considerations underlying the use of ultrasound to detect gallstones and foreign bodies in tissue*. Number Nr. 4. Naval Medical Research Institute, 1949.
- [94] W. Mak, Y. Wu, and M. Chan. Visibility-aware direct volume rendering. *Journal of Computer Science and Technology*, 26:217–228, 2011.

- [95] M. M. Malik, T. Möller, and M. E. Gröller. Feature peeling. In *Proceedings of Graphics Interface 2007*, pages 273–280, 2007.
- [96] N. Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995.
- [97] N. Max and M. Chen. Local and global illumination in the volume rendering integral. In *Scientific Visualization: Advanced Concepts*, pages 259–274, 2010.
- [98] J. A. McGrath and J. Uitto. *Rook's Textbook of Dermatology, Eighth Edition*, chapter Anatomy and Organization of Human Skin, pages 1–53. Wiley-Blackwell, 2010.
- [99] S. Meairs, J. Beyer, and M. Hennerici. Reconstruction and visualization of irregularly sampled three- and four-dimensional ultrasound data for cerebrovascular applications. *Ultrasound in Medicine & Biology*, 26(2):263–272, 2000.
- [100] L. Mercier, T. Langø, F. Lindseth, and D. L. Collins. A review of calibration techniques for freehand 3-d ultrasound systems. *Ultrasound in Medicine & Biology*, 31(4):449 – 471, 2005.
- [101] E. Merz. Surface Reconstruction of a Fetus (28 + 2 GW) Using HDlive Technology. *Ultraschall in Med*, 33(03):211–211, 2012.
- [102] E. Merz, D. Miric-Tesanic, and C. Welter. Value of the electronic scalpel (cut mode) in the evaluation of the fetal face. *Ultrasound in Obstetrics and Gynecology*, 16(6):564–568, 2000.
- [103] C. Montani and R. Scopigno. Rendering volumetric data using STICKS representation scheme. *SIGGRAPH Comput. Graph.*, 24(5):87–93, 1990.
- [104] K. D. Moreland. *Fast High Accuracy Volume Rendering*. PhD thesis, The University of New Mexico, 2004.
- [105] T. R. Nelson and D. H. Pretorius. Three-dimensional ultrasound imaging. *Ultrasound in Medicine & Biology*, 24(9):1243–1270, 1998.
- [106] T. R. Nelson, D. H. Pretorius, A. Hull, M. Riccabona, M. S. Sklansky, and G. James. Sources and impact of artifacts on clinical three-dimensional ultrasound imaging. *Ultrasound in Obstetrics and Gynecology*, 16(4):374–383, 2000.
- [107] G. M. Nielson. Normalized implicit eigenvector least squares operators for noisy scattered data: radial basis functions. *Computing*, 86(2-3):199–212, 2009.
- [108] L. Nilsson and L. Hamberger. *A Child Is Born*. Doubleday, 2004.
- [109] J. Noble and D. Boukerroui. Ultrasound image segmentation: a survey. *IEEE Transactions on Medical Imaging*, 25(8):987–1010, 2006.

- [110] B. Petersch and D. Hönigmann. Blood flow in its context: Combining 3d b-mode and color doppler ultrasonic data. *IEEE Transactions on Visualization and Computer Graphics*, 13:748–757, 2007.
- [111] H. Pottmann. Approximation algorithms for developable surfaces. *Computer Aided Geometric Design*, 16(6):539–556, 1999.
- [112] M. J. D. Powell. *The Theory of Radial Basis Function Approximation*. Oxford University Press, USA, 1992.
- [113] R. Prager, U. Ijaz, A. H. Gee, and G. Treece. Three-dimensional ultrasound imaging. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 224(2):193–223, 2010.
- [114] R. W. Prager, A. H. Gee, G. M. Treece, C. J. Cash, and L. H. Berman. Sensorless freehand 3-d ultrasound using regression of the echo intensity. *Ultrasound in Medicine & Biology*, 29(3):437–446, 2003.
- [115] Prenatal Imaging Centers - Kansas City. Diagnostic 3D/4D Fetal Sonography, available from <http://prenatalimaging.com/index.html>. Accessed: 2012-08-21.
- [116] Qmed. GE Healthcare launches its next generation womens health imaging, available from <http://www.qmed.com/news/ge-healthcare-launches-its-next-generation-womens-health-imaging>. Accessed: 2012-02-08.
- [117] R. N. Rankin, A. Fenster, B. D. Donal, L. P. Munk, F. M. Levin, and D. A. Vellet. Three-dimensional sonographic reconstruction: techniques and diagnostic applications. *American Journal of Roentgenology*, 161(4):695–702, 1993.
- [118] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda. Photographic tone reproduction for digital images. *ACM Transactions on Graphics*, 21(3):267–276, July 2002.
- [119] C. Rezk-Salama. Gpu-based monte-carlo volume raycasting. In *Pacific Conference on Computer Graphics and Applications*, pages 411–414, 2007.
- [120] C. Rezk-Salama and A. Kolb. Opacity Peeling for Direct Volume Rendering. *Computer Graphics Forum*, 25(3):597–606, 2006.
- [121] T. Ritschel. Fast GPU-based Visibility Computation for Natural Illumination of Volume Data Sets. In *Short Paper Proceedings of Eurographics 2007*, pages 17–20, 2007.
- [122] T. Ropinski, C. Döring, and C. Rezk-Salama. Interactive volumetric lighting simulating scattering and shadowing. In *IEEE Pacific Visualization Symposium*, pages 169–176, 2010.
- [123] T. Ropinski, J. Kasten, and K. H. Hinrichs. Efficient shadows for gpu-based volume raycasting. In *Proceedings of the 16th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG 2008)*, pages 17–24, 2008.

- [124] T. Ropinski, J. Meyer-Spradow, S. Diepenbrock, J. Mensmann, and K. H. Hinrichs. Interactive volume rendering with dynamic ambient occlusion and color bleeding. *Computer Graphics Forum*, 27(2):567–576, 2008.
- [125] G. Sakas and S. Walter. Extracting surfaces from fuzzy 3d-ultrasound data. In *SIGGRAPH Comput. Graph.*, pages 465–474. ACM, 1995.
- [126] San Francisco Perinatal Associates, Inc. Obstetrical Ultrasound, available from <http://www.sfperinatal.com/our-services/prenatal-diagnosis/obstetrical-ultrasound>. Accessed: 2012-08-21.
- [127] P. Schlegel and R. Pajarola. Layered volume splatting. In *Proceedings of the 5th International Symposium on Advances in Visual Computing: Part II*, pages 1–12, 2009.
- [128] M. Schott, V. Pegoraro, C. D. Hansen, K. Boulanger, and K. Bouatouch. A directional occlusion shading model for interactive direct volume rendering. *Computer Graphics Forum*, 28(3):855–862, 2009.
- [129] W. Schroeder, K. Martin, and B. Lorensen. *The visualization toolkit, third ed.* Kitware, 2004.
- [130] L. Schumaker. *Fitting Surfaces to Scattered Data*. New York, Academic Press, 1976.
- [131] D. Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the 1968 ACM National Conference*, pages 517–524, 1968.
- [132] K. Shoemake. ARCBALL: A User Interface for Specifying Three-Dimensional Orientation Using a Mouse. In *Proceedings of Graphics Interface '92*, pages 151–156, Vancouver, Canada, 1992.
- [133] S. W. Smith, K. Chu, S. F. Idriss, N. M. Ivancevich, E. D. Light, and P. D. Wolf. Feasibility study: Real-time 3-d ultrasound imaging of the brain. *Ultrasound in Medicine & Biology*, 30(10):1365–1371, 2004.
- [134] O. V. Solberg, F. Lindseth, H. Torp, R. E. Blake, and T. A. N. Hernes. Freehand 3d ultrasound reconstruction algorithms: A review. *Ultrasound in Medicine & Biology*, 33(7):991 – 1009, 2007.
- [135] V. Soltészová, D. Patel, S. Bruckner, and I. Viola. A multidirectional occlusion shading model for direct volume rendering. *Computer Graphics Forum*, 29(3):883–891, 2010.
- [136] M. Sramek and A. E. Kaufman. Alias-free voxelization of geometric objects. *IEEE Transactions on Visualization and Computer Graphics*, 5(3):251–267, 1999.
- [137] J. Stam. Multiple scattering as a diffusion process. In *Eurographics Rendering Workshop*, pages 41–50, 1995.
- [138] S. Standring and H. Gray. *Gray's Anatomy: The Anatomical Basis of Clinical Practice*. Churchill Livingstone/Elsevier, 2008.

- [139] R. Stephens. A survey of stream processing. *Acta Informatica*, 34:491–541, 1997.
- [140] A. J. Stewart. Vicinity shading for enhanced perception of volumetric data. In *IEEE Visualization*, pages 355–362, 2003.
- [141] M. Straka, M. Cervenansky, A. La Cruz, A. Kochl, M. Sramek, E. Gröller, and D. Fleischmann. The VesselGlyph: Focus & Context Visualization in CT-Angiography. In *IEEE Visualization*, pages 385–392, 2004.
- [142] E. Sundén, A. Ynnerman, and T. Ropinski. Image Plane Sweep Volume Illumination. *IEEE Visualization*, 17(12):2125–2134, 2011.
- [143] T. Szabo. *Diagnostic Ultrasound Imaging: Inside Out*. Biomedical Engineering. Elsevier Academic Press, 2004.
- [144] A. Tikhonova, C. Correa, and K. Ma. An exploratory technique for coherent visualization of time-varying volume data. In *Computer Graphics Forum*, pages 783–792, 2010.
- [145] J. W. Trobaugh, D. J. Trobaugh, and W. D. Richard. Three-dimensional imaging with stereotactic ultrasonography. *Computerized Medical Imaging and Graphics*, 18(5):315 – 323, 1994.
- [146] T. A. Tuthill, J. F. Krücker, J. B. Fowlkes, and P. L. Carson. Automated three-dimensional US frame positioning computed from elevational speckle decorrelation. *Radiology*, 209(2):575–582, 1998.
- [147] M. Van Gemert, A. Welch, W. Star, M. Motamedi, and W.-F. Cheong. Tissue optics for a slab geometry in the diffusion approximation. *Lasers in Medical Science*, 2:295–302, 1987.
- [148] G. Varga, G. Azumendi, and A. Kurjak. *3D and 4D sonography: How to use it properly - technical advice*, chapter 1, pages 1–13. Taylor & Francis, 2007.
- [149] I. Viola and M. Gröller. Smart visibility in visualization. In *Proc. of EG Workshop on Computational Aesthetics in Graphics, Visualization and Imaging*, pages 209–216, 2005.
- [150] I. Viola, A. Kanitsar, and M. E. Gröller. Importance-driven feature enhancement in volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):408–18, 2005.
- [151] R. Wagner, M. Insana, and S. Smith. Fundamental correlation lengths of coherent speckle in medical ultrasonic images. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 35(1):34–44, 1988.
- [152] L. Wanger. The effect of shadow quality on the perception of spatial relationships in computer generated imagery. In *Proceedings of the 1992 symposium on Interactive 3D graphics*, pages 39–42, 1992.

- [153] Wikia. Fetus, available from <http://psychology.wikia.com/wiki/fetus>. Accessed: 2012-08-27.
- [154] Wikipedia. Human skin, available from http://en.wikipedia.org/wiki/human_skin. Accessed: 2012-15-09.
- [155] Woo, Joseph. Obstetric ultrasound, available from <http://www.ob-ultrasound.net/>. Accessed: 2012-08-02.
- [156] Y. Wu and H. Qu. Interactive transfer function design based on editing direct volume rendered images. *IEEE Transactions on Visualization and Computer Graphics*, 13(5):1027–40, 2007.
- [157] A. N. Yaroslavskaya, S. R. Utz, and S. N. Tatarintsev. Angular scattering properties of human epidermal skin layers. In *Proceedings of SPIE*, volume 2100, pages 38–41, 1994.
- [158] J. Yen and S. Smith. Real-time rectilinear 3-d ultrasound using receive mode multiplexing. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 51(2):216–226, 2004.
- [159] J. T. Yen and S. W. Smith. Real-time rectilinear volumetric imaging using a periodic array. *Ultrasound in Medicine & Biology*, 28(7):923–931, 2002.
- [160] X. Yuan, M. X. Nguyen, B. Chen, and D. H. Porter. HDR VolVis: High Dynamic Range Volume Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):433–445, 2006.
- [161] H. Zhang and F. Banovac. Freehand 3D ultrasound calibration using an electromagnetically tracked needle. *Proceedings of SPIE*, 6141:775–783, 2006.
- [162] M. Zwicker. *Continuous Reconstruction, Rendering, and Editing of Point-Sampled Surfaces*. PhD thesis, ETH Zurich, 2003.

Curriculum Vitae

Contact Information

Name	Andrej Varchola
Address	Franzensbrückenstr. 13/22, 1020 Vienna, Austria
E-mail	av@cg.tuwien.ac.at

Personal Details

Date of Birth	July 8 th , 1980
Place of Birth	Prešov, Slovakia
Citizenship	Slovak
Sex	Male
Languages	Slovak (native), English, German, Russian

Education

- 2008 – Vienna University of Technology, Vienna, Austria
Doctoral studies
Faculty: Faculty of Informatics
Institute: Institute of Computer Graphics and Algorithms
Program: Technical Sciences
Major: Visualization
- 2004 – 2006 Slovak University of Technology in Bratislava, Bratislava, Slovakia
Master studies
Faculty: Faculty of Electrical Engineering and Information Technology
Institute: Institute of Telecommunications
Program: Telecommunications
Major: Digital Signal Processing
- 1998 – 2002 Slovak University of Technology in Bratislava, Bratislava, Slovakia
Bachelor studies
Faculty: Faculty of Electrical Engineering and Information Technology
Program: Informatics
- 1994 – 1998 Gymnázium v Sabinove, Sabinov, Slovakia
Secondary education

Employment History

- 2009 – Institute of Computer Graphics and Algorithms, Vienna University of Technology, Vienna, Austria
Research assistant
Natural Fetoscopic Rendering of Ultrasound Data.
- 2006 – 2009 Commission for Scientific Visualization, Austrian Academy of Sciences, Vienna, Austria
Research assistant
Visualization of CT Angiography Data.

Publications

G. Mistelbauer, A. Varchola, H. Bouzari, J. Starinský, A. Köchl, R. Scherthaner, D. Fleischmann, and M. E. Gröller, and M. Šrámek. Centerline Reformations of Complex Vascular Structures, *IEEE Pacific Visualization Symposium*, 233-240, Feb. 2012

A. Varchola, A. Vasko, V. Solcany, L. I. Dimitrov, and M. Šrámek. Processing of Volumetric Data by Slice- and Process-based Streaming, *Proceedings of the 5th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa*, 101-110, October. 2007.