D I S S E R T A T I O N

Virtual Endoscopy for Preoperative Planning and Training of Endonasal Transsphenoidal Pituitary Surgery

ausgeführt zum Zwecke der Erlangung des akademischen Grades
eines Doktors der technischen Wissenschaften

unter der Leitung von

Ao.Univ.Prof. Dipl.-Ing. Dr.techn. Eduard Gröller,
Institut E186 für Computergraphik und Algorithmen,

und

Dipl.-Math. Dr.techn. Katja Bühler,
VRVis Zentrum für Virtual Reality und Visualisierung,

eingereicht an der Technischen Universität Wien,
Fakultät für Informatik,

von

Dipl.-Ing. André Neubauer,
Matrikelnummer 9625734,
Bahnstraße 52/2/17,
A-2345 Brunn am Gebirge

Wien, im Mai 2005

Abstract

Virtual endoscopy is the navigation of a virtual camera through anatomy, computationally reconstructed from radiological data. Virtual endoscopy mimics physical minimally invasive surgical activity and is used for diagnosis (e.g., the detection of colon polyps), planning of endoscopic interventions, postoperative assessment of surgical success and training for inexperienced endoscopists.

This thesis introduces STEPS, a virtual endoscopy system designed as a planning and training tool for endonasal transsphenoidal pituitary surgery, a method used to minimally invasively remove tumors of the pituitary gland. A rigid endoscope is inserted into the nose and advanced through intracranial cavities towards the bony wall covering the pituitary gland. This bone is then opened and the tumor is removed.

STEPS reconstructs the boundaries of the investigated cavities primarily using iso-surfacing in original CT data. Presegmented objects of interest can be added to the scene and displayed behind the semi-transparent isosurface. These objects (e.g., the tumor, the pituitary gland and important blood vessels) provide an augmented picture of patient anatomy to guide the surgeon, aid in planning the endoscopic approach, and help the user find an ideal site for actual surgical activity. Visual information can be further enhanced by display of rigid structures beyond the isosurface. The user can freely decide upon colors and lighting conditions.

All rendering techniques applied by STEPS are completely CPU-based, ensuring a high amount of flexibility and hardware-independence. Nevertheless, rendering is efficient resulting in smooth interaction.

STEPS allows free navigation through the nasal and paranasal anatomy, but can also be used to simulate the movement parameters of the rigid endoscopes. This includes simulation of surgical instruments and haptic feedback. Furthermore, STEPS allows the simulation of angled endoscopes and the simulation of barrel distortion as exhibited by real endoscope optics.

This thesis gives an overview about existing techniques and applications of virtual endoscopy, introduces the field of application, and, in detail, describes STEPS, the required preprocessing, the rendering techniques and the user interface.

Kurzfassung

Virtuelle Endoskopie ist die Navigation einer virtuellen Kamera durch mittels Computergraphik aus radiologischen Daten rekonstruierte Anatomie. Virtuelle Endoskopie simuliert echte minimalinvasive chirurgische Eingriffe und wird zu Diagnosezwecken (z.B. zur Erkennung von Darmpolypen) ebenso herangezogen wie zur Planung von endoskopischen Eingriffen und zur postoperativen Begutachtung. Weiters findet sie Verwendung als Übungswerkzeug in der Ausbildung von Ärzten.

Diese Dissertation stellt STEPS vor, ein Softwaresystem, das als Werkzeug zur Planung und zum Training von endonasaler transsphenoidaler Hypophysenchirurgie konzipiert ist. Endonasale transsphenoidale Hypophysenchirurgie wird zur schonenden, minimalinvasiven Entfernung von Hypophysentumoren herangezogen: Ein starres Endoskop wird in die Nase des Patienten eingeführt und durch die natürlichen Luftwege bis in die Keilbeinhöhle (sinus sphenoidalis) gesteuert. Anschließend wird der Sellaboden, die knöcherne Wand, die die Keilbeinhöhle von der Hypophyse trennt, geöffnet. Durch das entstandene Loch kann der Tumor entfernt werden.

STEPS rekonstruiert die Wände der vom virtuellen Endoskop durchquerten Hohlräume als Iso-Oberfläche. Vorsegmentierte interessante Objekte können der Szenerie beigefügt und hinter der semi-transparenten Iso-Oberfläche dargestellt werden. Diese Objekte (z.B. der Tumor, die Hypophyse, sowie wichtige Blutgefäße) helfen dem Benutzer, sich ein umfassenderes Bild der Anatomie des Patienten zu machen, fungieren als Orientierungshilfen und helfen bei der Planung der tatsächlichen Operation, zum Beispiel bei der Entscheidung, an welcher Stelle die Öffnung des Sellabodens durchgeführt werden soll.

Knöcherne Strukturen hinter der Iso-Oberfläche können mittels diverser Darstellungsmethoden kommuniziert werden. Der Benutzer kann Farben und Lichtverhältnisse frei wählen.

Alle von STEPS angewandten Visualisierungsalgorithmen werden ausschließlich auf der CPU ausgeführt, um größtmögliche Flexibilität und Hardware-Unabhängigkeit zu gewährleisten. Durch die Anwendung optimierter Visualisierungsalgorithmen läuft das Programm dennoch flüssig und vollkommen interaktiv.

Der Benutzer von STEPS kann entweder frei durch den rekonstruierten Patienten navigieren, oder die tatsächliche Operation simulieren. In diesem Simulationsmodus ist die Bewegungsfreiheit eingeschränkt, da die Bewegungsparameter eines starren Endoskops in den engen nasalen und transnasalen Hohlräumen nachempfunden werden. Durch die Verwendung eines Force-Feedback Joysticks werden Kollisionen des virtuellen Endoskops mit der dargestellten Oberfläche haptisch fühlbar. Mittels virtueller chirurgischer Instrumente kann beispielsweise die Öffnung des Sellabodens simuliert werden. Weiters unterstützt STEPS die Simulation von Winkeloptiken und Linsenverzerrungen.

Diese Dissertation gibt einen Überblick über existierende Methoden und Anwendungen der virtuellen Endoskopie, erklärt das medizinische Anwendungsgebiet und beschreibt ausführlich das Softwaresystem STEPS.

Contents

Chapter 1

Introduction

The pituitary gland is the central gland for the hormonal metabolism of the whole human body. Pituitary adenomas are benign slow-growing tumors that arise within the pituitary gland. If left untreated, these tumors can cause a variety of symptoms, from mild headaches and progressive visual loss to death due to electrolyte imbalance.

Minimally invasive endoscopic procedures are constantly gaining importance in modern medicine [Bar03b]. Endonasal transsphenoidal pituitary surgery is a relatively new endoscopic procedure used for the removal of pituitary tumors: A rigid endoscope and surgical instruments are inserted into the patient's nose and advanced through the natural nasal airway into the *sphenoid sinus*, a cavity separated from the pituitary gland only by a thin bony wall, the *sellar floor*. The sellar floor is surgically opened and the tumor is divided from the surrounding tissue and removed through the patient's nose.
The procedure is associated with some risk: The *internal carotid arteries* pass just laterally, and the optic nerves superiorly to the pituitary gland behind the sellar floor. Damage to these structures and the pituitary gland itself must be prevented. They are, however, invisible to the surgeon who can only gather hints about their locations through preoperative analysis of radiological images or intraoperatively, if the canals containing the vessels and nerves bulge the wall of the sphenoid sinus.
The surgeon therefore must be skilled, experienced and well-prepared to conduct such an intervention ensuring the required degree of safety.

Virtual endoscopy can be used as a tool for preoperative planning, intraoperative support and training. Virtual endoscopy uses techniques from visualization and virtual reality to generate a virtual reconstruction of the anatomy of interest, through which the user can navigate on the computer screen.

This thesis introduces *STEPS*. STEPS (Simulation of Transsphenoidal Endonasal Pituitary Surgery) is a virtual endoscopy system designed to aid neurosurgeons as a training and planning device for endonasal transsphenoidal pituitary surgery.
STEPS allows the user to obtain simulated endoscopic images of anatomy of interest. According to the user's needs, the system depicts either only the surfaces of the investigated cavities, or also bony structures beyond the surfaces. Background objects of interest can be segmented and displayed. Foreground and background are rendered using two different
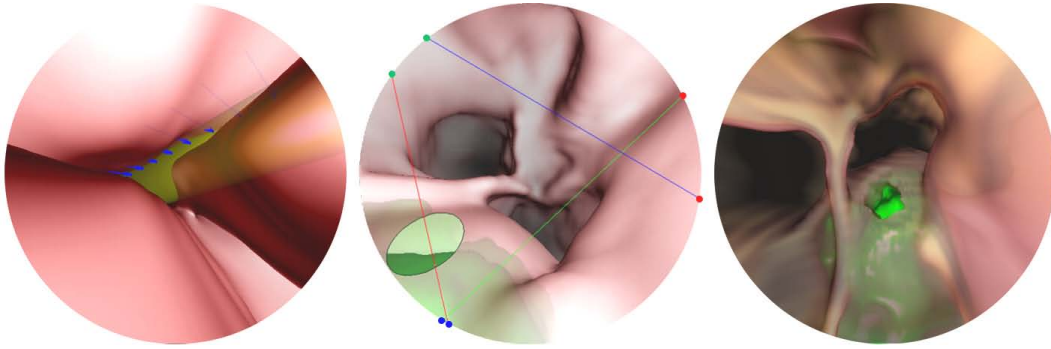
Figure 1.1: Screen shots from STEPS: Left: The endoscope (yellow) inside a narrow nasal passage; Center: Lateral view inside the sphenoid sinus, the tumor (green) is rendered in the background; the red, green and blue lines are navigation aids; the elliptical mark is the preview of the virtual bone punch; Right: The sphenoid sinus, the tumor and bony structures are rendered in the background

CPU-based first-hit ray casting techniques.

The user can choose whether light intensity should decrease with depth, as in reality, and whether the optics of the virtual endoscope should exhibit realistic barrel distortion.

The virtual endoscope can be maneuvered either completely freely, therefore allowing to view structures invisible in reality, or, for training purposes, in a way such that real rigid endoscopy is mimicked. Virtual surgical instruments are provided, allowing simulation of actual surgical activity.

The system as described in this thesis has been implemented in Java and requires no dedicated hardware. It is completely CPU-based and performs well on today's standard PCs.

Chapter 2 provides a concise overview of medical terms encountered throughout the thesis. Chapter 3 presents the field of application of STEPS in more detail. The medical background and anatomy is studied and the benefits of virtual endoscopy for transsphenoidal pituitary surgery are outlined. Chapter 4 introduces virtual endoscopy. After outlining the advantages and disadvantages of virtual endoscopy, existing systems are presented as well as techniques for rendering and navigation. Chapter 5 describes the required data preprocessing and the tools and techniques offered by STEPS. The virtual endoscopy module is the core part of STEPS. It is described in Chapters 6 and 7. Chapter 6 presents rendering techniques. Chapter 7 describes the user-interface of the application, introduces navigation and perception aids and explains how STEPS can be used to simulate the transsphenoidal pituitary surgery. Chapter 8 points out benefits and problems experienced by neurosurgeons working with the application. Finally, chapter 9 illustrates the basic steps of a complete virtual transsphenoidal surgical procedure.

Chapter 2

Medical Terms

The following medical terms are needed for full understanding of the upcoming chapters:

- Adenoma: A benign tumor arising from *glandular* tissue which may cause it to produce abnormal amounts of hormones, e.g., an adenoma of the *pituitary gland*.

- Amenorrhoea-Galactorrhoea: A syndrome with abnormal *lactation* from *endocrinological* causes or from a *pituitary* tumor.

- Anterior: The front, as opposed to the *posterior*. The anterior surface of the heart is toward the breast bone.

- Arachnoid Cyst: A fluid-containing cyst, most often found near the *pituitary gland*. Arachnoid cysts often are an incidental finding on imaging, and patients usually are asymptomatic even if the cyst is quite large.

- Cavernous Sinus: Either of a pair of large venous sinuses situated in a groove at the side of the body of the sphenoid bone, bordering to the *sphenoid sinus*. Each *internal carotid artery* passes through a cavernous sinus.

- Cerebral: Concerning the brain.

- Cerebrospinal Fluid (CSF): A clear fluid surrounding the brain and spinal cord.

- Choana: The posterior aperture of the nasal cavity connecting it to the digestive tract.

- Colon: The large intestine.

- Colon Polyp: A fleshy growth on the inside of the *colon*. Polyps give rise to colon cancer.

- Concha: A *nasal turbinate*.

- Craniotomy: A temporary surgical opening of part of the *cranium*.

- Cranium: The skull.

- Cushing's disease: A syndrome including the over-production of cortisol in the adrenal *gland*. The disease can be caused by a *pituitary adenoma*.

- Dura Mater: A membranous covering of the brain and spinal cord.

- Endocrinology: The endocrine system is the network of *glands*. Endocrinology is the branch of medicine dealing with disorders of the endocrine system and its specific secretions called *hormones*.

- Endonasal: Inside the nose, or through the nose.

- Fluoroscopy: A special type of x-ray used to project live images onto a monitor.

- Gigantism/Acromegaly: A syndrome causing enlargement of the head, face, hands, feet, and upper torso. It is due to the excess production of growth *hormone* by the *pituitary gland*.

- Gland: Any of various organs that synthesize substances (e.g., *hormones*) needed by the body and release them into the bloodstream.

- Hormones: Chemical messengers mostly formed in *endocrine glands* and carried via the bloodstream to their target organs where they trigger or regulate functional activity.

- Hypertension: High blood pressure.

- Hypothalamus: The part of the brain next to and communicating with the pituitary gland.

- Internal Carotid Arteries: The internal carotid arteries pass up the neck and enter the skull, each at first lying in a *cavernous sinus* then dividing into its terminal branches, the anterior and middle cerebral artery. They supply large portions of the brain.

- Intersinus Septum: The *septum* subdividing the *sphenoid sinus* in two parts.

- Intracranial: Inside the *cranium*.

- Kidney Stone: Solid piece of mineral and crystalline material formed within the kidney or urinary tract. Also called *renal calculus*. Kidney stones are a common cause of blood in the urine and pain in the abdomen, flank, or groin. Kidney stones occur in 1 in 20 people at some time in their life.

- Lactation: The period of the secretion of milk after giving birth.

- Laparoscopy: Endoscopy in the abdomen. The laparoscope is inserted through a small incision to view, for instance, the peritoneal cavity, the ovaries or the uterus.

- Larynx: The voice box.

- Lateral: In anatomy, the side of the body or a body part that is farther from the middle or center of the body. Typically, lateral refers to the outer side of the body part, but it is also used to refer to the side of a body part. For example, when referring to the knee, lateral refers to the side of the knee farthest from the opposite knee.

- Lesion: Any pathological or traumatic discontinuity of tissue or loss of function of a part of anatomy.

- Macroadenoma: An *adenoma* over 10 mm in diameter, large enough to be easily visualized by usual radiological techniques.

- Microadenoma: An *adenoma* not exceeding 10 mm in diameter.

- Maxillary Sinuses: The largest *paranasal sinus*. It is located lateral to the *nasal cavity*, intimately related to the upper teeth, tear duct, and the *orbit*.

- Meningitis: An infection which causes inflammation of the membranes covering the brain and spinal cord. Neurosurgical complications may lead to meningitis postoperatively.

- Morbidity: The incidence of a disease in a population.

- Nasal Cavity: The respiratory passages on either side of the nasal septum, starting at the nostrils, lined with mucosa.

- Nasal Septum: The dividing wall that runs down the middle of the nose so that there are normally two sides to the nose, each ending in a nostril.

- Nasal Turbinates: Small, shelf-like, bony structures forming the inner walls of both the left and the right part of the nasal cavity. Also called *conchae*. Their task is to filter and humidify the incoming air. On each side of the nasal septum there is an inferior, a middle and a superior nasal turbinate.

- Onodi Cell: An air-filled cell next to the sphenoid sinus, associated with the optic nerve.

- Operating Microscope: A binocular microscope used in (micro)surgery to provide a clear view of small and inaccessible parts of the body.

- Optic Nerve: The nerve responsible for communication between the eye and the brain.

- Orbit: The bone around the eye. Its medial wall is connected to the nasal bone.

- Paranasal Sinuses: Four pairs of *sinuses* adjacent to the *nasal cavity*.

- Pituitary (Gland): The "Master"-*gland* of the human body, responsible for the *hormone* metabolism, see Section 3.1.

- Pneumatization: The formation of air-filled cells or cavities in tissue. The *sphenoid sinus*, for instance, is developed through pneumatization of the *sphenoid bone*.

- Posterior: The back or behind, as opposed to the *anterior*.

- Renal Calculus: Condition related to the presence of stones in the urinary tract. Also called *kidney stone*

- Rhinotomy: A surgical procedure in which an incision is made in the nose to allow access to the *cranium*.

- Rongeur: A forceps-like instrument for cutting tough tissue, particularly bone ("bone punch").

- Sella Turcica: The bony cavity containing the *pituitary gland*.

- Sellar Floor: The floor of the *sella turcica*. A thin bony structure separating the *sphenoid sinus* and the *sella turcica*.

- Septum: A dividing wall, usually bony or cartilaginous.

- Sinus: Any of various air-filled cavities in the bones of the skull.

- Situs: The position of an organ or part of the body, especially the normal position.

- Spheno-Ethmoid Recess: The anterior wall of the *sphenoid sinus*.

- Sphenoid bone: A prominent, irregular, wedge-shaped bone at the base of the skull. During childhood, it is usually *pneumatized* and the *sphenoid sinus* is developed.

- Sphenoid Ostium: A small opening connecting the *sphenoid sinus* with the *nasal cavity*.

- Sphenoidotomy: The process of enlarging the *sphenoid ostium* to allow endoscopic access to the *sphenoid sinus*.

- Sphenoid Sinus: The most posterior *sinus*, located next to the *sella turcica* containing the *pituitary gland*.

- Stenosis: A local narrowing of a luminous anatomical structure (e.g., an artery, the *ureter*).

- Sublabial: Underneath the upper lip.

- Trachea: Air tube extending from the *larynx* into the chest where it divides into the left and right main bronchi.

- Transcranial: Through the *cranium*.

- Transsphenoidal: Traversing the *sphenoid sinus*.

- Ureter: The passageway that takes the urine from the kidneys to the bladder.

- Urethra: The passageway that takes the urine from the urinary bladder to the outside.

Chapter 3

Medical Background

This chapter provides the motivation for the development of STEPS. Important anatomical structures will be introduced, and the medical field of application will be explained. After a short history of transsphenoidal pituitary surgery and a quick introduction into radiological imaging, the currently widely applied endoscopic technique will be described. Finally, the applicability of virtual endoscopy for transsphenoidal pituitary surgery will be discussed.

3.1   The Pituitary Gland

Located centrally at the base of the brain (see Figure 3.1), inside a bony cavity called *sella turcica*, the pituitary is the central gland for the hormone metabolism of the whole body. The gland is governed by the hypothalamus and attached to it by nerve fibers. It consists of two parts, the *anterior lobe* and the *posterior lobe*. Each part produces a set of hormones: The anterior lobe produces the growth hormone (also called *somatotropin*), prolactin (to stimulate milk production after giving birth), ACTH (*adrenocorticotropic hormone*, to stimulate the adrenal glands), TSH (*thyroid-stimulating hormone*, to stimulate the thyroid gland), FSH (*follicle-stimulating hormone*, to stimulate the ovaries or testes) and LH (*luteinizing hormone*, to stimulate the ovaries or testes). The posterior lobe produces ADH (*antidiuretic hormone*, to increase absorption of water into the blood by the kidneys) and oxytocin (to contract the uterus during childbirth and stimulate milk production).

3.2   Pituitary Adenomas

Pituitary adenomas are benign slow-growing tumors that arise within the pituitary gland. Comprising around 15 percent of all primary intracranial tumors [STO+03], they are a common and important neurosurgical problem: If left untreated, these tumors can cause a variety of symptoms, from mild headaches and progressive visual loss due to the mass effect of the tumor to death due to electrolyte imbalance. Most importantly, pituitary adenomas can lead to pituitary dysfunction which causes either a lack of hormones, yielding, e.g., fatigue and sexual dysfunction, or an excess of certain hormones with Cushing's disease, gigantism/acromegaly, and amenorrhoea-galactorrhoea as clinical syndromes.
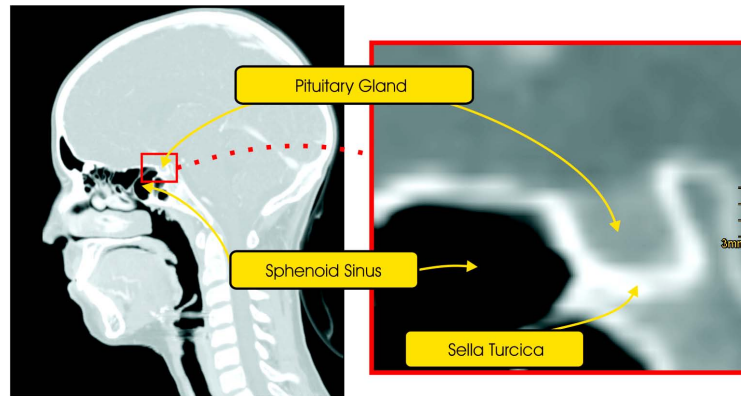
Figure 3.1: The pituitary gland is located inside a cavity called *Sella Turcica*

Pituitary adenomas are very frequent. Hall et al. [HLD$^+$94] conducted MRI examinations of patients asymptomatic for pituitary disease. Pituitary adenoma were found in about 10 percent of the probands.

3.3 A Brief History of Transsphenoidal Pituitary Surgery

The way pituitary adenomas are treated has been subject to a rapid development during the twentieth century and in recent years [For03, GP01]. Until the end of the 19th century, pituitary adenomas were accessed through craniotomies. This often led to complications and bore the danger of severe postoperative damages. The first attempt at transsphenoidal surgery was undertaken in 1907 by Schloffer in Innsbruck. To lay the path for the operation, he performed a lateral rhinotomy, turning the entire nose toward the right side, and removed the nasal septum, the turbinates, the medial wall of the orbit, and the maxillary sinus. The patient died 2 months later of acute intracranial hypertension. An autopsy examination revealed that a large portion of the tumor had remained.

Between 1907 and 1912 many surgeons attempted to develop innovative approaches to the pituitary gland while trying to minimize invasiveness, the incidence of postoperative meningitis and damage to the visual system. General surgeons in Europe were among the first to approach the pituitary gland transnasally. Because of their lack of experience with illumination and depth of field in such a restricted surgical space, however, they created unnecessarily large external nasal openings. This approach was soon replaced in favor of less invasive procedures.

In 1909, the Viennese surgeon Hirsch conducted the first successful endonasal transsphenoidal pituitary intervention. He performed hundreds of these operations, first in Vienna until 1938 and then for another 20 years in Boston. In a modified form, his technique is still in use today. Inspired by Hirsch's experiences, American surgeon Harvey Cushing performed about 250 transsphenoidal pituitary procedures between 1910 and 1929. In contrast to Hirsch, he accessed the nasal sinuses sublabially, rather than endonasally. Due to the difficulties and the significant danger of complications, however, the method failed to be established as a widely accepted standard procedure. Even Cushing himself abandoned the

transsphenoidal approach in favor of the transcranial access.

For the next years the transsphenoidal approach remained essentially forgotten. It was not until 1962 that it was reestablished. Numerous innovations caused new interest in transsphenoidal surgery. Gérard Guiot of Paris introduced the technique of intraoperative fluoroscopy, allowing the surgeon for the first time to visualize the depth and positioning of surgical instruments intraoperatively. The major advantage of fluoroscopy was the monitoring of the instrumental maneuvers on the television screen while performing intracranial interventions. This modality of real-time visualization revolutionized the technical aspects of pituitary surgery and set the stage for subsequent innovations in intraoperative image guidance.

After the establishment of the basic operative techniques, further advances in the surgeon's ability to visualize the operative field and tumor bed were required to improve patient outcomes and expand the versatility of the technique. Inadequate illumination of the operative field had always limited the transsphenoidal method. Cushing had used a headlight. Some surgeons such as Hirsch had operated using a mirror attached to the surgeon's face and reflecting the bundled light of a strong light source onto the operation site. Others had tried to improve illumination by attaching small light bulbs near the tip of an operating instrument.

Although surgeons had used operating microscopes since the 1920s when performing ear-related surgery, its application to transsphenoidal surgery was pioneered by the Canadian neurosurgeon Hardy many years later, in 1965. Not only did this improve illumination, it also added intraoperative magnification and stereoscopic visualization. The microscope and instruments were quickly adopted and have become an essential part of transsphenoidal approaches to the pituitary gland. From that time, due to better visualization, the danger of complications and the morbidity rate could be substantially reduced. Furthermore, since then it is possible to remove pituitary adenomas without damaging the pituitary gland.

Although the operating microscope improved a lot upon safety and acceptance of the transsphenoidal procedure, the growing need for even better accuracy and visualization was apparent. With the operating microscope, visualization during pituitary surgery is limited to a straight and narrow view. Angled optics allowing views 'around the corner' or optics with large opening angles, increasing the field of view, are not available. More recently, endoscopy has been recognized as an acceptable tool in pituitary surgery.

Endoscopy emerged in the early 1900s when the first attempts to view inside the body with lighted telescopes were made. These initial devices were mostly fully rigid. In the 1930s, semi-flexible endoscopes called gastroscopes were developed to view the inside of the stomach. Fiber-optic endoscopy was pioneered by the physician Basil Hirschowitz at the University of Michigan in 1957. Widespread use of fiber optic endoscopes began in the 1960s. A fiber optic cable is a bundle of microscopic glass or plastic fibers that allows light and information to be transmitted through straight as well as curved structures.

Although introduced to neurosurgery almost a century ago, the use of endoscopy in neurosurgery was for a long time primarily limited to intraventricular procedures. In 1963, Guiot was the first to report the use of an endoscope during sublabial transsphenoidal surgery. Additionally, the endonasal route first used by Hirsch in 1909 was reintroduced by Griffith and Veerapen in 1987. During the mid-1990s, endoscopy used additionally to the microscope was described in multiple studies as a way to provide improved visualization of the

pituitary gland and surrounding structures. The authors claimed that endoscopy facilitated better dissection of the tumor. Supporters of the technique went also one step further by using endoscopy, without an accompanying microscope, for the removal of pituitary tumors. In 1992 this kind of operation was first reported. Nowadays, the use of endoscopy as a stand-alone instrument in pituitary surgery is becoming increasingly popular.

However, still there is an ongoing discussion about whether endoscopy or the microscope is more suitable for transsphenoidal surgery. One advantage of endoscopes is the possibility to use angled optics, which, diverting the principal viewing vector such that it is not parallel to the longitudinal axis of the endoscope, but rotated (e.g., by 45, 60 or 90 degrees), allow for visualization of otherwise blind corners. Another positive point of endoscopy is the ability to change intrinsic camera parameters and thus perspectives between close-up and panoramic settings. Furthermore, endoscopes are more freely maneuverable and allow an intervention with less postoperative trauma for the patient.

The lack of binocular viewing, the subsequent lack of depth of field, the increased difficulty coping with intraoperative bleeding, the lack of dedicated instruments, and the large learning curve for neurosurgeons already trained in microsurgery, however, have hampered the universal adoption of endoscopy in pituitary surgery.

3.4 Radiological Imaging

With the introduction of x-ray imaging by Wilhelm Konrad Röntgen in 1895, the field of radiology emerged. It enables physicians to acquire visual information of the inside of the human body without actually opening it. Over the years, radiological imaging has evolved reaching its temporary climax in today's computer-driven scanners which are capable of generating high-quality images of inner anatomy at high resolution and speed.

Some of the currently most important radiological modalities are:

- **Computed Tomography (CT):** In CT, an x-ray tube rotates around the patient's body and numerous x-ray beams are emitted from about a thousand different directions and passed through the anatomy at different angles. Special sensors measure the amount of radiation absorbed. The acquired data are computationally processed to form cross-sectional images, or *slices* of the anatomy of interest. These slices are called tomograms, hence the name "computed tomography." After computation of one slice, the tube moves on along the body to generate another slice. The resulting CT volume data set is therefore a stack of slices. Modern CT scanners compute up to 64 slices simultaneously (MDCT, Multi-Detector CT) which considerably reduces the scanning time.

  Since the amount of radiation absorbed is directly proportional to tissue density, a CT in fact measures tissue density. The standard scale used to describe CT data is the *Hounsfield Scale*, measuring intensity in *Hounsfield Units* (HU). CT data can quite reliably be mapped to this scale, with respect to constituted reference values. The value of 0 HU was, for example, defined to describe the density of distilled water, while -1000 HU is the radiodensity of air. Massive bone is typically 400 HU or greater. Thin bones may be captured at lower density. Metallic implants (e.g., stents)

are usually approximately 1000 HU.

By applying a contrast agent, blood vessels can be emphasized in the resulting data set (CTA - *Computed Tomography Angiography*). Contrasted blood is usually scanned at a similar level as massive bones, or higher.

The biggest downside of CT examination is that the patient is exposed to radiation.

- **Magnetic Resonance Imaging (MRI):** MRI uses radiofrequency waves and a strong magnetic field rather than x-rays to provide detailed pictures of internal organs and tissues. The magnetic field lines up protons in the tissues, which are then excited by a beam of radio waves. As they align themselves again with the magnetic field, they produce signals that are picked up by a receiver in the scanner. The technique has proven very valuable for the diagnosis of a broad range of pathologic conditions in all parts of the body, including cancer, heart and vascular disease, stroke, and joint and musculoskeletal disorders. MRI allows the evaluation of some body structures that may not be as visible with other imaging methods. Downsides of MRI, as compared to CT, are higher cost (partially caused by the need for specialized equipment and expertise), a higher degree of noise in the resulting data, numerous different sources of high- and low-frequency artifacts hampering automated data analysis and the absence of a reliable absolute data scale.

- **Magnetic Resonance Angiography (MRA):** MRA is the magnetic resonance counterpart of CTA. In MRA data sets, blood vessels are emphasized. MRA is actually MRI with the use of a certain adjustment and hence, other than CTA, does not require the injection of a contrast agent, though a contrast agent may be used to enhance the signal. Also, since bones are not captured by MRA, blood vessels can be well distinguished from bones, which is often difficult in CTA data. The disadvantage of MRA, compared to CTA is the longer scanning time, which makes MRA sensitive to patient movement.

- **Ultrasound (US):** Ultrasound uses high-frequency sound waves to image a particular region or organ in the body or to monitor blood flow. Sound waves are focused at a part of the body, they interact with the tissue and the returning echoes are processed by a computer to yield images of the part of the body being examined. The returning sound waves, or echoes, reflect size and shape of the structure and whether it is solid, fluid, or something in between. Similarly to CT and MR, US generates two-dimensional images. Usually the US probe is manually maneuvered over the region of interest. For 3D reconstruction, however, positional orientation devices are necessary for registering the spatial coordinates of each US image. Some systems use motors that move the US probe in a predetermined manner. An alternative is to use a position sensing system.

  US images frequently contain severe artifacts. This is the main reason why US has lagged behind CT and MRI, although recent machines have advanced significantly in addressing this issue.
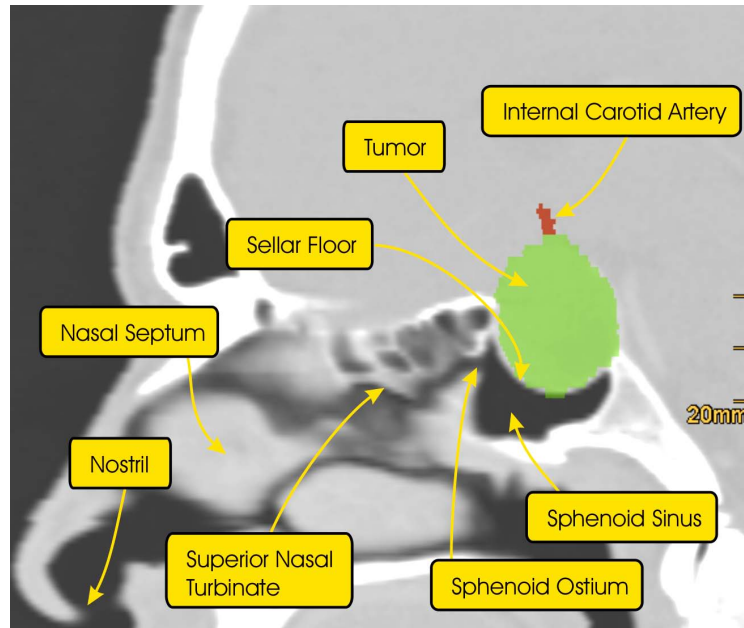
Figure 3.2: Anatomy as captured by a CT. A sagittal slice through a CT data set of a head.

3.5 Endonasal Transsphenoidal Pituitary Surgery

Endonasal transsphenoidal pituitary surgery is a relatively new endoscopic procedure used for the removal of pituitary adenomas: A rigid endoscope and surgical instruments are inserted into the patient's nose and advanced through the natural nasal airway along the nasal septum and through the artificially enlarged *sphenoid ostium* into the *sphenoid sinus*. Inside the sphenoid sinus, the tumor is usually occluded by the *sellar floor*. The sellar floor is opened and the tumor can be divided from the surrounding tissue and removed through the patient's nose. Figures 3.2 and 3.3 illustrate the traversed anatomy.

3.5.1 Preoperative Evaluation and Planning

The first steps in preoperative evaluation are cerebral MRI and CT examinations. Based on MRI data, knowledge about properties of the tumor can be gathered. The MRI provides a suitable contrast between the tumor, the pituitary gland and neighboring structures, giving valuable information about the anatomical character and relationships of the adenoma.
CT images record the anatomy of the patient's skull. They give information about the shapes of the paranasal sinuses and other cavities and the maneuverability inside them. This facilitates planning the route to be followed during the endoscopic procedure. The nasal septum, for instance, may be significantly curved, rather than straight. Acute bowing and deviation of the nasal septum occur in about 20 percent of the population. When severe, the deviated septum can cause obstruction, hindering the approach through one or both nostrils. CT data exactly record the shape of the nasal septum.
CT data provide accurate information about the existence of *septa* in the sphenoid sinus. The space within the sphenoid sinus is subdivided by one or several septa. Single septa do
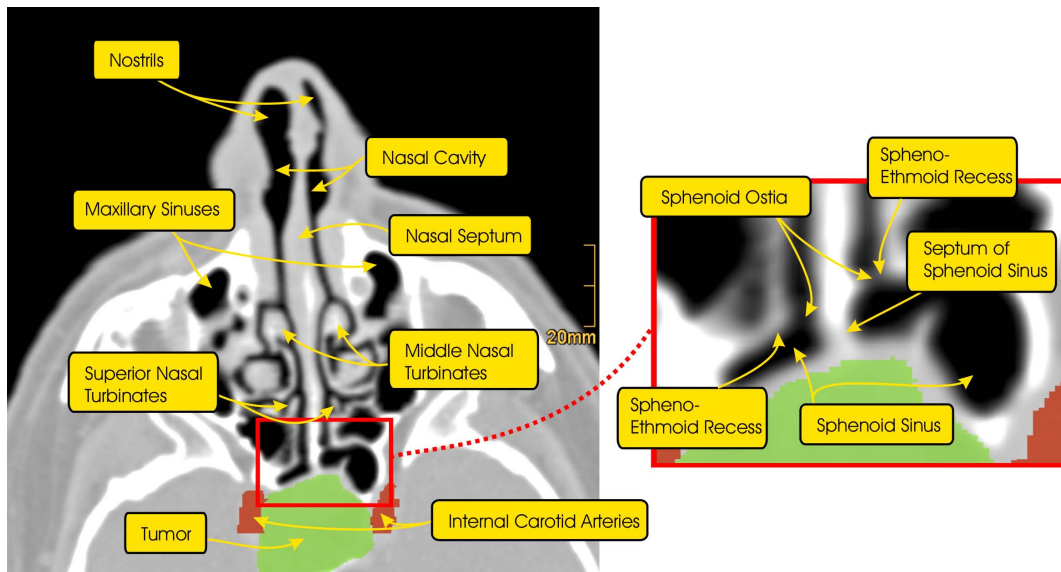
Figure 3.3: Anatomy along the path taken by the endoscope.

not always mark the midline of the sphenoid sinus. Additionally the chambers of the sinus may be subdivided by incomplete septa (see Figure 3.4). In the transsphenoidal approach, sphenoid septa block the way to the sellar floor and must be removed. Preoperative knowledge about the locations and shapes of septa is important since it influences the surgical strategy.

Information about the size and shape of the sellar floor and the existence of a possible *Onodi-cell* can be retrieved from CT data. According to literature, Onodi cells occur in only a small percentage of the population. Intraoperatively, Onodi cells influence the surgeon's visual impression of the operation side and can induce navigation difficulties.

Furthermore, based on CT (and MRI) data, a virtual endoscopy navigation study can be performed, if suitable software is available.

An endocrinological examination is conducted to assess the patient's hormone metabolism. Furthermore, the function of the visual system is tested.

3.5.2 The Operating Theater

After the application of general anesthesia, the patient's head is aligned horizontally and fixed using a headholder. The patient is draped sterile leaving visible only the nose. The navel is kept quickly accessible, in case an autologic transplantation of fat becomes necessary. To facilitate the access to the sphenoid sinus, the nasal airways are widened using decongestion. To prevent inflammation, disinfection is applied.

The complete procedure is performed using endoscopic vision. The endoscope employed must be a rigid scope, 4 mm or less in diameter and 18 to 30 cm in length (see Figure 3.5). During phases in which the endoscope traverses the patient's head, the endo-
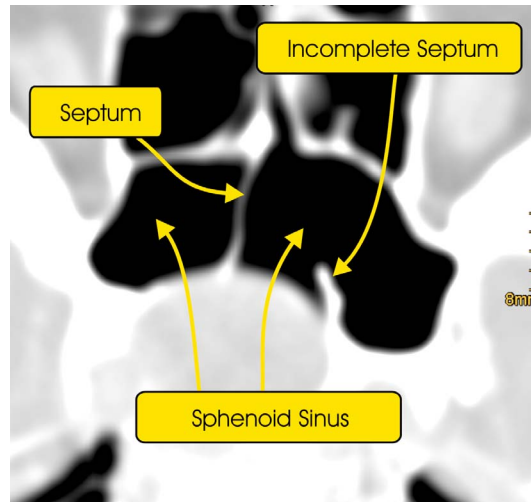
Figure 3.4: The sphenoid sinus is divided into chambers by one or more complete or incomplete septa

scope is manually maneuvered by the non-dominant surgeon's hand, the other hand is used to handle surgical instruments. In phases of surgical activity, the endoscope can be fixed using an adjustable endoscope holder (see Figures 3.5 and 3.6, leaving both the surgeon's hands vacant and providing a fixed image of the operating field.

The endoscope is used to, supported by vision, lay a linear path to the operating site. Any instruments needed can then be easily inserted next to the endoscope.

3.5.3 The Surgical Procedure

Depending on the width of the nasal cavity and the position of the tumor, one nostril is chosen for the surgery. Surgeons often inspect both nostrils with the endoscope to gather additional (3D) information to determine the most suitable approach. The unilateral (through one nostril) approach is the standard method. Depending on patient anatomy and properties of the tumor, a bilateral approach using both nostrils may be helpful for improved operating conditions. The bilateral approach may, however, be associated with increased patient trauma. The surgical intervention is illustrated in Figure 3.7. The endoscope is inserted into one of the patient's nostrils (image 1, Figure 3.7) and advanced along the nasal septum. The traversal of the endoscopic camera through the nasal cavity is usually referred to as the *nasal phase* of the procedure. The first important structure to be identified is the *middle nasal turbinate* which represents a landmark in the determination of the correct direction of further traversal (see image 2, Figure 3.7). The middle nasal turbinate may have to be slightly dislocated to create room for the endoscope. Then the endoscope is advanced along the middle nasal turbinate until the *spheno-ethmoid recess* is reached. In the upper half of the spheno-ethmoid recess there is a natural opening, the *sphenoid ostium*. The ostium in its natural state is too small to be traversed by the endoscope (see image 3, Figure 3.7). It must be enlarged using a *bone punch* or *rongeur* (see image 4, Figure 3.7). This process is called *sphenoidotomy*. The ostium is usually enlarged to a diameter of 15 to 20 mm. Depending

Figure 3.5: The rigid endoscope during surgery. The image is courtesy of the Institute of Neurosurgery, Medical University Vienna.

on the anatomy and the size of the sella turcica and the tumor, the opening may have to be further enlarged rostrally (towards the face). As soon as the ostium is wide enough, the endoscope can be advanced through it. At that point, the nasal phase is finished and the *sphenoid phase* of the procedure is commenced. The septa in the sphenoid sinus usually must be removed (see image 5, Figure 3.7), again using the bone punch. This is not only to get sufficient access to the *sellar floor*, but also to be able to gather hints about the size of the sella turcica (and the pituitary gland), the locations of important anatomical structures that must not be damaged and properties of the remaining anatomy.

Next, at the beginning of the *sellar phase* of the procedure, the mucosa covering the sellar floor is removed. After this, the sellar floor can be opened using a drill or a small chisel (see images 6 and 7, Figure 3.7). The generated hole is then enlarged using the bone punch (see image 8, Figure 3.7). The hole allows access to the pituitary gland and the tumor. In a later phase of the intervention, the tumor will be removed through the same hole.

Before that, the *dura mater* underneath the hole is opened. If the tumor is a *microadenoma* and obscured by the pituitary gland, the gland might have to be dissected to reveal the complete adenoma. *Macroadenomas* are usually clearly visible at this stage.

A biopsy is taken from the tumor for histological analysis of the tumor tissue.

The tumor must be completely resected (divided from surrounding tissue and removed). In cases of CSF leakage, the space that was filled by the adenoma must be filled via an autologic transplantation of fat or muscle tissue taken from the patient's abdomen. The sellar floor is reconstructed using fibrinous glue and possibly the still available piece of bone that was removed during the opening. The sphenoid sinus and the nasal cavity are finally examined for mucosal lesions. This concludes the intervention.
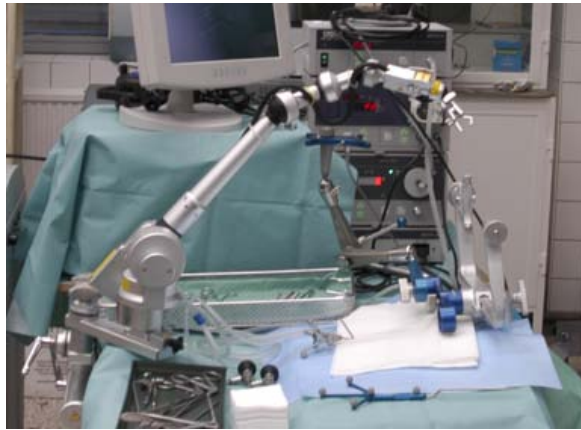
Figure 3.6: An adjustable endoscope holder. This holder allows 9 DOF endoscope movements. The image is courtesy of the Institute of Neurosurgery, Medical University Vienna.

3.5.4 Postoperative Evaluation

The patient, if free of complications, usually leaves the hospital after five to eight days. After two weeks endocrinological examinations are conducted. Generally, after three, nine and twenty-one months, MRI examinations are performed.

3.5.5 Risks of the Intervention

The surgical procedure is associated with some risk: The internal carotid arteries (ICAs) pass just laterally and the optic nerves superiorly to the pituitary gland behind the sellar floor and are therefore invisible to the surgeon. Damage to these structures and the pituitary gland itself must be prevented. In most patients, the cavernous sinuses containing the ICAs bulge the posterio-lateral wall of the sphenoid sinus. These prominences are called *carotid protuberances*. Furthermore, the intersinus septum often deviates from the midline of the sphenoid sinus and may be attached to the prominence of the ICA. The ICAs can therefore be well localized. This, however, is not true for all patients and should not be fully relied on. The channel containing the optic nerve may also cause a bulging of the roof of the sphenoid sinus (the *optic protuberance*).
A large tumor is often in contact with the artery wall and may even surround it. Its resection thus includes surgical activity in immediate vicinity to an ICA.
Intraoperative spatial orientation is sometimes made difficult by variations in the pneumatization of the sphenoid sinus, a possible additional air-filled cell (Onodi cell) above the sphenoid sinus and variations in the anatomy of the spheno-ethmoid recess.
The depth of the sphenoid sinus and the visibility of the sella turcica are dependent on the pneumatization of the cranium. If the sphenoid bone is poorly pneumatized (as typically observed in children), only a few landmarks may be available for identification of the sellar floor and localization of the ICAs.
In order to keep risk at a low level, it is obviously of utmost importance for the surgeon to be skilled and experienced as well as to get familiar with the patient's anatomy prior to each

Figure 3.7: The single steps of transsphenoidal endoscopic pituitary surgery. 1.: The endoscope is entered into the nose, 2.: The endoscope is advanced along the nasal septum, the middle nasal turbinate becomes visible; It presents an important orientation landmark; 3.: The sphenoid ostium is too small to pass through; 4.: The ostium is enlarged using a punch; 5.: Removal of sphenoid septa; 6.: The sellar floor is opened using a small chisel; 7.: Result after application of the chisel; 8.: The hole has been widened using a rongeur; The images are courtesy of the Institute of Neurosurgery, Medical University Vienna.

surgical intervention.

3.6 Virtual Endoscopy for Transsphenoidal Pituitary Surgery

As outlined in the previous section, the opening of the anterior sellar wall and the exposure of the dura is an important stage in a transsphenoidal pituitary adenoma operation. The landmarks inside the sphenoid sinus, such as the intersinus septum and the possible partial septations, the sella turcica itself, the wall of the sinus below the sella and the possible prominences of the ICAs and optic nerves, must be identified correctly before the sella can be safely opened. The exact projections of the pituitary gland, adenoma and ICAs on the posterior sinus wall should be identified by the surgeon before the sellar wall is opened. This is traditionally achieved by analysis of radiological images. Acquisition and analysis of both MRI (or MRA) and CT scans of the patient's head are usually part of the preoperative routine diagnostic workup. Besides standard 2D viewing techniques, the surgeon can therefore, without any additional radiological examinations, use virtual endoscopy. Virtual endoscopy can be of benefit in three different ways:

Preoperative Planning: While careful slice-by-slice-analysis of the raw images obtained from the modalities can give a sound overall picture of the situation to a skilled surgeon, this picture can still be enhanced by additionally using simulated 3D views as provided by virtual endoscopy. This helps to get familiar with the anatomy of the patient, decide on the path to take and plan where exactly to open the sellar floor.

Also, semi-transparency can be used: Rendering the tumor, the pituitary gland, the ICAs
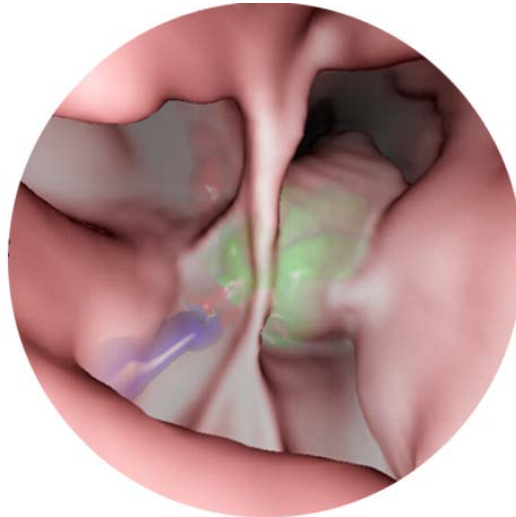
Figure 3.8: The sphenoid sinus. The tumor (green), the internal carotid arteries (red) and the optic nerves (blue) are rendered in the background. The pituitary gland is completely occluded by the large tumor.

and the optic nerves behind the semi-transparent sellar floor (see Figures 3.8 and 3.9) can give the surgeon a profound impression of how to minimize the risks associated with the procedure.

Intraoperative Support: Virtual endoscopy can be used during the real surgical procedure both as a navigation aid and to decide on the exact opening site of the sellar floor. If the position of the real endoscope is tracked, the viewing parameters can be retrieved and transferred to the virtual endoscopy system.

Training: Safe execution of the procedure requires training. It is, for instance, a hard task for beginners to find the correct path to the sphenoid ostium. Inexperienced surgeons might easily take a wrong direction and end up, for example, in the patient's throat. Beginners are advised to follow certain anatomical landmarks. For instance, Cappabianca et al. [CAdDT01] propose to follow the inferior margin of the middle nasal turbinate until the upper edge of the nasal *choana* appears. At that point, the well visible *superior nasal turbinate* marks the direction to the sphenoid ostium (see Figure 3.10). This strategy works well, but must be trained until it can be safely applied. Also, most importantly, the decision upon the opening site of the sellar floor must be trained.

Simulation of the real surgical operation by not only providing the visual impression but also modeling haptic parameters and constraints of movement and simulating surgical instruments can serve as a tool to exercise the complete intervention. Furthermore, virtual endoscopy can help the trainee to get used to visual parameters offered by endoscopy: These include image distortion, non-stereoscopic 3D images and a limited range of light. De-Divitiis et al. [dDCC03] state that the surgeon, especially if inexperienced, must prevent the endoscope from performing too quick movements in the vicinity of the surgical target,

24
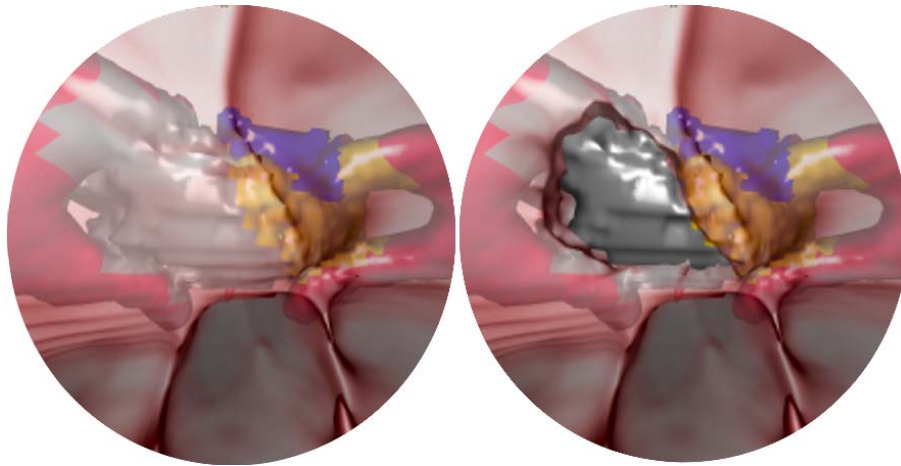
Figure 3.9: The background objects indicate where to open the sellar floor. Here the tumor (grey) is adjacent to the internal carotid artery (red). Besides the adenoma, the patient also harbors an Arachnoid Cyst (blue) near the pituitary gland (yellow). The jagged boundaries between objects reflect the voxel-structure of the underlying segmented objects (see Section 6.3). In order to avoid damage to the ICA and the pituitary gland, the opening of the sellar floor should be confined to the parts not overlying these structures. In this patient, accuracy is obviously critical. The images are courtesy of the Institute of Neurosurgery, Medical University Vienna.

because the two-dimensional endoscopic image can cause the surgeon to lose the sense of depth inside the nasal cavity and the sphenoid sinus, increasing the danger of causing serious damage.

De Divitiis et al. [dDCC03] recommend to perform a virtual endoscopy study, if possible, prior to each transsphenoidal surgical intervention.
Virtual endoscopy has been reported to be usable for visualization of the nasal cavity and the paranasal sinuses [NSS97].
Talala et al. [TPK+00] assessed the benefits of a combination of virtual endoscopy with non-perspective semi-transparent imaging of surface landmarks for identifying a safe strategy for opening the sellar floor in transsphenoidal pituitary surgery. In their system, parts of the ICAs passing close to the sellar floor are manually segmented and projected to the rendered surfaces. This gives a useful preoperative impression of patient anatomy. The lack of perspective multi-modal rendering, however reduces the similarity to reality and therefore the applicability. Still, the authors state that the bony opening of the anterior sellar wall was significantly facilitated by the 3D CT imaging and an additional safeguard was achieved by marking the projections of the ICAs on the 3D image. Virtual endoscopy enabled the surgeon to be familiarized preoperatively with the endoscopic orientation of the landmarks in the individual sphenoid sinus.
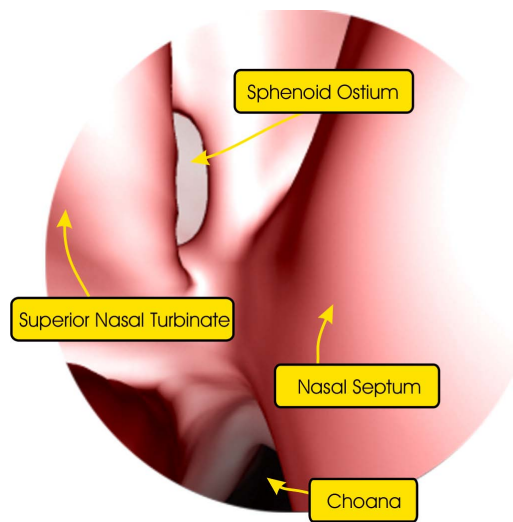
Figure 3.10: The choana is the posterior aperture of the nasal cavity. Near its upper edge, the superior nasal turbinate can be detected. The turbinate must be passed laterally to find the sphenoid ostium.

Chapter 4

Virtual Endoscopy - History, Techniques and Applications

This chapter introduces the concept of virtual endoscopy, presents techniques developed to facilitate interactive navigation and gives an overview of fields of applications.

Although endoscopic surgery presents an improvement upon traditional open surgery, still several drawbacks are associated with it: Endoscopy is minimally invasive, but still involves deterrent and deleterious effects. It is unpleasant for the patient and often requires sedation. Endoscopic procedures are cheaper than open surgical procedures, but still expensive, requiring manpower, expensive equipment and medication. Some areas of interest cannot be reached by an endoscope (e.g., due to folds and anatomical constrictions). Minimally invasive procedures lack the fast access of open surgery in case of serious complications, such as strong bleeding. Furthermore, handling of endoscopes is difficult, mainly due to limited flexibility and limited field of view through the endoscope. Endoscopic procedures therefore require the surgeons to be skilled, experienced and well-prepared.

Virtual endoscopy is a concept allowing the user to navigate through computationally reconstructed patient anatomy using a virtual camera, mimicking real endoscopy. It fuses radiological imaging with advanced techniques for rendering three-dimensional images to produce views of an anatomical structure of interest similar to those obtained during physical endoscopy. Depending on the area of application, virtual endoscopy can be used either instead of or additionally to physical endoscopy.

Virtual endoscopy principally derives from digital medical imaging, but has also roots in non-medical areas of 3D visualization, including terrain guidance, flight path planning and flight simulation [Rob00]. David Vining who used virtual endoscopy inside the colon [VG94] is often referred to as the inventor of virtual endoscopy. Further early work on virtual endoscopy was published by Lorensen et al. [LJK95], performing 3D fly-throughs of carotid arteries and A-V malformations, Robb and Cameron [RC95] who began with patient specific 3D organ visualizations and progressed to interactive organ fly-throughs,
Hara et al. [HJR$^+$96] who published early clinical observations in the colon, Kikinis et al. [KGJ95] who used 3D visualization and image fusion for stereotactic neurosurgery, and

Napel et al. [NRB⁺96], applying simulated endoscopy to visualizations inside a variety of organs.

Based upon this early work, virtual endoscopy became a very active area of research. Still, every year, a substantial number of researchers both from the medical and the technical community, present and assess new techniques and fields of application. Supported by this wide field of collected knowledge, virtual endoscopy is now making its way from research into clinical practice.

This chapter provides an overview of past and current research in the field of virtual endoscopy. Other surveys of this topic were written by Vilanova [Vil01] and Bartz [Bar03b].

Virtual endoscopy is based on radiological data. Before the virtual examination, the investigated part of patient anatomy is scanned using one (or more) of the following modalities:

- **Computed Tomography (CT):** Most virtual endoscopy systems are based on CT data. The CT is the preferred modality, since it often provides a suitable contrast between the inner and outer of the investigated structure of interest.

- **Magnetic Resonance Imaging (MRI):** MRI has the advantage that the patient is not exposed to deleterious radiation, so should be applied instead of CT wherever applicable. The usability for virtual endoscopy is, however, restrained by its disadvantages, including disturbing artifacts and noise. Thus, MRI data usually has to be preprocessed (e.g., segmented) before used for virtual endoscopy.

- **Magnetic Resonance Angiography (MRA):** MRA is often applied instead of MRI if the display of blood vessels is essential. Otherwise, MRI is preferred due to shorter scanning times.

- **Ultrasound (US):** Due to low image quality, ultrasound is of minor importance in virtual endoscopy, but included here, because its use and its feasibility have been reported, e.g., in the assessment of Kawasaki disease [AHR01].

The result obtained from the scanning process is a volume data set in which the anatomical region under investigation is represented as a discrete three-dimensional array of scalar data samples, so-called *voxels*. The voxel (volume element) is the three-dimensional equivalent of a pixel (picture element) and represents a quantity of 3D data just as a pixel represents a point or cluster of points in 2D data.

The function $\rho(x, y, z)$, assigning a data value to any point inside the volume must be reconstructed from the voxel values. The function most often used for reconstruction is *trilinear interpolation*. Here the value at a certain position is a weighted average of the values of the eight surrounding voxels, with the weights depending on the axial distances between the point and each voxel.

The obtained volume data set is used to reconstruct the walls of the investigated anatomical structure.

Virtual endoscopy is nowadays used for a variety of purposes:

- **Diagnosis:** Some applications of endoscopy have a very prominent diagnostic aspect (e.g., colonoscopy). In these applications, surgery is needed only, if pathologies are detected. In case the probability of detecting existing pathologies is sufficiently high, virtual endoscopy can be used as a tool for diagnosis. Virtual endoscopy avoids the risks associated with real endoscopy and is substantially more comfortable for the patient.

- **Preoperative Planning:** When used prior to performing an actual endoscopic exam, virtual endoscopy can minimize procedural difficulties and decrease the rate of morbidity, especially for endoscopists in training. Preoperative planning is, traditionally, performed on the basis of analysis of radiological images. Virtual endoscopy can, to some extent, improve upon this. It is generally admitted that the viewer can better appreciate spatial relationship in a three-dimensional display than in a series of 'flat' two-dimensional images. Although 3D rendered images contain no more information than conventional radiological images, they offer a more rapid appreciation of spatial information. Ghani et al. [GPPA04] state that a real understanding and appreciation of anatomical relationships comes with studying cadavers or three-dimensional models. They predict that in the future it will become economically unfeasible and practically difficult for doctors to continue processing the large volumes of axial data that are currently obtained. Inevitably data will need to be displayed in a manner that is most easily accommodated by the human eye.

- **Intraoperative Support:** Virtual endoscopy can be performed simultaneously to a real endoscopic procedure, providing a tool for navigation and for obtaining increased insight. The latter can be gained, for example, by using virtual endoscopy to look beyond anatomical surfaces, but also to view regions outside the field of view of the physical endoscope. The position of a real endoscope may be tracked (using a 3D navigation system) and mapped into the image stack acquired from the modality. In this case, a coupled display, generated by physical and virtual endoscopy can be provided, showing the real endoscopic film and supplementary information provided by virtual endoscopy on one screen.

- **Postoperative Follow-Up:** Virtual endoscopy can often be of use in follow-up treatment and assessment. In cases when patients have to undergo a further radiological examination a few weeks after a surgical intervention, virtual endoscopy can give important information about the success of the operation.

- **Training:** The improvement and standardization of education in endoscopy has become a major issue during the past years. Problems have been noted primarily in teaching endoscopic novices, as newcomers to endoscopy usually need a large number of procedures to gain competency and suffer from reduced time for individual learning. Virtual reality simulators offering virtual endoscopy under 'realistic' circumstances are more and more emerging as suitable and cost-effective training tools. Simulator-based training accelerates and improves training in endoscopy for beginners, maintains competency when endoscopic procedures are not regularly performed and enables testing and learning of new, mainly interventional methods, before the procedure is performed on a patient. Computer-based training is simple, relatively

inexpensive, and requires minimal instruction. Ferlitsch et al. [FGG$^+$02] suggest that virtual endoscopy simulation should comprise one step in a structured endoscopic training program, depending on the level of expertise and the progression of skill in novice endoscopists.

Other than for the other areas of application mentioned here, where accurate rendering is one of the major factors that determine usability, for training purposes the accurate representation of individual patient anatomy is of minor importance. Here the emphasis is laid on realistic simulation of the procedure and quick, possibly realistic, depiction of representative anatomy.

Apart from the training effect, virtual endoscopy can also provide a validated and much needed method for objective assessment of the endoscopist's technical skills, giving, for instance, a measure of whether an inexperienced endoscopist can be admitted to physical surgery.

Virtual endoscopy has the following inherent advantages, compared to physical endoscopy:

- **Improved Access:** Virtual endoscopy allows visualization of regions that are difficult to access physically, either due to disease (e.g., strictures, stenoses), or due to general impossibility of placing a physical camera into certain parts of anatomy (e.g., the interior of blood vessels). Furthermore, the field of view provided by a physical endoscope is usually restricted. Virtual endoscopy is capable of depicting any body part that can be captured by a radiological modality.

- **Further Decreased Invasiveness:** Virtual endoscopy is less invasive than physical endoscopy and thus avoids complications such as perforation, stricture formation, infection or hemorrhage. In contrast to physical endoscopic procedures, virtual endoscopy does not require any patient sedation.
  These factors significantly increase patient comfort and acceptance and therefore the chance of patients acceding to the conduction of an examination.

- **Parameter Control:** In virtual endoscopy the user is given an increased amount of control over the visual outcome. The user (or, at least, the developer) can control colors, appearance of objects and other aspects of the user-computer interface, like the handling of virtual instruments.
  The amount of light transported through the fiber-optic of a physical endoscope is limited, often leading to dark images. In virtual endoscopy, the brightness of images can be made freely adjustable to optimize either visibility or realism.

- **Lower Cost:** Virtual endoscopy can in some cases be a considerably cheaper alternative to real endoscopy, since usually no sedation, patient preparation and hospitalization are required.

- **Geographical Decoupling:** Data acquisition and the virtual procedure need not necessarily be conducted at the same location. This facilitates tele-medical procedures.

- **Additional Information:** Virtual endoscopy can be used either before or after physical endoscopy, increasing the amount of information gained about the patient anatomy,

either reducing the risks associated with the physical procedure or to assess its outcome.

In virtual endoscopy, walls of investigated cavities can be rendered semi-transparently. This allows to display structures (e.g., blood vessels, nerves) which in physical endoscopy are occluded.

On the other hand, virtual endoscopy suffers from a number of limitations:

- **Artifacts:** Virtual endoscopy is based on radiological, i.e., sampled data. A significant amount of knowledge about tissue properties and shapes is lost during the production of these data. The Nyquist theorem states that sampling preserves details of up to half the sampling frequency, higher frequency detail is not captured. Scanning resolution (and therefore frequency) can be increased by using modern, faster scanners, but it remains limited, whereas the frequency spectrum of anatomical detail is theoretically unbounded. Resulting artifacts are especially significant in virtual endoscopy, since perspective distortion is essential and often results in small structures filling a large fraction of the screen. Bartz [Bar03a] lists three different undesired effects resulting from violation of the sampling theorem. They are:

  - **Partial Volume Effects:** The function defining a data volume acquired from a radiological scanner has a Gaussian-like profile with a finite support [SK98]. It is low-pass filtered, suppressing high frequencies and blurring object edges. Partial volume effects are the result of filtering artifacts due to sampling at insufficient frequency. They arise in regions showing high intensity gradients, e.g., where bone meets soft tissue. Radiological data does not capture such transitions accurately. Voxels near the gradient acquire an averaged value, somewhere between the data levels of the involved tissue types. The contrast is therefore smoothed out. The results are artificial connections between actually disjoint objects and, on the other hand, loss of thin structures.

  - **Staircase Artifacts:** Since radiological data is defined only for discrete points *(voxels)*, the data in between voxels must be acquired using interpolation. Often, voxels are arranged in such a way that they define a grid of cuboid (but not cube)-shaped cells. This can result in artifacts giving the impression of stairs in reconstructions of smooth surfaces. Very disturbing discontinuities may appear on surfaces near significant gradients (e.g., at the boundaries of binary segmented objects).

  - **Loss of Small Structures:** Structures smaller than the inter-voxel distance in any direction may be completely lost.

- **No Tissue Interaction:** Since virtual endoscopy is not applied to the patient, but to a radiological image of the patient, it is impossible to take a biopsy, needed for histological examination of tissue to test for certain parameters, e.g., malignancy of tumor tissue. If a medical procedure includes the removal of tissue (e.g., adenomas, colon polyps), it cannot be replaced by virtual endoscopy.

  Soft tissue deformation and haptic feedback can be added only by means of simulation.

- **Lack of Mucosal Detail:** Existing radiological imaging devices do not differentiate the fine detail in the mucosal lining of anatomic structures that can be seen with conventional endoscopic procedures. Thus, textural detail sometimes relevant for diagnosis, such as structure, color and reflections, is not available in the reconstructed anatomy. Any applications depending on this information will not succeed with virtual endoscopy.

- **Dependency on Segmentation:** Often, significant parts of anatomy are not sufficiently emphasized within the underlying radiological data (e.g., due to missing gradients at object boundaries). Display of these structures is dependent on segmentation which can be a tedious, time-consuming task. Incorrect segmentation can lead to erroneous judgment by the physician.

- **Scalarization:** Since during the scanning process all tissue properties are integrated to scalar data values (e.g., tissue density in CT, unit-less signal intensities in MRI), tissues of different types can be matched to similar data levels. These tissue types can, without the use of additional high-level knowledge, not be distinguished any more.

- **Exposure to Radiation:** X-ray based computed tomography (CT) is the most widely used image acquisition modality for virtual endoscopy. CT exposes the patient to radiation which can have malignant effects on the organism.

These limitations show why virtual endoscopy can not completely replace real endoscopy. Still, its advantages not only make it a valuable addition, but also motivate further research and improvement, as well as the search for further areas of application.

The following section introduces ways to reconstruct and render patient anatomy.

4.1   Rendering Techniques for Virtual Endoscopy

The core part of any virtual endoscopy system is the rendering engine which reconstructs anatomical structures from the volume data. Since interaction is vital in virtual endoscopy, rendering must be fast, resulting in interactive frame rates. A variety of rendering techniques have been introduced and found useful for virtual endoscopy. They can roughly be classified into two groups - volume rendering techniques and surface rendering techniques.

- **Volume Rendering:** In volume rendering, a transfer function is used to map data values to color and opacity levels. According to this information, the volume is then projected to screen space, resulting in a layered rendering, with different tissue types accumulating to the final image appearance.

- **Surface Rendering:** In surface rendering, only surfaces are reconstructed. Surfaces can be of two types: The first type are isosurfaces. An isosurface is the set of all points (x, y, z) inside the 3D data set, with $\rho(x, y, z) = v_{iso}$, with v_{iso} being called the *iso-value* or *threshold*. Iso-surfacing is equal to volume rendering with a special transfer function in which an opacity value other than zero is assigned only to data

value v_{iso}. The second type are user-defined surfaces, constructed to mark the result of a segmentation process. This type of surface is in general independent of data values (segmentation can, for example, be performed fully manually).

Surface rendering usually has the advantage of increased rendering speed. The downside, however, is that information beyond the reconstructed surface is discarded. Surface rendering therefore uses only a small percentage of the given amount of data.

The following sections will, in more detail, describe different rendering methods used for virtual endoscopy. Furthermore, optimization techniques will be introduced and examples of systems will be given.

4.1.1 Polygon-Based Surface Rendering

Polygon rendering is probably the most widely used technique for virtual endoscopy. The walls of the investigated anatomical structures are reconstructed as a mesh of polygons (most often triangles). Since polygon-rendering hardware is nowadays part of every graphics adapter, polygons can be rendered at low computational cost. The downside of polygon rendering is that only surfaces are rendered. Additionally, there is an increased amount of rendering artifacts, since the surfaces consist of planar patches. At high magnifications due to perspective distortion, single polygons can become visible.

Polygons are passed on to the graphics card and projected to the screen in a correct way using a Z-buffer algorithm implemented in dedicated hardware.

A polygonal model can be extracted from a volumetric data set using the Marching Cubes technique [LC87] or a variant of this classic method. The marching cubes algorithm iterates through all cells of the data set. In each cell, an adequate number of polygons is created, such that a smooth surface is constructed, separating voxels whose values are greater than v_{iso} from those with a value smaller than v_{iso}.

The number of triangles generated by these approaches is usually large. Several million triangles for large data sets are not uncommon. Straightforward rendering of the triangles may not be possible at interactive frame rates. Therefore, in general, optimization techniques must be applied to reduce the amount of polygons rendered in each frame. The two basic principles are occlusion culling (failing an efficient check on visibility, large parts of the geometry can be discarded) and surface simplification (reduction of the complexity of the mesh by decreasing the number of polygons).

Occlusion Culling

A simple occlusion culling method is *Backface Culling*. All polygons facing away from the view point are simply discarded.

More elaborate techniques were presented in literature:

A widely used technique is to extract the PVS (potentially visible set): In a preprocessing step, the anatomical structure of interest is divided into cells and for each cell, the potentially visible set of polygons is determined. During rendering, only the set of polygons potentially visible from the cell containing the current view point must be rendered.

Sharghi and Ricketts [SR02a] proposed a PVS algorithm for virtual colonoscopy: The colon is divided into sections based on the curvature of the center line of the colon. The boundary between two sections is specified by a portal, i.e., a cut plane whose boundary is the colon wall and which is perpendicular to the center line. They use distance fields (see Section 4.1.2) to find portals of minimum area, i.e., where the colon is narrowest.

Bartz et al. [BS99] used a PVS together with hardware assisted occlusion culling in their VIVENDI system. Additionally they composited the scene in an octree. The bounding boxes of the subvolumes represented by octree nodes are tested, first, whether they contain a part of the isosurface, and second, for intersection with the viewing frustum. All geometry remaining after this step is depth-sorted and passed on to the occlusion culling test. The first 10 percent of octree nodes holding the isosurface are not tested, since they are rarely occluded. The octree-based space subdivision ensures that the system is not limited to tube-shaped structures (like, e.g., colons). They achieved a total visibility culling rate of 90.6 percent, enabling frame rates of more than 10 fps [Bar03b].

Hong et al. [HMK$^+$97] use a decomposition of the anatomy along its center line, therefore the system is limited to tube-shaped structures. The resulting cell/portal structure is used together with a frame buffer-based visibility test to reduce the geometric complexity to about 10 percent. The problem of these algorithms is the expensive preprocessing step.

Hietala et al. [HO00] use sparse ray casting at the beginning of each frame. This method only requires a fraction of all rays that are cast by traditional ray casting methods. The purpose is not to determine pixel values by tracking a ray through each pixel, but to process the view dependent set of potentially visible voxels of the volume. The density of rays ensures that each potentially visible cell is pierced by at least one ray. After the ray casting step, visible cells containing part of the surface are triangulated using the marching cubes algorithm and rendered using graphics hardware.

Higgins et al. [HHP03] presented a visibility culling algorithm for virtual bronchoscopy. First the patient's lung system is segmented and the acquired model of the bronchial tree is divided into branches. During the virtual bronchoscopy, the current view point is used to determine the currently investigated branch (e.g., left bronchus), the parent branch of the current branch (e.g., trachea) and all branches directly derived from the current branch. Three separate tests are applied to determine, which parts of the triangle mesh are to be rendered: First, only triangles from branches that lie in the region determined above (current branch, parent branch, child branches) are included. Second, branches that have no part of their extent within a fixed distance from the current view point are excluded. Third, branches that are not within the viewing pyramid are excluded. The set of visible triangles can be completely reused for a new frame, if the viewing direction changed a sufficiently small amount. Otherwise, in some cases, only the third test must be rerun.

Del Rio et al. [dRBJ$^+$03] applied stereoscopic rendering to virtual endoscopy. In each frame, two images, one for the left and one for the right eye of the user are generated. Both images are rendered from view-points which are slightly translated and rotated to keep the focus point. Since two images are rendered, a naive approach would involve performing visibility culling twice per frame. Here, however, the authors simply use an extended viewing frustum for visibility determination. The extended frustum contains the viewing frusta of both the left and the right eye. Thus the visibility information can be used for the generation of both images.

Surface Simplification

Much research has been carried out on simplifying the meshes of geometric 3D models. Some techniques simply replace the polygon mesh by one with equivalent shape and topology, yet fewer polygons, resulting in an increase in rendering speed and a reduction of storage cost. The disadvantage of this method is that the original model remains lost. In some applications it is, however, of advantage to adapt the *level of detail* (LOD) at which objects are rendered during interaction. This would include, for example, rendering objects which are close to the view point at higher detail than objects far away from the eye. One possibility to achieve this is to store each object or each distinct part of the scene a number of times, at different LOD (*static LOD*). Apart from increased memory consumption, this method has the further disadvantage that popping artifacts may occur when objects change from one LOD to another. Example algorithms include decimation techniques based on vertex removal [SZL92] and topology modification [Sch97]. He et al. [HHK$^+$95] use a volume sampling approach, where the triangles are convolved into a volume to generate scalar values (e.g., scalar value may be distance from original mesh).

There are techniques which simplify the meshes iteratively (i.e., the number of mesh vertices is decreased by one at a time) and store information about each modification. This can then be used to progressively reconstruct the lossless original model. PM (Progressive Meshes) [Hop96] is a well-known method for three-dimensional mesh simplification. A progressive mesh is a linear sequence of increasingly coarse meshes built from an input mesh by repeatedly applying edge collapse operations. It provides a continuous resolution representation of an input mesh and is useful not only for efficient rendering, but also for efficient storage and transmission.

View dependent simplification originated as an extension of the PM technique. Xia et al. [XESV97] organized the PM as a vertex hierarchy (or view-dependent progressive mesh (VDPM)) instead of a linear sequence. A similar approach employing octree-based vertex clustering operations used for dynamic simplification was introduced by Luebke and Erikson [LE97].

4.1.2   Volume Rendering

Ray tracing of volume data was first introduced by Kajiya et al. [KH84] in 1984 and by Levoy [Lev88] in 1988: In this technique, nowadays referred to as *ray casting*, a transfer function mapping data values to colors and opacity is used. Rays are cast from the view point into the volume. Each ray samples the volume at evenly spaced positions. The transfer function is used to convert each data sample to a color and an opacity value. The color and opacity are accumulated along the ray. As soon as the overall opacity of the ray reaches a certain maximum (theoretically a hundred percent), ray casting is stopped (*early ray termination* and the according pixel is assigned the accumulated color.

This basic ray casting technique, when applied to virtual endoscopy, is in general too slow to achieve interactive frame rates on the CPUs of today's standard PCs. One possibility to speed up rendering is to simplify the transfer function and render only isosurfaces, as in the polygon-based approach. Using ray casting to render surfaces is often called *first-hit ray casting*. Based on the original ray casting approach, rays are cast from the view point into

the volume and intersected with the depicted surface. In practice, the ray position, starting at the eye point, is iteratively advanced in viewing direction and the underlying data is sampled. As soon as a data sample exceeds the iso-value v_{iso}, an intersection with the isosurface has been detected. The exact intersection is then calculated and, according to the orientation of the isosurface at the point of intersection, the pixel is shaded. While first hit ray casting is still substantially slower than polygon-based surface rendering, it has the advantage of increased flexibility. If, for example, v_{iso} is changed, the performance of first-hit ray casting as described so far is not affected, whilst a polygon mesh would have to be completely rebuilt.

Most existing algorithmic methods for speeding up the process of ray casting rely on one or more of the following principles:

- **Fast Empty Space Traversal:** Rays often have to travel many steps through the volume until they hit the first opaque voxel. This can take a significant portion of rendering time. Therefore some techniques which aim at accelerating traversal through empty regions of the volume have been developed.

- **Data Reduction (object order ray casting):** Some optimization algorithms divide the volume into subvolumes, identify all subvolumes which contain non-transparent parts of the volume, and apply ray casting adaptively, according to projections of interesting subvolumes. Since rays are not cast pixel by pixel as in the original (*image-order*) ray casting algorithm, but triggered by object locations, this method is often called *object-order ray casting*.

- **Improvement of Memory Management:** Ray casting in large volumes requires a large memory bandwidth. If memory access is organized such that the number of memory fetches causing a stall of the CPU pipeline is reduced, ray casting can be significantly accelerated.

- **Usage of Frame to Frame Coherency:** In interactive volume visualization, two successive frames are often very similar. This similarity can be exploited to speed up the rendering process.

- **Usage of Pixel Space Coherency:** A pixel whose neighbors all have similar color values can be expected to have a color similar to its neighbors. This fact can be exploited in a way that some pixels acquire their final colors not through ray casting, but through interpolation between the colors of neighboring pixels.

The following sections present single optimization techniques which use the concepts outlined above. An overview of early acceleration techniques for volume rendering is given by Yagel [YSW$^+$96].

Fast Empty Space Traversal

A technique for fast traversal of empty space was presented by Levoy [Lev90a]. It is called *hierarchical spatial enumeration*. For a data set measuring N voxels on a side with

$N = 2^n - 1$ for some integer n, a pyramid of $n + 1$ binary volumes $V_{0..n}$ is established. An element in the basis volume V_0 represents a cell (the space enclosed by a cube made up by 8 neighboring voxels) in the original volume. An element in this basis volume is assigned a zero, if all of the voxels belonging to the according data cell in the volume have opacity equal to zero. An element in any volume V_m, with $m > 0$, represents eight elements on level $m - 1$, and is assigned a zero if all those eight elements contain zeros. Ray casting is started at the top level. Whenever a ray reaches a cell that is represented by a volume element containing a one, the algorithm moves down one level, entering the element representing whichever cell encloses the current location. This technique helps, to efficiently leap over empty regions within the data set.

A very common way to accelerate rays through empty space are *distance fields*. Before ray casting is started, all cells in the volume are classified with respect to their data values as to whether being opaque (visible) or empty. A cell is opaque if the integral of the transfer function along the interval spanned by the voxel values of the cell is greater than zero. So in first hit ray casting an opaque cell would be a cell containing at least one data value above and one data value below the threshold. A distance transform algorithm is applied to calculate a *proximity* or *skip* value for each empty cell. This value denotes the distance from the empty cell to the nearest opaque cell. The *skip* value therefore is the distance that can be safely skipped along any ray that samples this cell. This method, of course, includes a quite expensive initialization phase, which has to be repeated every time the transfer function changes. Zuiderveld et al. [ZKV92] were the first to propose the use of the distance transform to accelerate ray casting and Wan et al. [WTK$^+$99] applied this to virtual endoscopy.

You et al. [YHJ$^+$97] introduced a combination of direct volume rendering and polygon rendering. The isosurface is first polygon-rendered. The resulting content of the hardware Z-buffer gives information about the distances from the image plane to the wall through each pixel. A ray can then be started at about this distance from the image plane. The disadvantages of this system are the overhead induced by polygon rendering and the fact, that a change of the transfer function might call for an adaptation of the polygon mesh.

Another interesting technique is the so called Lipschitz method that was introduced by Kalra et al. [KB89] and extended to volume rendering by Stander et al. [SH95]. A function is a Lipschitz function if its derivative is bounded. A mapping between metric spaces $f : D \subset \mathbf{R}^n \to \mathbf{R}$ is a Lipschitz mapping, iff

$$||f(\mathbf{x}) - f(\mathbf{y})|| \leq L||\mathbf{x} - \mathbf{y}|| \quad \forall \mathbf{x}, \mathbf{y} \in D$$

The minimum L satisfying this equation over a given domain D is known as the *Lipschitz constant* of f in D. The CT volume represents a function

$$f : \mathbf{R}^3 \to \mathbf{R}$$

assigning a density value to each point of the volume. The isosurface that should be visualized is the set of points

$$\{\mathbf{x} \in \mathbf{R}^3 : f(\mathbf{x}) = v_{iso}\}$$

for a given iso value v_{iso}. The volume is separated into cubic sub-volumes. The sub-volumes are connected by a hierarchical data structure, e.g. an octree. For each sub-volume

a Lipschitz constant L is calculated and saved. L is equal to the maximum gradient inside the sub-volume. If $\mathbf{x}_0$ is the current position of a ray in space and d denotes the distance from $\mathbf{x}_0$ to the farthest corner of the sub-volume, then the condition

$$|f(\mathbf{x}_0) - v_{iso}| > L \cdot d$$

guarantees that the sub-volume is either entirely interior or entirely exterior with respect to the visualized isosurface. Also it is always possible to move a distance of

$$d = (f(\mathbf{x}_0) - v_{iso})/L$$

in any direction within the current sub-volume before any scalar values could possibly reach the threshold v_{iso}. The Lipschitz method underestimates the distance from $\mathbf{x}$ to the nearest point of the isosurface. Therefore the speedup gained by using this technique is smaller compared to the speedup gained by distance fields. An advantage over space leaping, however, is, that changing the threshold causes no need for repeating any preprocessing.

Vilanova et al. [VGK00] introduced a technique suitable for virtual endoscopy in tubular-shaped anatomic structures (e.g., colon, vessels). The shape of the organ is approximated by a structure composed of cylinders, such that each cylinder is completely inside the organ. This helps accelerating direct volume rendering using the following technique: Each ray can, if its current position is inside a cylinder, directly pass on to the nearest intersection point with the cylinder in positive ray direction without missing any part of the object. The advantage to distance fields (see Section 4.1.2) is that the step taken from a voxel inside the organ is often longer than the distance between the voxel and the nearest part of the organ wall, especially for rays almost parallel to the organ wall. As soon as the ray leaves the approximated organ structure, distance fields can be used for further acceleration.

In a method introduced by vanWalsum et al. [WHVP92], the ray starts sampling the volume in low frequency, taking large steps between sample points. If a large value difference is encountered between two adjacent samples, additional samples are taken between them.

Shargi and Ricketts [SR01] approximated the space inside the investigated structure by spheres and then in the ray casting process, this approximation was used to traverse the space efficiently. Later, the same authors proposed the following strategy [SR02b]: Groups of neighboring rays enter non-transparent material via the same first non-transparent cell. In this method, the distance to the first non-transparent cell is obtained by casting one of the rays from the group. This distance is then used for all other rays of the same group to skip the space and start ray casting inside this non-transparent cell. The group of rays piercing the same cell first is determined using an approximation: The cell is projected to the image plane and rasterized. All pixels covered by the projection footprint form the group. This approximation leads to errors - some rays miss non-transparent parts of the scene. The authors, however, state that this yields only a small image error.

Lakare and Kaufman [LK04] presented a technique in which only a subset of all pixels is filled by casting a ray from the eye point. These rays are called detector rays. For each

detector ray, the location of the first intersection with non-transparent tissue is stored. Casting of neighboring rays is started near this distance.

Object Order Ray Casting

Object order ray casting techniques identify subvolumes which contain all necessary information and cast rays according to the location of these subvolumes. This yields acceleration of the rendering process at the expense of more preprocessing.

Livnat et al. [LSJ96] introduced a technique for iso-surfacing. In this algorithm, subvolumes are mapped to a 2D coordinate system, the so called *span space*, according to their minimum and maximum data values. The horizontal X axis depicts the minimum data value of the subvolume, the vertical Y axis depicts the maximum data value. This is especially useful for iso-surfacing: For a given iso-value v_{iso}, a sub-volume S that has associated coordinates (x, y) in the span space, such that $x < v_{iso}$ and $y > v_{iso}$, contains a portion of the isosurface. That means that all points in the span space that are part of the rectangle ranging from 0 to v_{iso} in X direction and from v_{iso} to the maximum in Y direction represent isosurface cells. A number of techniques have been developed to quickly identify all interesting cells, i.e., all cells belonging to that rectangle. One of them is based on lattice subdivision [SHLJ96]. The data domain is decomposed by subdividing the span space into a two dimensional $L \times L$ lattice. Points in span space are then assigned to the according lattice elements. Sub-volumes represented by points in lattice elements, which are completely enclosed by the rectangle mentioned above, are quickly identified as surface cells. Lattice elements which only overlap the rectangle have to undergo further testing.
Rymon-Lipinski et al. [vRLHJ$^+$04] recently presented a similar technique. They called it the "span-triangle". It is a data structure permitting even faster access to a list of all isosurface-cells.
The biggest disadvantage of the span space and similar techniques is, that by mapping subvolumes to the span space, neighborhood information is lost. If, for example, surface cells have to be processed in order of distance from the eye point, they have to be sorted after being identified.

Li et al. [LWCK99] introduced a technique for virtual endoscopy that they called rendering with sub-volumes. The data set is uniformly divided along each axis. Each sub volume has a 50 percent overlap with its nearest neighbors in all three directions. The overlapping increases the probability, that all visible parts of the iso surface are located in the same sub-volume. First only this sub-volume is rendered. Only if pixels of the image remain unprocessed other sub-volumes have to be computed. The major drawback of this approach is the data duplication. The method requires up to 8 times the storage due to the 50 percent overlap.

Improvement of Memory Management

Law and Yagel [LY96] presented multi frame thrashless ray casting with advancing ray front. The data set is divided into sub-volumes consisting of n^3 J density values. Each ray is marked with the cell it initially enters and is linked to a list of rays waiting for this cell to become active. During the ray casting process only one cell is active at a time. All rays waiting for the active cell are advanced until they exit the cell. This implies that cells once used will not be accessed again before the next frame. So the phenomenon of thrashing is guaranteed not to occur, a fact, which may reduce rendering time for large data sets.

Novins et al. [NSG90] use a similar method to reduce the number of cache misses. Here at least two slices of data (one slice of data consists of all data values that have the same Z coordinate) are required in memory at a time. A pair of slices is called a *slab*. The system maintains a *slab queue* of rays waiting for each slab. Regardless of view direction and perspective, all rays coming from the eye point must fall into one of three categories: First, the ray may stay in the Z plane containing the eye point, thus requiring access to only one slab. Second, it may diverge in the positive Z direction from the eyepoint. In this case it requires access to the slab containing the eye point and to slabs with successively larger Z values, in order. The third case, when the ray diverges in negative Z direction, is analogous. If the principal viewing direction is not close to parallel to the Z axis, X or Y slices can be used instead of Z slices. Slabs are only loaded into memory or cached, if they are really needed.

Recently, Grimm et al. [Bru04, GBKG04, Gri05] presented a ray casting system which is predominantly accelerated by a cache efficient memory layout for volumetric data and highly efficient approaches for addressing and processing it.

Exploitation of Frame to Frame Coherency

Much of the work invested to produce one image may be used to expedite the generation of the next image. Yagel and Shi [YS93] presented a method which exploits coherency between consecutive images to shorten the paths rays take through the volume. The algorithm starts by projecting the volume into a coordinate buffer, which stores, at each pixel location, the object space coordinates of the first non empty voxel visible from that pixel. For each change in viewing parameters the buffer is transformed accordingly.

Bishop et al. [BFMZ94] extended the idea of keyframe-less rendering to even frame-less rendering. Their system relies on randomized immediate pixel update. All pixels on the screen are ordered randomly and updated in that order. When the viewpoint moves, new pixels are rendered already according to the new camera parameters. So the image becomes, for some time, a mixture of past and present images. If the changes in the scene are large, this mixture will be quite confusing. In order to compensate for this, the display system might switch to lower resolution or to double buffering during rapid scene changes.

Usage of Pixel-Space Coherency

Usually there is a high coherency between pixels in image space. That is, it is highly probable that a pixel located between pixels of identical or similar color will acquire the same (or a similar) color. Therefore possibly one could avoid sending a ray through that pixel. Levoy [Lev90b] developed a method for adaptive image super-sampling for volume rendering: The image plane is divided into square sample regions measuring n pixels on a side, and rays are cast from the four corner pixels of each region. If the range of colors returned by the four rays in the sample region is less than some value ϵ, no further processing is performed on that region. Otherwise, the region is divided into four subregions and more rays are cast. Subdivision continues until the range of colors falls below ϵ or the size of the region reaches some specified minimum. When all sample regions have been processed, the available color values are interpolated to form the final image. Resulting errors are corrected by progressive refinement in subsequent frames, if the view point remains still. A similar approach is presented by Bergman et al. [BFGS86].

Koloszar and Jae-Young [KJY03] start casting rays at a reduced resolution. A pixel located between two already rendered pixels can be interpolated if two conditions are met: Interpolation is safe if both already rendered pixels are *bright enough*, meaning that the surface region being rendered is close to facing the camera. The second condition checks the difference in depth between the two already rendered hit location. If they differ too much, interpolation is deemed unsafe. If either check fails, the pixel in question is shaded from scratch.

4.1.3 Other Volume Rendering Techniques

Splatting [Wes90] is an object-order approach, projecting voxels to the image plane. The accumulated footprints on the image plane represent the visualization. Each voxel generates a *Gaussian splat*. This means that the pixels inside a certain region (the kernel) around the projection point obtains color and opacity from the voxel, with opacity decreasing with distance from the center, according to the Gaussian distribution. The kernel shape and size is critical for the quality of the result. The accumulation, or compositing process, can be done back-to-front, which guarantees correct visibility. Front-to-back compositing is faster, because the process can be stopped when pixels are fully opaque.
Though splatting is of less significance in virtual endoscopy than ray casting, it has been reported to be used [YSW$^+$96].
Laur and Hanrahan [LH91] presented a technique based on a pyramidal volume representation to accelerate splatting. An octree is fit to a pyramid satisfying a user-defined precision. The octree is then rendered using a set of splats each scaled to match the size of the projected octree node. This allows a progressive refinement of the image according to the desired tradeoff between quality and speed.

The probably fastest software volume rendering to date is the shear-warp algorithm [LL94]. The technique works as follows: A base plane is chosen corresponding to the face of the volume data that is most parallel to the viewing plane. The volume is divided into slices parallel to the base plane. These slices are sheared according to the factorization

of the viewing matrix. Afterwards, the slices are projected to the base plane using a back-to front or front-to-back compositing. The projection is performed using bilinear interpolation. The base-plane approach and bi-linear interpolation can result in poor image quality, if the resolution of the final image is significantly bigger than the volume resolution. Thus it is generally not suited for large magnification factors as encountered in virtual endoscopy.

4.1.4 Hardware-Assisted Volume Rendering

All techniques presented so far have the potential to make volume rendering faster, but they fail to make it work at interactive frame rates for reasonable image resolutions. Real-time perspective volume rendering has so far only been achieved using hardware acceleration.
To overcome the problem of reduced rendering speed, some systems use an excessive amount of processing units. You et al. [YHJ$^+$97] used a 16 processor SGI Challenge for parallel rendering of isosurfaces in 1997. In 1998, Parker et al. [PSL$^+$98] achieved interactive (more than 10 fps) isosurfacing of the Visible Female-Data Set using 64 or more processors.
Another way to make use of hardware-acceleration is to make use of the steadily increasing set of features in nowadays general-purpose graphics hardware, powered by the computer game industry. Vining et al. [VSA$^+$97] were among the first groups using texture-mapping-based volume rendering [CN93] for virtual endoscopy in 1997. At that time, however, the capabilities were strongly limited by the size of the available texture memory and by the lack of shading. Gobbetti et al. [GPZT98] used the 3D texture mapping hardware abilities of high-end graphics systems in 1998 for perspective volume rendering in virtual endoscopy. Again, the lack of shading significantly reduced the visual quality. In 1998, Westermann and Sevenich presented 3D texture mapping-based volume rendering with isosurface shading [WS01]. Due to the nature of the algorithm, perspective rendering can be achieved at no additional cost. Meißner et al. extended this approach with advanced clipping methods and shading of volumetric data. Texture mapping-based volume rendering approaches have since then been subject to a lot of research [HBH03]. Recently, Scharsach et al. [Sch05] presented a system for hardware-accelerated real-time volume ray casting in virtual endoscopy. The problem of texture memory being too small for keeping the full data set can nowadays be efficiently solved by bricking techniques.
Apart from this, special-purpose hardware with built-in volume renderers has been introduced.
The VolumePro board (TeraRecon, originally developed by Mitsubishi Electric Research Labs, MERL) presents a hardware implementation of ray-casting. It provides real-time rendering with compositing, classification with density based transfer functions and Phong shading. Unfortunately the VolumePro does not support perspective projection which is mandatory for endoscopic explorations. Vilanova et al.[VWKG01b] proposed to decompose the volume in view direction into several slabs, each containing several volume slices. These slabs are one-by-one rendered by VolumePro and warped to simulate a perspective projection. These methods suffer from reduced quality. If the number of slabs is too low, the image quality suffers badly from the error introduced by the parallel projected slabs and the warping.
VIZARD II [MKS98] is a special purpose PCI card on which a perspective volume rendering algorithm is implemented. Rays are cast through a volumetric, possibly non-uniform,

regular data set. To ensure high image quality, sampling is freely selectable in each dimension. Different rendering modes, such as MIP (maximum intensity projection), unshaded volume rendering, etc. are supported.

4.1.5 Image-Based Rendering

Image-based rendering is an approach in which pre-rendered (or captured) source images are used to compose a virtual scene. Since image-based rendering replaces the 3D representation of the scene with source images, it usually substantially increases rendering speed, but at the expense of decreased flexibility. The image synthesis time is independent both of the complexity of the scene and the (pre)rendering technique used. Expensive (volume) rendering can be applied, adding cost only to the preprocessing step. On the other hand there is the disadvantage that information not contained in any of the source images cannot be depicted during interactive navigation.

The simplest (but a widely used) method for image-based rendering is pre-specifying a fixed path, preprocessing all views obtained when moving along this path, generating an animation and providing only Video Cassette Recorder (VCR)-like functionality [Bar03b].

QuicktimeVR [Che95] is a method mainly used for architectural visualization. This technique uses 360-degree cylindrical panoramic images to compose a virtual environment. Variants of this approach have been applied to virtual endoscopy:

Wegenkittl et al. [WVH$^+$00] map source images to a cube, rather than a cylindrical shape. In a preprocessing step, the cube is moved in small steps along a predefined path and after each step along the path, the scene is projected to each of the six faces of the cube. The images are stored in a video format. During interaction, the user can arbitrarily move along the predefined path, the images on the cube faces are adjusted accordingly. The camera can be rotated to an arbitrary direction. In each frame, the six images are warped to the screen, according to current camera orientation. Adjustable semi-transparency can be achieved by generating more than one video for each cube face. The videos then can be arbitrarily blended onto each other.

Later, Vos et al. [VSvG$^+$01] presented a similar approach. In their system, the viewing cube (consisting of the six prerendered images for the current view point) is unfolded and rendered completely, presenting a full 360-degree omnidirectional field of view. The image presenting the view ahead is placed in the center. Tiede et al. [TvSGSH02] proposed yet another similar technique: Additionally they use the built-in distortion correction of the QuickTime player to map cubic panoramas to spherical ones.

The keyframe-less rendering method introduced by Qu et al. [QWQK00] is a combination of volume rendering and image-based rendering. *Key frames* are frames which are fully rendered by conventional rendering techniques and thus exhibit high quality. Frames that are not key frames are created by interpolation between key frames (e.g., for MPEG-video compression). Keyframe-less rendering means, that apart from the first image in a sequence, no further frame is completely rendered by the high quality but time consuming rendering technique, e.g. ray casting. Instead, new frames are created by warping the final image of the preceding frame according to the change in viewing parameters. Only those parts of the screen that are left blank after the warping procedure are filled by pixelwise rendering. To improve the quality, the frame generation process often is enhanced by a third step in

which some other pixels are selectively rendered. This reduces one of the problems of this approach, the non-uniform image quality among frames.

4.1.6 System Examples

The VIVENDI system [BS99] of the University of Tübingen and the VICON system of SUNY Stony Brook use polygon rendering assisted by occlusion culling.

FreeFlight [VSA$^+$97] of the University of Wake Forest provides both polygon rendering and hardware-assisted volume rendering. The VirEn system [VKG99] of the Vienna University of Technology uses polygon rendering. When the view point remains still, the image is enhanced using volume rendering. Yagel et al. [YSW$^+$96] used splatting in a system for the training of surgeons in virtual sinus surgery and for teaching in virtual endoscopy.

The V3D viewer of Viatronix uses optimized first-hit ray casting accelerated by space leaping. The rendered surface can be made semi-transparent, using volume rendering. In that case, however, the frame rates break down [Bar03b]. Syngo by Siemens Medical Solutions uses ray casting accelerated by space leaping. Easy Vision by Philips Medical Systems uses first-hit ray casting at lowered image resolution during interaction and at full resolution for the final image [Bar03b]. The VoxelView system, developed in the mid-nineties by Vital Images, applies volume rendering in an offline process and, during interaction, presents only a video animation [Bar03b]. The 3D Slicer software of the MIT uses polygon rendering which can be accelerated using surface simplification which, however, reduces the accuracy of the surface model [Bar03b].

4.2 User Interface

Designing the user interface of a virtual endoscopy application is a challenging task. This includes, first of all, the provision of intuitive and easy-to-use navigation of the virtual endoscope through the area of interest. Section 4.2.1 introduces various navigation paradigms. A second important aspect of the user interface are navigation guides pointing out the position of the virtual camera, the connection between the 3D image and original data and, possibly, proposing further directions of movement. Section 4.2.2 deals with navigation aids.

Tracking allows shifting the user interface to the physical endoscopic devices intraoperatively. The position of the endoscope camera is determined and transferred to the virtual endoscopy. Real and virtual endoscopy can thus be synchronized. Section 4.2.3 deals with tracking.

To utilize virtual endoscopy for training purposes, it is important to provide a user interface that is close to reality, without working on a real patient. Section 4.2.4 introduces systems and techniques for simulation of endoscopic surgery.

4.2.1 Navigation Control

Besides rendering quality and speed, the camera navigation paradigm used determines the usability of the system. Controlling a virtual camera in a virtual examination and getting

useful information out of it is an important and challenging task.

Various approaches to implement camera navigation have been introduced so far. They can be roughly classified into three types: automatic navigation, guided navigation and free navigation [Bar03b].

Automatic Navigation

In automatic navigation, a camera path is generated semi-automatically or automatically. Then an offline animation of a fly through along this path is generated and presented to the user [HKW+95]. The biggest disadvantage of this scheme is that interaction is reduced to VCR-like functionality. An example system where key-framed animations are rendered offline is given by Hong et al. [HKW+95]. If the generated video is not sufficient to achieve the desired result (e.g., to obtain a correct diagnosis), the preprocessing step must be repeated, or, in a costly refinement step, the generated animation must be adjusted. With regard to using this paradigm for virtual colonoscopy, Dachille et al. [DKW+01] noted that only approximately 70 percent of the colon surface are viewed during a single direction fly-through and up to 95 percent for a bi-directional fly-through. To view remaining parts of anatomy, interactive camera control is required. The advantage of automatic navigation is that, since rendering is performed off-line, rendering needs not be interactive, thus expensive volume rendering techniques can be used [HKW+95].

Free Navigation

The other extreme is complete manual or free navigation. Here no constraints are posed upon the user's movement, leaving the path and actions to take completely up to them. Unfortunately it is often very hard to navigate through complex anatomical structures. One way to reduce this problem is to provide navigation aids (see Section 4.2.2). Alternatively one can use a guided navigation scheme, as discussed in the following paragraph.
Gobbetti et al. [GPZT98] use free navigation accompanied with a force/friction model, based on original volume data. If opaque parts of the scene are entered, the model detects collision and the respective collision detection reaction is calculated.
A completely different free navigation paradigm was introduced by Haker et al. and Vilanova i Bartroli et al. [VWK+01, VWKG01a]: The shape of the organ of interest (here: the colon) is unfolded (flattened) and converted into a two-dimensional height field which can be rendered interactively using standard visualization techniques (here: polygon rendering). Thus the lumen can be interactively examined without traversing it.

Guided Navigation

Guided navigation enables (more or less) flexible navigation, combined with user guidance, e.g., enabling the user to return to a predefined path, whenever they like. Different kinds of guided navigation have been proposed, some adding constraints to navigation, some simply adding guidance. Similar to free navigation, the usability of guided navigation requires

the application of a fast rendering technique. An example for guided navigation in virtual colonoscopy was given by Hong et al. [HMK$^+$97]. Here the camera is mounted on a virtual submarine, which is immersed within a potential field and moved according to a set of kinematic equations. With this physically-based model, the interior of the investigated part of anatomy (here: the colon) mimics a water tunnel, with the water running downstream (here: away from the rectum). The submarine is moved by the water flow as well as by any external influence imposed by the physician, but never collides with the tunnel walls. The potential field is calculated based on two distance fields. The first one is called *DFS* (distance from the start) encoding the distances of voxels from a user-specified start point, following the colonic lumen. The other distance field is the *DFB* (distance from the boundary), encoding the distance between each voxel and the nearest point on the colon surface. The DFS-distance field encourages movement from the start point towards the end point of the path through the colon. The DFB-distance field prevents collision on the colon wall by driving the camera away from the surface. Per default, the camera follows the center-line of the colon, derived from the DFB-distance field. Embedded interactive modes allow users to deviate from the path towards the surface for detailed inspection of structures of interest. When returning to automatic navigation, the virtual camera is gently pushed away from the colon wall back to the center flight path. The VIVENDI system of the University of Tübingen [BS99] adopted the same concept.

Wegenkittl et al. [WVH$^+$00] introduced a more constrained approach for guided navigation. The camera may move only along a predefined path, but can be rotated into arbitrary directions. Haigron et al. [PEO$^+$04] presented an active navigation approach in which the motion of the virtual camera is driven by visual perception during the navigation. Camera navigation combines the on-line computation of the scene view and its analysis, to automatically define the three-dimensional sensor path.

Perchet et al. [PFP04] present a guided navigation technique for virtual bronchoscopy. The user can either select a bronchial branch which is subsequently explored, or select a target point. In both cases, the system automatically detects a path, based on distance fields, and follows it.

System Examples

Most current commercial systems provide both free and either guided or automatic navigation. The VoxelView/Vitrea2 system by Vital Images (VI) uses a navigation mode which steps slice by slice through the investigated cavity [Bar03b]. This conservative way of navigation is motivated by the traditional way radiologists examine data.

Path Detection

Both automatic and guided navigation require the definition of a path through the investigated anatomical cavity. A possible way is to automatically extract the center line of the cavity and use this as the path. Thinning algorithms [VKG99] iteratively peel off the object boundary layer by layer until one central layer remains - the center line. The challenge associated with this technique is to keep the connectivity of the object. So-called *simple points*

whose removal would affect the object topology must be detected [MS96].

Some algorithms are based on Voronoi diagrams. The edges of the Voronoi diagram are the set of points equidistant to at least two surface points, therefore the medial axis of a polygon is a subset of its Voronoi diagram. This can be generalized to 3D objects: The center line can be extracted as a subset of the Voronoi diagram [OK95].

The VESA system by General Electrics (GE) uses algorithms from robot motion planning to generate a flight path through the reconstructed anatomy [Bar03b].

Some algorithms use a geodesic distance map [HDL$^+$01]. In this map, each object voxel is associated with its geodesic distance to a source point indicated by the user. In this sense, the geodesic distance is the distance along the shortest legal path, i.e., a path not colliding with the surface of the investigated lumen. The 3D skeleton is obtained by extracting the minimal path between each end point of the 3D object and the source point, by following the gradient of the distance map. Perchet et al. [PFP04] introduce such an algorithm and point out that one advantage of these techniques is the effective branching detection method, resulting in quick identification of the branches of an object, especially useful in virtual bronchoscopy (see Section 4.3.4): Voxels that have the same geodesic distance are grouped in a cluster or *propagation front*. Successive clusters are considered starting from the source point and extending towards the end points. A branching point is detected as soon as one of the clusters splits.

There are algorithms which extract paths, guaranteeing that any part of the surface of the investigated structure is visible from at least one point along the path. One such algorithm was presented by He et al. [HHCL01].

4.2.2   Navigation Aids

Navigation aids are essential to give the user information about the current location of the view point and camera orientation.

A simple but effective method is to indicate both view point and viewing direction in relation to original volume data, e.g., via a multi-planar reconstruction (MPR) or simply original slices [LKS01]. An example for this is given in Sections 7.1 and 7.2.

Some systems provide three-dimensional maps, volume rendered overviews of the complete scene, and indicate the view point and direction and, possibly, a predefined camera path in that overview [LKS01].

Wan et al. [WDK$^+$01] presented a system in which the center line of the investigated lumen is depicted in the 3D scene.

Better navigation can be also achieved by a better understanding of depth. One possible tool to enhance depth perception is depth shading (see Section 6.1.2). Some researchers also examined the use of stereoscopic display in virtual endoscopy yielding a real 3D impression [dRBJ$^+$03, DGS$^+$02].

Serlie et al. [SVvG$^+$01] describe how navigation is improved by providing a 3D view not only into viewing direction, but also for views back-, side-, up- and downwards.

The display of presegmented anatomy can provide valuable information facilitating navigation (see Section 7.2) [BSG$^+$01].

Views displaying the unfolded surface of the structure of interest can be used to indicate camera position and orientation [VWK$^+$01, VWKG01a].

In order to facilitate navigation, some systems allow highlighting of parts of the anatomy already seen, indicating to the physician both the path taken so far and parts of the surface yet to be examined.

4.2.3 Tracking

The ultimate goal of *image-guided endoscopy* is to enable the endoscopist to navigate through both real and virtual anatomy at the same time, using virtual endoscopy to gain information not available from views generated by physical cameras. To achieve this, it is essential to allocate and visualize where the surgical tool is in the body at any time during the surgical procedure. There are two ways of how to achieve this: The first one is to track the position of the endoscope using special-purpose hardware. Examples include magnetic tracking and tracking based on optical markers. The latter can only be applied if stiff endoscopes are used, since the optical markers, logically must remain outside the body and endoscope deformations cannot be detected optically.

The alternative way is camera calibration, i.e., modeling of the relationship between the 2D images and the 3D world such that an image of a virtual 3D scene can be mapped to a certain camera position and orientation. There are two aspects of camera calibration: calibration of the intrinsic parameters of a camera, including opening angle, focus and distortion, and, on the other hand, the extrinsic parameters, including the position and the viewing direction of the camera.

Intrinsic parameters include lens distortion (barrel distortion) which causes straight lines in the scene to appear as curves in the image. This is often accounted for in the camera model by adding some distortion parameters [WCH92]. Barrel distortion is usually significant in endoscope cameras due to their small focal length. The distortion parameters can be estimated in the captured images, e.g., by identifying depicted structures which should appear straight in an undistorted image and finding the parameters which straighten these structures [AF00]. This process is sometimes called 'pre-calibration'. Then all images can be undistorted before calibration proceeds. With the camera moving around, only extrinsic parameters are adapted, therefore the pre-calibration remains valid. However, if the focus/zoom of the endoscope camera is adjusted, the extrinsic *and* intrinsic parameters of the camera are changed which complicates calibration. The calibration of zoom-adjustable cameras usually involves first calibrating a conventional static camera model at a number of lens settings spanning the control space of the lens. To accelerate this calculation, pre-calculated lookup tables can be used [FSHM02].

Dey et al. [DGS$^+$02] track the position of the endoscope using fiducial markers. Virtual endoscopy is performed additionally to real endoscopy. The real endoscopic textures, taken from the endoscopic video are mapped to the polygonal surfaces generated from pre-operative radiological images. Thus, the acquired endoscopic texture remains permanently mapped onto the surface even when the endoscope has moved to a different location. This allows virtual endoscopy with realistic texture and with increased diagnostic value after the physical endoscopic session is finished. The method also allows generating stereoscopic virtual views and therefore a real 3D impression which is not acquired from real endoscopy.

The authors state that navigation after the procedure is also a valuable tool for training of endoscopy-assisted surgical procedures.

Other groups conducted similar work: Konen et al. [KST98] mapped anatomical landmarks derived from pre-operative MRI images to live video sequences acquired with a tracked endoscope. Jannin et al. [JBS$^+$97] merged microscope images onto surfaces reconstructed from radiological images for enhanced visualization. Clarkson et al. [CRK$^+$99] registered endoscopic video to tomographic images by maximizing mutual information (see Section 5.1) and texture-mapped 2D stereoscopic video images onto reconstructed surfaces.

Higgins et al. [HHP03] proposed another method performing live endoscopic video tracking using image registration.

4.2.4 Surgical Simulation

More and more open surgical interventions are replaced by minimally invasive endoscopic procedures. This increases the acceptance among patients and reduces deleterious effects, leading to shorter hospitalization times. Endoscopes are inserted through natural openings of the body (e.g., the nose) or small perforation holes (e.g., in the abdomen) rather than relatively large cuts as in open surgery. The price of achieving these advantages is paid by the surgeon who loses direct contact with the operation site. The necessary visual information is mediated and limited by a specialized camera. It is presented on a screen rather than on the immediate site of surgical activity, which is counterintuitive to normal hand-eye coordination. Furthermore, today's endoscopic images are in most cases still monoscopic, i.e., the surgeon must work without real three-dimensional understanding of the site of activity. The constraints imposed by the external control of the surgical instruments reduce haptic feedback and manipulative freedom. This way even simple tasks such as cutting or knot tying pose unusual challenges for an inexperienced surgeon. It is therefore clear that an extensive amount of training is required until a surgeon can perform interventions on human patients.

Traditional ways of training are training on cadavers or animals or the usage of training devices involving synthetic tissue mounted in rigid cases. The latter is suboptimal for training, since conditions remain the same for all training procedures, significantly limiting the training effect. Cadaver training is costly and suffers from a limited availability. Moreover, it is most likely that a trainee will not see rare anatomic variations or perform revision surgery on a cadaver. A normal variant without disease will be found in most of the specimens, the tissue is often changed by formalin preservation, and no bleeding occurs. Training on animals is of limited use due to the differences between animal and human anatomy.

In 1997 Hochberger and Neumann [HNea97] presented the 'Erlangen Biosimulation Model' which was commercialized as the 'Erlangen Endo-Trainer'. The Erlanger Endo-Trainer offers a large spectrum of training possibilities in endoscopic techniques using life-like biological specimens. In 2004, a simplified version, the compactEASIE, was introduced by Hochberger et al. [HEN$^+$04]. In this system, and other animal-part simulators, endoscopy is performed in animal organs harvested from slaughterhouses. These are not expensive but need more effort for preparation and cleaning and are therefore not practicable or accessible for smaller hospitals.

Virtual endoscopy is increasingly recognized as a training tool for inexperienced surgeons. A number of virtual reality (VR)- based endoscopic simulators have been realized in the past few years. Although complete realism can not yet be achieved, these tools are increasingly accepted as additional devices to bridge the gap between basic training and performing actual interventions on patients.

More and more virtual reality simulators of endoscopic surgery are produced and evaluated. Virtual reality-based endoscopic simulators allow for unlimited numbers of procedures with a single system and a single (virtual) patient. Every anatomic variation can be simulated or retrieved from radiological data. Furthermore, special conditions like a postoperative situation with missing landmarks can be included in the virtual environment.

In contrast to the application of virtual endoscopy for diagnosis or preoperative planning, an educational objective focuses more on the visual quality demonstrating the general topological and geometric aspects of patient anatomy and the realistic user-interface, mimicking a real physical procedure, while the accuracy of fine details is only of limited importance, if the details are not subject of the examination.

Most systems require powerful workstations, however there are also tools working on low-end personal computers. Besides academic research projects, also first industrial products have emerged (Musculo-Graphics, HT Medical, Virtual Presence, Voxar, Boston Dynamics,...).

Székely et al. [SBD$^+$00] name the following essential technological components of a virtual reality surgical trainer, the following paragraphs introduce these requirements in more detail:

1. Construction of the underlying anatomical model

2. Visualization of the operation site

3. Modeling organ behavior

4. Force feedback for mediating haptic sensation

Anatomical Model Building

Building a model of the site of application of surgical activity involves creation of a computational representation of the involved anatomical structures. To achieve this, anatomical structures of interest (e.g., radiological data sets, the Visible Human data set) must be recorded, mostly using radiological scanners. Structures of interest might have to be segmented.

Visualization of the Operation Site

A lot of work has been done in the simulation of appearance of anatomical structure, e.g., generation of organ-specific texture. All structures of interest are covered by some microstructure that gives visual information about the type of the tissue and its relative smoothness or coarseness. Realistic texturing not only increases overall realism, but serves also

as a cue to space perception. In a simulator which has the objective of improving the trainee's diagnostic skill, exact simulation of textures can be of significant importance. Organ specific texturing can be accomplished, for example, by mapping real endoscopic imagery [DGS$^+$02] (see Section 4.2.3) or by interpolation based on the Visible Human data set [SBD$^+$00]. Gargalowicz and Ma [GM85] introduced the following technique: As a starting point, a statistical description of a small texture sample taken from real endoscopic imagery is computed in an analysis phase. In a subsequent optimization step, a three-dimensional texture block, initially consisting of white noise is adjusted until the statistical description of its texture is similar to the statistical description of the sample texture. This method suffers from the shortcoming that it does not take into account the low-frequency variations that often occur on organ surfaces. Székely et al. [SBD$^+$00] use a method based on L-systems. Their method is aimed at simulating the biological growing process of blood vessels. The procedure is based on simplified models of angiogenesys. The formation of new vessels is triggered by the perfusion demand of growing tissue as well as physical laws of blood circulation. Additionally to visible blood vessels along organ surfaces, characteristic fluctuations of tissue color are mapped, yielding realistic organ appearance.

Modeling Tissue Deformation

Realistic simulation of tissue behavior during endoscopic and other medical interventions is one of the most challenging research areas in medical visualization. Various models have been proposed. Most current methods, however, work on a simplified representation of the input data. Often, deformation is performed on meshes (e.g., triangle meshes, tetrahedral meshes), rather than directly on the data volume. Such methods include, for instance, freeform deformation techniques from computer graphics. These methods use parametric interpolative models (e.g., superquadrics) for deformation estimation of surface primitives [BGG98]. While the analogy to physical deformation processes is not always obvious, these techniques are popular due to fast calculation of the deformation result. Different simple physically inspired approximations have been used, including space-filling spheres [SHE$^+$98] or the ChainMail algorithm [Gib97] (see section 7.4.9).
Elastically deformable surface models [TPBF87], as used also in computer vision for image segmentation, calculate surface deformation by solving linear elasticity equations. This allows tissue deformation based on physical principles.
A commonly used technique for soft tissue deformation is finite-element modeling
(FEM) [ZT94]: A body is subdivided by a finite number of well-defined elements (such as hexahedrons, tetrahedrons and quadrilaterals). The motion of material points is calculated by solving a discrete system of differential equations. The major drawback of FEM is its high computational cost.
Another popular technique are *mass-spring models*, used for example by
Çakmak et al. [cK00]. Mass knots on the object surface and inner mass knots are connected with each other by virtual springs and damping elements. Motion of single mass points is determined by solving the Lagrange equation of motion, a second order differential equation.
An approach similar to the Mass spring model was presented by Teschner et al. [THMG04]. In this approach, tissue needs to be represented by a tetrahedral mesh. Three distinct forces

based on potential energies at the mass points are derived. These three forces attempt to preserve distances between mass points, preserve the surface area of the object and preserve the volume of each tetrahedron. The properties of the material are governed by weighted stiffness coefficients. The computational cost of this approach is comparable to the one associated with mass-spring models. Both techniques allow interactive deformation on standard hardware.

Haptic Feedback

Application of force-feedback requires special hardware capable of force generation. There is a wide variety of such hardware devices, such as the GROPE teleoperator master arm, force-feedback joysticks, 3-D pointers or exo-skeletons.

A force-feedback joystick is a normal joystick, as used for computer games, which can exert force via its handle, supporting or counteracting the user's maneuvers. For an example application see Section 7.4.

Many groups, including Székely et al. [SBD$^+$00], use the commercially available PHAN-TOM device (SensAble Devices). A Phantom device is a robot arm that is attached to a computer and used as a pointer in three dimensions. It allows movement in six degrees of freedom. Motors allow the Phantom to exert force on a user's hand. The cheapest model exerts force in the X, Y, and Z dimensions; other models can also exert torques. The forces give the feeling of interacting with virtual objects in three dimensions. An overview of alternative tools and approaches is given by Baumann [Bau97].

Examples of Systems

At the Karlsruhe Research Center, an endoscopic surgery trainer for minimally invasive gynecological and laparoscopic surgery was developed [cK99, cK00, KcM00]. A 'phantom box' with real surgical instruments and a virtual endoscopic camera are used as a human-computer interface. Three-dimensional models of the instruments are graphically displayed in the virtual scene. The elastodynamic behavior of deformable objects and the interactive manipulation of deformable objects are simulated using a mass-spring model. A smoke module for visualization and animation of smoke and steam was added, based on procedural 3D textures and spline-key-frame animation techniques. The endoscopic camera lens will steam up, dependent on the density of the smoke and the duration of coagulation. A force-feedback device provides intuitive interaction with soft tissue. Injuries of arteries activate the pulsating bleeding simulation. Bleeding and irrigation are simulated using particle systems. Particles, rendered as transparent, textured spheres with motion blurring, are moving according to Newton's laws of motion.

The 'Nasal Endoscopy Simulator' (NES) [BMV$^+$99] was developed in a cooperative project of the Technical University Darmstadt and the University Hospital in Mainz. It provides training of endonasal sinus surgery using three different levels. In one level, hand-eye coordination can be trained by navigating the virtual rigid endoscope through an abstract virtual environment. In the next level, some diagnostic tasks with 3D models of the nasal

cavities and sinuses can be performed. The trainee navigates through the virtual situs, identifying landmarks in some exemplary physiologic and pathologic cases. An electromagnetic tracking system is used to register the position and orientation of the endoscope. Angled optics are available. The third level gives the trainee the opportunity to use surgical instruments to perform a therapeutic intervention on the virtual situs.

During simulated interventions, trainee errors, such as collisions of instruments and high sensitive tissues, are detected and evaluated. The system provides a real-time simulation of the deformation behavior, based on coarse meshes. To build up a database of exemplary cases, tomography data of interesting patients are collected at the University Hospital in Mainz.

The GI-Mentor system by Simbionix is a commercial virtual reality endoscopy simulator allowing simulation of both upper and lower endoscopic diagnostic and therapeutical procedures including an authentic endoscope, a torso (a physical installation mimicking the patient body) and a customizable training program. The system offers force feedback realized by pneumatic brakes acting on the tube of the endoscope. Ferlitsch et al. [FGG+02] assessed the training effect of this system. In this test, a total of 13 beginners and 11 experts were included. Seven beginners were randomly allocated to a training group whose members were allowed to practice using the simulator. The skills of both experts and beginners were tested both before and after the training process. The authors state that at the end of the training process, differences between the experts and the training group were no longer seen.

The Minimally Invasive Surgery Training-Virtual Reality (MIST-VR) trainer of the Medical College of Virginia proved effective in training the novice in basic laparoscopic skills [AMKD02].

At the universities of Mannheim and Heidelberg a virtual reality simulator for flexible endoscopy has been developed [KM02]. Flexible endoscopes are used, for example, in gastroscopy and colonoscopy. For navigation, the doctor can bend the tip of the endoscope in two orthogonal directions by rotating small wheels attached to the head of the endoscope. The limited view during flexible endoscopic procedures forces the physician to use haptic cues for orientation. When passing flexures, for example, the tip of the endoscope is bent due to the collision with the organ surface and thus a force can be realized at the navigation wheels. This force is generated by a motor. If the doctor applies movements against a resistance, e.g., soft tissue, force feedback can be exerted using brakes to hamper wheel movement. Since these forces are active, i.e., artificially generated, they can be intuitively translated to virtual endoscopy. The simulator uses two force feedback devices, one for rotation of the tip of the endoscope, simulating the navigation wheels and one for global movements of the endoscope. The latter is realized by attaching the tube mimicking the endoscope to a toothed drive belt which is attached to a motor.

Future Outlook

Ferlitsch et al. [FGG+02] predict that endoscopic simulators will advance to focus on training of special but crucial situations. It will be possible to simulate advanced and demanding

after-hours procedures, such as emergency endoscopic hemostasis or colonic decompression, stent implantation and transmural pseudocyst drainage. These situations are currently hardly ever trainable, due to a lack of availability for scheduled training sessions. The training scenarios could also be enriched with critical incidents such as expulsion of air, unconscious movements, screaming, oversedation, apnea and cardiac complications.

4.3 Applications

This section gives an overview of fields of application of virtual endoscopy. Another detailed review of virtual endoscopy applications is given by Rogalla et al. [RvSH02].

4.3.1 Virtual Colonoscopy

Colorectal cancer is the third most frequently diagnosed cancer world-wide and represents the second leading cause of cancer death in most industrialized countries. Early diagnosis and preventive removal of pre-malignant adenomatous polyps are believed to be the most probable reasons for a gradual decrease in both incidence and mortality within the last twenty years. The five-year survival rate for colon cancer is nearly 90 percent for localized disease versus about 6 percent for distant metastases.

It is generally accepted that the vast majority of colorectal cancers arise from adenomatous polyps. Therefore early detection of colorectal polyps may be regarded as one of the major goals in health care management.

Polyps may be found in about 25 percent of the population by the age of 50 years. The incidence increases with age. Malignant transformation (i.e., from polyp to tumor) takes between 10 and 15 years and is found in approximately 1 percent of polyps of less than 1 cm in size, compared to 10 percent of larger polyps. At present, the size designating a clinically significant polyp is, however, a matter of controversy.

In colonoscopy, a flexible fiber-optical probe is inserted into the colon through the rectum. By manipulating the camera attached to the tip of the probe, the physician examines the inner surface of the colon to identify abnormalities. Unfortunately, colonoscopy underachieves in reduction of colon cancer deaths due to its invasiveness and the resulting lack of patient acceptance.

Virtual colonoscopy was introduced by Vining [VG94] as a reliable, cost effective and less invasive alternative to physical colonoscopy, with respect to diagnosis. It has been shown to assist very well in finding polyps. Thus it has the potential to limit the application of physical colonoscopy to cases in which either a suspicious polyp was found or which was inconclusive in virtual colonoscopy. Virtual endoscopy can detect polyps of sufficiently large size, depending also on the scanning protocol used. By increasing scanning resolution, the probability of the detection of a small polyp can be raised. Fenlon et al. [FNS+99] state that the sensitivity of virtual colonoscopy depends also on polyp malignancy. The sensitivity is actually higher for adenomatous polyps than for harmless polyps.

False-negative results in polyp detection can occur with flat lesions and inadequate insufflation, while false-positive results are often due to undetected residual feces.

Virtual colonoscopy was the first area of application of virtual endoscopy and has evolved

farthest, in terms of acceptance among physicians and patients. Some centers no longer consider virtual colonoscopy a research protocol and are offering it as a screening tool despite its limitations [WR02].

Wood and Razavi [WR02] propose to use virtual colonoscopy also for preoperative evaluation of the colon, since it defines the exact anatomic location of abnormalities and the proximity of adjacent structures, whereas conventional colonoscopy only estimates the locations of lesions.

In a critical comment, Rogalla et al. [RMRH01] state that most studies about the value of virtual colonoscopy to date have been conducted in selected patient populations, whereas a true 'screening population' would have a lower prevalence of colorectal cancer or adenomatous polyps. A low specificity for virtual endoscopy in the screening setting would result in a large number of subjects with false-positive findings who would need to be reexamined by flexible endoscopy. The costs incurred by false-positive results are substantial and must be included when assessing the cost-effectiveness of virtual endoscopy.

Virtual Colonoscopy: the Workflow

Virtual colonoscopy starts with a bowel cleansing procedure, similar to that used in conventional optical colonoscopy. It is followed by inflating the colon with room air or CO_2 introduced through a rectal insert. Carbon dioxide produces less discomfort but is more expensive [Rex02]. Virtual colonoscopy is therefore still invasive, but to a much lesser extent, compared to physical colonoscopy. Then, using either CT or MRI, abdominal images are taken during a breath-hold and, finally, processed to provide the simulated endoscopic images.

The next step in virtual colonoscopy is segmentation: The set of voxels representing the colon must be identified. Chen et al. [CLW$^+$00] proposed an algorithm in which principal component analysis is used to construct a feature vector for each voxel in the volume. The similarities of the feature vectors are analyzed to classify image voxels. Then the air (or CO_2) inside the colon is extracted using region-growing methods. Langer et al. [LHB05] used hardware-assisted seed-based region growing based on non-linear diffusion, achieving satisfying results for colon segmentation.

To refine the segmentation, it is desirable to remove voxels representing residual stool from the segmented colon. This process is usually called *electronic colon cleansing*. Effective electronic colon cleansing methods might once relieve the patient completely of the need for bowel cleansing, which would further improve patient compliance to colon screening. Liang [LYW$^+$97] proposes to use contrast material to tag residual fluids, resulting in quick and reliable elimination of the regarding voxels from the lumen. Unfortunately, the usage of a contrast agent does not yet allow for segmentation of residual stool using thresholding. One reason for this are partial volume effects. Lakare et al. [LCL$^+$02] propose to sample CT intensity along rays orthogonal to the inner colon surface. The basic idea behind their approach is that the intersection of two distinct-density regions possesses a unique property. This property is the intensity profile at each intersection as the ray moves from one tissue into the other. As the ray passes through the region affected by the partial volume effects, the affected voxels therefore can be accurately identified and removed.

Virtual Colonoscopy: Techniques

Virtual colonoscopy can be substantially accelerated by providing automatic detection of polyps [SJP⁺01]. Yao et al. [YMFS04] presented a method that is based on a combination of knowledge-guided intensity adjustment, fuzzy c-mean clustering, and deformable models [YMFS04].

Most virtual colonoscopy systems rely on extracting (either automatically or semi-automatically) a path along the center line of the colon and employing either an automatic or a guided navigation scheme.

Liang [Lia01] suggests the following feature: The user is given the opportunity to define regions where polyps are suspected beforehand. The camera then moves quickly between these regions. When a region of suspicion is reached, the camera stops with the view centered on the area of interest.

Wan et al. [WDK⁺01] propose a technique for more accurate colon cancer diagnosis by using direct volume rendering to augment virtual colonoscopy. In a first pass, the colon walls are depicted, using, for example, standard polygon rendering. In regions of interest, i.e., in structures suspected to represent a polyp, volume rendering is performed for a few steps behind the surface. The interior density distribution, depicted in color and opacity, gives, as the authors state, a valuable clue, as to whether a structure is a polyp. Volume rendering is therefore used to reduce the problem that in virtual endoscopy no biopsies can be taken to exactly analyze the tissue a certain structure consists of. The authors call this technique *virtual biopsy*. Similarly, Shahidi [Sha96] stated that volume rendering enables the clinician to closely examine a suspicious region and its underlying tissue and thereby assists in detecting lesions and distinguishing between benign and malignant lesions.

Vilanova Bartoli et al. [VWK⁺01, VWKG01a] perform virtual colonoscopy by flattening the virtual colon. A path through the colon is defined. A distance field is generated, encoding the distances of all voxels from the path. Then, from each of a number of discrete points along the path, rays are cast, evenly distributed rotationally around the path and starting off in orthogonal direction to the path. The rays are tracked non-linearly until they intersect the colon surface. They follow the steepest gradients in the distance fields, which ensures that in curved parts of the colon, the concave parts of the surface are stretched and the convex parts of the surface are compressed, ensuring correct unfolding and introducing solely a scaling error. The final two-dimensional result is generated by taking the position of the ray source points along the path as one dimension and the rotational angle of initial ray directions as the second dimension and assigning the shading information acquired by the according rays to the according pixels.
In studies [SWB⁺02], the investigation of unfolded virtual colons was assessed to be of superior value for diagnosis of polyps compared to traditional three-dimensional fly-throughs.

4.3.2 Virtual Endoscopy of the Small Intestines

Physical colonoscopy is, though associated with discomfort for the patient, relatively un-problematic, from a technical point of view. The small intestines, on the other hand, cannot completely be examined using an endoscope. To view the inside of the small intestines, usu-ally a very uncomfortable procedure is applied: A probe is inserted into the patient's nose and advanced through the gastric tract into the small intestine. Through the probe, contrast agent is applied and the intestines can be viewed via x-ray. Schreyer et al. [SHK$^+$02] pro-pose to replace this examination with virtual endoscopy. They also state MRI as a suitable modality.

4.3.3 Virtual Gastroscopy

Carrascosa et al. [CCS$^+$02] assessed the value of virtual endoscopy in the maw, virtual gastroscopy. Virtual gastroscopy was compared to physical endoscopic exams in terms of detection of gastric lesions. They report a high number of true-positive findings versus zero false-positive and only one false-negative finding, concluding a high feasibility of virtual gastroscopy.

4.3.4 Virtual Bronchoscopy

The function of the respiratory tract is an integral part of human metabolism. Diseases, including tumors, pulmonary embolism, collapse of the lungs (atelectasis) and asthma, can severely impede the function of the tracheo-bronchial tree.

Diagnosis and treatment of such diseases often involve a bronchoscopic procedure: Dur-ing bronchoscopy, a camera, attached to an instrument called *bronchoscope* is inserted into the patient's mouth and advanced through the throat, larynx, trachea, and lower bronchial airways of the lungs. Bronchoscopy may be done to identify problems that may be causing inflammation and bleeding, diagnose certain lung diseases by collecting samples of lung tissue for biopsy or by collecting mucus samples to be examined in a lab for specific ab-normalities or infections, diagnose and determine the extent of lung cancer, remove foreign objects or thick mucus that may be blocking the airway, destroy growths in the airway or for planning of an open surgical intervention. In the latter case, the anatomical relation of diseased bronchi to non-diseased areas is inspected preoperatively, e.g., to determine re-sectability or to decide upon a surgical strategy [Bar03b]. Bronchoscopy is invasive, requires sedation and cannot be tolerated in some patients, especially children. For solely diagnostic purposes, virtual bronchoscopy has been proven capable of replacing this unpleasant proce-dure [HCO$^+$03]. With the development of Multi-Detector CT (MDCT), the high-resolution assessment of airways over the entire lungs is now possible during a single breathhold. The volumetric CT acquisitions made 3D investigation of the bronchial network for diagnosis, surgical planning and follow-up purposes possible. Virtual bronchoscopy can anatomically locate lesions, detect carcinoma and reliably evaluate airway stenoses [SAS$^+$02]. Physical bronchoscopy is restricted to airways the endoscope can reach, which forbids the investi-gation of small caliber bronchi or of severe stenoses and areas downstream. Virtual bron-

choscopy overcomes these limits.

The disadvantages of virtual bronchoscopy are its inability to obtain biopsy material and show color, friability and detail of the mucosal surface. Distal portions of the bronchial tree can become difficult to assess if they are filled with viscous secretions, such as blood [FMA$^+$97].

Wegenkittl et al. [WVH$^+$00] developed a planning tool of a trans-bronchial biopsy. In case of a bronchial tumor, a biopsy has to be taken from inside the bronchus via the working channel of an endoscope. Traditionally, the tumor is located by the radiologist using axial slice imaging and the position is communicated to the endoscopist. If the tumor is hidden behind bronchial walls, the endoscopist must do a blind biopsy, relying on the information acquired from the axial images. Important blood vessels which pass the site are also occluded. Using virtual endoscopy and rendering the presegmented tumor and the vessels behind semi-transparent bronchial walls, can significantly add to the safety of the procedure.

Kourelea et al. [KVM$^+$00] conducted a study in which virtual bronchoscopy significantly outperformed fiber-optic bronchoscopy for patients who coughed up blood from the respiratory tract. In 16 patients, virtual endoscopy managed to detect more existing pathologies than fiber-optic bronchoscopy. The authors even claim that in these patients, fiber-optic bronchoscopy was of no value, due to permanent occlusion of the endoscopic image by bleeding.

Kagadis et al. [KPK$^+$01] showed that virtual bronchoscopy is useful in the diagnosis and planning of surgical resection in patients with Tracheal Bronchus (TB), a rare congenital anomaly.

Cebral et al. [CS04] utilize virtual bronchoscopy and computational fluid dynamics to investigate tracheal and central bronchial aerodynamics. They show how disturbances in airflow through narrowed air passages can be visualized. Their system can be utilized for prediction of the need for intervention before a narrowing causes symptoms, documentation of successful treatment and prediction of transport of inhaled pharmaceutical aerosols.

Schultza et al. [SWM$^+$03] investigated the contribution of virtual endoscopy to the investigation of laryngotracheal pathological conditions. They state that coupling both virtual and physical endoscopy significantly improved the diagnostic investigation and facilitated management of the disease in patients with stenoses of the laryngotracheal tract. Still, they identify limitations of virtual endoscopy and state that spreading of lesions can be more accurately assessed by axial scan imaging.

Performing a virtual bronchoscopy analysis often requires an accurate segmentation of the bronchial tree from the original radiological data. Segmentation of the tracheo-bronchial tree is difficult, since the higher order branches are captured at a diameter of only a few voxels by the modality. The quality of the representation is further reduced by partial volume effects. Also, if CT is used, there is only limited contrast between the participating tissues. Perchet et al. [PFP04] propose an algorithm which combines advanced mathematical morphology and energy-based propagation. A set of low order airways is first extracted by using

a connection cost operator, applied in a multiresolution scheme. An energy-based propagation involving axial and radial potentials then allows recovering high-order bronchial segments. Bartz et al. [BMF$^+$03] propose a hybrid segmentation technique based on a pipeline of three segmentation stages: 3D region growing, 2D wave propagation and 2D template matching. Fetita et al. [FPBAG04] exploit shape-analysis characteristics of mathematical morphology to extract airways up to seventh-order branches.

Laryngoscopy (endoscopy in the larynx) is used for detecting causes of voice problems, such as a breathy voice, weak voice, or no voice, to discover causes of throat and ear pain, to evaluate difficulty in swallowing, a persistent sensation of a lump in the throat, or mucous with blood. It is further used to detect injuries to the throat, narrowing of the throat (strictures), or obstructions and masses in the airway. Virtual laryngoscopy was tested by Walshe et al. [WHM$^+$02]. In that study, all laryngeal lesions were correctly diagnosed on virtual laryngoscopy before direct laryngoscopy. Although virtual reality provided highly accurate representations of vocal cord lesions, it could not identify functional lesions of the vocal cords.

4.3.5 Virtual Endoscopy in Urology

A cystoscopy is an endoscopic inspection of the bladder and the urethra. It is performed either as an aid to diagnosis of lower urinary tract symptoms; or as part of a treatment plan relating to a specific condition, e.g., bladder tumor or stones. Virtual endoscopy has been applied in the urinary tract for a variety of purposes.

Generally, to create images of the interior of a lumen, based on radiological data, there must be either a naturally present high visual gradient between tissues (such as air next to bowel) or sufficient contrast material to allow differentiation between structures. In the urinary tract contrast medium can be used, or, alternatively, the tract can be insufflated with air or CO_2.

Prando [Pra02] demonstrates the various applications of CT-virtual endoscopy for the assessment of urinary tract abnormalities, based on the evaluation of 63 patients. He shows that CT-virtual endoscopy is not a competitive technique to conventional endoscopy of the urinary tract. It, however, has been proved a useful complementary tool. He concludes that CT-virtual endoscopy is a useful procedure, particularly in the follow-up of bladder tumors, complimentary evaluation of areas of difficult approach by endoscopy, and in the differential diagnosis of intrinsic versus extrinsic lesion of the renal pelvis and ureter.

The bladder was the first organ of the urinary tract to be assessed by virtual endoscopy [GPPA04]. Early systems produced poor results, based on low resolution CT data and surface rendering which performed poorly discriminating flat or small lesions. Later Kim et al. [KAP$^+$02] used higher resolution MDCT data and volume rendering. They scanned 73 patients and managed to detect 95 percent of existing bladder lesions.
MR cystoscopy in the bladder was also found feasible. With MR, a distended bladder filled with urine provides sufficient contrast for imaging. Lammle et al. [LBS$^+$02] report a study

59

in which surface-rendered MR cystoscopy detected all bladder lesions with a diameter exceeding 1 cm, but performed less well with smaller lesions. Due to poorer spatial resolution and higher cost, MR is unlikely to overtake CT cystoscopy in the foreseeable future.

Ghani et al. [GPPA04] state that due to the limitations of virtual endoscopy, including the inability to take biopsies, the lack of mucosal detail and risks associated with the application of contrast material and ionized radiation, it is unlikely that virtual cystoscopy can replace physical cystoscopy, but it may have a role in difficult cases and provide supplementary information of possible bladder disease.

There is a limited number of studies assessing virtual endoscopy in the upper urinary tract consisting of the kidneys and the ureters. Still, this application achieves promising results in the detection of tumors [THT$^+$99, THK$^+$00].

The applicability of virtual endoscopy in kidney stone surgery has been tested. First experiments with *virtual renoscopy* as a tool for examining patients with renal calculi proved to be advantageous [MCPA02]. The combination of the diagnostic accuracy of CT with virtual endoscopy for appreciating the location of a calculus may completely replace traditional examination techniques [GPPA04].

The usability of renoscopic simulators in surgical training has been tested with the result that this application may become an important part of future surgical training for renoscopy [MHS$^+$02].

Ureteral stents are thin catheters threaded into segments of the ureter. Ureteral stents may be placed on a long-term basis (months to years) in order to bypass ureteral obstruction. Short-term stenting (weeks to months) may be used as an adjunct to open surgical procedures of the urinary tract to provide a mold around which healing can occur, or to divert the urinary flow away from areas of leakage. Siablis et al. [SKL$^+$03] report upon a study in which virtual endoscopy was applied in patients with ureteral metallic stents. Virtual endoscopy was used as a follow-up tool, assessing the quality of stent placement and ureteral patency. The authors state that abnormally enlarged parts of the urinary tract and possible stricture formation could be well visualized. Virtual endoscopy outperforms real endoscopy in this application due to the possibility to easily bypass stenoses.

4.3.6 Virtual Endoscopy of the Nose and the Paranasal Sinuses

Three-dimensional imaging of CT data has been found to be helpful in the preoperative evaluation of some craniofacial pathologic conditions such as congenital anomalies and complicated fracture cases. With virtual endoscopy techniques the anatomy of the nasal cavities and paranasal sinuses can be demonstrated well.

Morra et al. [MCC$^+$98] assessed 3D virtual endoscopy with respect to its applicability in the analysis of radiological data. They state that virtual endoscopy provides a clear visualization of the anatomic structures of the nasal cavity and sinuses and that images are similar to those of conventional endoscopy. Moreover virtual endoscopy visualizes the paranasal sinuses, which are not accessible at conventional endoscopy. The main limitations of virtual

endoscopy are the arbitrary choice of the transfer function (or the iso-value) and the scalarization resulting in homogenization of different tissue densities, which reduces the contrast quality. Overall the authors view virtual endoscopy as a complementary technique to standard axial viewing, with application especially for didactic activity.

Bisdas et al. [BVS$^+$04] stated that in comparison to looking at plain radiographs, virtual endoscopy allows physicians to better understand the complex and varied intracranial structures. They tested virtual endoscopy in 50 patients with the diagnosis of an acute or chronic nasal obstructive disease. They used surface rendering with a fixed threshold of -400 Hounsfield Units. Virtual and real endoscopic images were compared with respect to diagnosis and the recognizability of 20 important anatomical structures, including the nasal septum, inferior, middle, and superior turbinate, maxillary and sphenoid sinus, spheno-ethmoid recess, choanae, and throat. A diagnostic concordance of approximately 92 percent was achieved. The 3D images allowed a clear demonstration of the various pathologies within their anatomical contexts, indicating the degree of the obstruction as well as morphological characteristics of diseases larger than 3 mm. The authors list the inability to define the type of a detected tumor and observed artifacts (including partial volume effects, spiral patterns and an unnatural unevenness of the cavity wall) as major disadvantages. Also, they point out that their virtual endoscopy software proved not to be adequate when the nasal cavities were totally filled with soft tissue, such as in extensive acute inflammation or in a heavily obstructive disease. However, the general agreement with intraoperative findings indicates the ability of virtual endoscopy to reliably demonstrate the anatomical structures of the nasal cavity as well as intranasal masses of sufficient size and soft-tissue pathologies.

Rogalla [Rog99] reports a prospective study in which two radiologists assessed the coronal reconstructions and virtual endoscopy with respect to the ease of finding pathology. In 30 patients, the surgeons were asked to rank the degree of assistance of the preoperative virtual endoscopy. A high degree of similarity between virtual endoscopy and intraoperative impression was reported. Due to surface rendering with a fixed threshold, however, certain anatomy (e.g., upper nasal conchae) were not displayed and the maxillary sinuses could not be approached. Still, virtual endoscopy was found applicable as a preoperative assistance for the surgeon.

Virtual endoscopy of the head usually involves the acquisition of a CT image of the head, which is not unproblematic, because it exposes the patient to deleterious radiation. Dammann et al. [DMTR$^+$00] describe how the radiation dose exerted to the patient can be reduced significantly during the CT scan of the head by adjusting the parameters of the modality. They state that using up-to-date CT scanners, radiation exposure may be reduced by up to 95 percent compared to that of conventional CT techniques.

4.3.7 Virtual Endoscopic Neurosurgery

Virtual endoscopy has been applied to neurosurgery in various ways. Burtscher et al. [BBD$^+$02] report on a study assessing virtual endoscopy as a tool for preoperative analysis for neurosurgical interventions, including third ventriculostomies, cyst removals, ven-

tricular arachnoidal cyst fenestrations, and endoscopic septostomies. They state that in 8 of 12 patients virtual endoscopy profoundly influenced surgical planning and surgery itself. They conclude that virtual endoscopy can significantly add to the safety of neurosurgical procedures.

Auer and Auer [AAK97] assessed virtual endoscopy and its use in neurosurgery. They state that virtual endoscopy provides impressive views of intracranial and intracerebral cavities, with visualization of many anatomic details of the brain's inner and outer surfaces. They conclude that the method seems especially suited for the simulation and planning of operations of intraventricular lesions. As a major shortcoming, they mention that different tissue compartments cannot be stained in their original coloring, which would facilitate their recognition and thus orientation in space by anatomic landmarks.

One well known area of application of virtual endoscopy is *third ventriculostomy*. The CSF (cerebrospinal fluid) is produced and resorbed in the ventricular system of the human brain. Sometimes the cerebral aqueduct - a narrow passage between the third and fourth ventricle - is occluded due to a tumor, an injury, meningitis or a native defect, thus blocking the natural flow of the CSF. This leads to a dangerous increase of brain pressure and, possibly, severe damage of the brain. To avoid this, a ventriculostomy can be applied: A new CSF drain is realized in the floor of the third ventricle using endoscopic tools. The *basilar artery* is located directly below the floor of the third ventricle and not visible from the third ventricle. Damage to this vessel can lead to a fatal bleeding. Virtual endoscopy, which can render translucent walls and objects of interest (e.g., vessels) in the background, can be applied to avoid such traumas. The VIVENDI system [BS99] of the University of Tübingen was applied to aid third ventriculostomy [BSG+01]. Data acquired from MRA (Magnetic Resonance Angiography) providing a good contrast for important blood vessels is combined with data from conventional MRI which enables segmentation of the ventricular system. The resulting multimodal renderings can be used to facilitate pre-surgical planning. In the background, VIVENDI depicts large blood vessels whose being damaged would lead to lethal consequences. Also smaller vessels are rendered, since damage to those leads to bleeding resulting in a sudden loss of visibility through the endoscope. In that case, the procedure must be aborted.

The system can also be used as a 3D navigation aid, by tracking the tip of the endoscope and mapping the position into the MRI data set. The position is loaded into VIVENDI and the real and virtual images are fused.

Burtscher et al. [BDB+00] assessed the value of endoscopy for the planning of third ventriculostomy procedures. Virtual endoscopy was used to preoperatively evaluate the individual intraventricular and vascular endoscopic anatomy at the floor of the third ventricle. The virtual views were correlated with the real endoscopic images, intraoperatively obtained by a standard ventriculoscope during endoscopic third ventriculostomy procedures. The authors state that the virtual endoscopic images were comparable with the real intraoperative endoscopic view in 8 of 9 patients. Virtual endoscopy was found useful displaying the positions of important cerebral arteries relatively to visible anatomy, facilitating a safe procedure. The authors conclude that preoperative virtual endoscopy planning can intraoperatively assist the neurosurgeon to find a safe location for third ventriculostomy.

Shigematsu et al. [SKH$^+$98] investigated the potential of intracranial virtual MRI endoscopy applied to visualization of a variety of pathologies (including trigeminal neuralgia, hemifacial spasm, acoustic neuroma, suprasellar germinoma, Langerhans cell histiocytosis, lateral ventricle nodules and pituitary dwarfism). In 12 out of 13 patients, the pathologies were well seen in the 3D images, only a small acoustic neuroma (a benign tumor near the middle-ear) was missed by virtual endoscopy.

Himi et al. [HSS$^+$00] used virtual endoscopy for imaging of the middle ear and assessment of otological disease, including the detection of lesions. As a time-saving alternative to traditional means of middle ear imaging, virtual endoscopy may have a significant role in preoperative planning, surgical training and postoperative evaluation in otology. The authors propose to equip systems using surface rendering with interactive threshold adjustment which would improve the detection of fine lesions of the ossicles.

Talala et al. [TPK$^+$00] used virtual endoscopy to preoperatively familiarize the surgeon with patient anatomy in endonasal transsphenoidal pituitary surgery.

4.3.8 Virtual Angioscopy

Injuries and diseases of the blood circulation system can result in serious, potentially life-threatening conditions. Diseases can cause stenoses or aneurisms, which both constitute potential causes of lethal conditions.
Currently the most widely used method for the visual assessment of the state of the vascular system is 2D angiography: A contrast agent is injected via an endovascular catheter. Subsequently, an X-ray radiograph is taken to obtain a 2D projection of the vessels. These images are of high resolution but lack spatial information.

Virtual endoscopy has been applied in blood vessels [DLR$^+$96], e.g., in the assessment of abdominal or cerebral aneurisms, carotid stenoses and atherosclerotic plaques. Characterization of broad-based aneurisms can help to determine whether surgical treatment is preferable to coil embolization (a catheter-based procedure that allows precise occlusion of an aneurism) [Bar03b, KSK$^+$99]. The latter treatment can only applied to *non-fusiform* aneurisms, i.e., aneurisms which are separated from the principal blood flow through the diseased vessel. In these aneurisms, a constriction (a *neck*) can be identified and this neck can be closed resulting in exclusion of the aneurism from blood flow. Virtual endoscopy can give valuable information about the shape, the position, the orientation and the connection of an aneurism and therefore how it can be treated [BSSW99].

Cardiac diseases are among the leading causes of death in industrialized countries. The under-supply of the heart muscle (myocardium) with oxygen through the blood leads to critical conditions which can result in a possible collapse of blood circulation. The actual cause of an under-supply of the myocardium is usually a stenosis or an occlusion of a coronary artery. For the diagnosis of these malfunctions, four-dimensional data sets can be acquired using ECG(Electro-Cardio-Gramme)-triggered CT or MRI, capturing the dynamics of the myocardium and the cardiac system. While myocardial dynamics are usually captured by
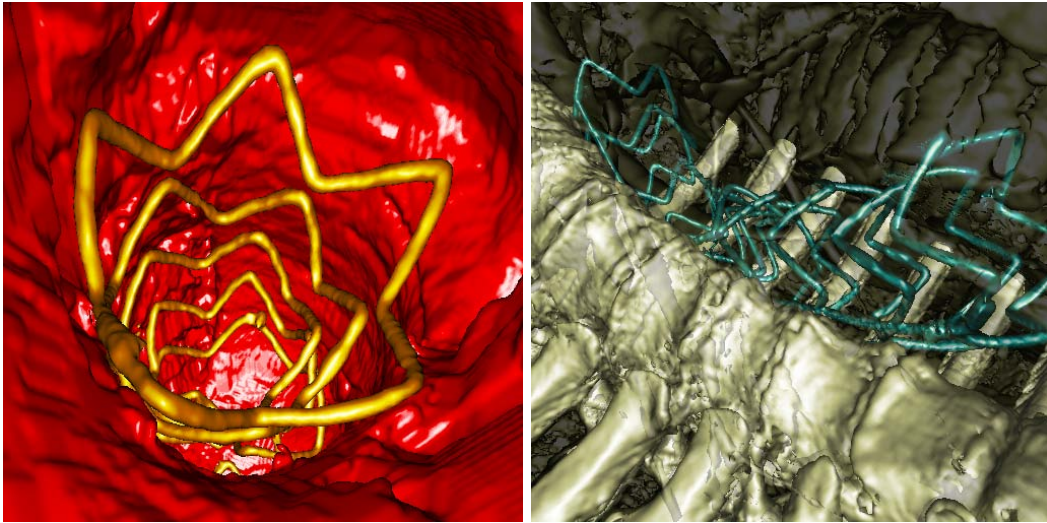
Figure 4.1: A metal stent inside a human aorta. Virtual endoscopy can be used to assess the quality of stent placement.

segmenting the myocardium [NW03b, NW03a, NW03c] and evaluating parameters of its movement (e.g., analyzing expansion and contraction patterns) over time, the reasons for abnormalities can be diagnosed with the aid of virtual endoscopy, e.g., by detecting and analyzing stenoses, possibly even using four-dimensional cardiac data. This, however, poses the problem that, to retain endoluminal views, the view point of the virtual camera must be adapted with the scene. Bartz et al. perform virtual endoscopy inside the coronary blood vessels [BGL$^+$01].

Kawasaki Disease is an inflammatory illness of young children that can seriously affect the cardiovascular system. Ayeni et al. [AHR01] propose virtual endoscopy based on intravascular ultrasound (IVUS) images of patients with Kawasaki Disease. The disease is well characterized by IVUS images - therefore this modality is used. The IVUS images are segmented using a semi-automatic technique. All voxels belonging to the segmented lumen are then simply set to zero, enabling a virtual endoscopic fly-through. Images are, logically, fractionated due to the gated image acquisition and noisy ultrasound images. Still, the authors state that the diseased region of the coronary vessel is well defined by the difference in luminal size and by difference in appearance of arterial wall texture and shape during angioscopic fly-through, enabling analysis of the presence and damage caused by the disease. Three-dimensional representation provides information otherwise unattainable. Still, the quality of the result is rather heavily dependent on the quality (resolution, contrast) of the acquired data and the accuracy of the segmentation.

Neubauer et al. [NMHW02] report upon the usage of virtual angioscopy for the assessment of aortic stent placement (see Figure 4.1). A stent is a small tubular prosthesis that is inserted into an artery via an endovascular procedure and used, for instance, to bridge a stenosis. Virtual endoscopy can provide valuable information about the effects of a stent on blood flow.

Davis et al. [DLR$^+$96] used virtual intraarterial endoscopy based on MRI data on patients with various pathologic aortic conditions. Whilst noticing also limitations, they state that virtual endoscopy was useful in detecting abnormalities, including the two lumina in a congenial double aortic arch and the single lumen after correction, vessel narrowing in coarctations, characteristics of Leriche syndrome, stenoses, occlusions and the true and false lumina of dissections.

Wahle et al. [WOS04] perform virtual angioscopy utilizing fused information from X-ray angiographic images and intravascular ultrasound data.

Summers et al. [SPMH99] calculate the fractal dimension of vessel surfaces to detect and quantify artherosclerosis. They found that the fractal dimension of the vessel of a healthy volunteer was lower (the reconstructed vessel wall was smoother) than that of a patient with atherosclerosis. They measured the fractal dimension of surfaces by implementing the *variation method* by Dubuc et al. [DZT$^+$89].

Chapter 5

STEPS - Preprocessing of Input Data

STEPS (Simulation of Transsphenoidal Endoscopic Pituitary Surgery) is a virtual endoscopy system for the planning and training of transsphenoidal interventions. It is designed to aid surgeons in getting acquainted with the endoscopic view of the anatomy, the handling of instruments, the transsphenoidal approach and challenges associated with the procedure [NWF$^+$04, NWF$^+$05]. STEPS also assists experienced surgeons in planning a real endoscopic intervention by getting familiar with the individual patient anatomy, identifying landmarks, planning the approach and deciding upon the ideal target position of the actual surgical activity. The application provides interactive visualization, navigation and perception aids and the possibility to simulate the procedure, including haptic feedback and simulation of surgical instruments.

This chapter deals with preprocessing of input data. The virtual endoscopy system is described in Chapters 6 (rendering techniques) and 7 (the user interface). A discussion of the advantages and disadvantages of STEPS is given in Chapter 8. A complete virtual procedure is illustrated in Chapter 9.

Before STEPS can be used to interactively explore a patient's nasal and paranasal cavities or to perform a virtual training operation, some preprocessing steps must be applied to obtain suitable input data. Radiological images of the patient's head are generated using both CT and MRA. CT data give information about skull anatomy. The pituitary gland, the tumor, the optic nerves and the ICAs are usually well captured by MRA. The size of a typical CT data set amounts to approximately $512 \times 512 \times 170$ voxels, with a voxel size of $0.44mm \times 0.44mm \times 1mm$. The MRA data sets used comprise approximately $512 \times 512 \times 100$ voxels, with a voxel size of approximately $0.4mm \times 0.4mm \times 1mm$. Objects of interest are extracted (segmented) from the MRA image. Since the visualization of the boundaries of the investigated cavities during the virtual endoscopy procedure is based on the CT-data, the two data sets must be geometrically aligned (registered) to ensure exact correlation of object locations.

After registration, objects of interest can be segmented in the MRA image and displayed at correct positions during the CT-based virtual fly-through. Figure 5.1 illustrates the rendering pipeline of background objects.

STEPS was implemented as a collection of three modules, embedded into the Java-based medical workstation J-Vision (Tiani Medgraph, Vienna, Austria). Registration, segmenta-
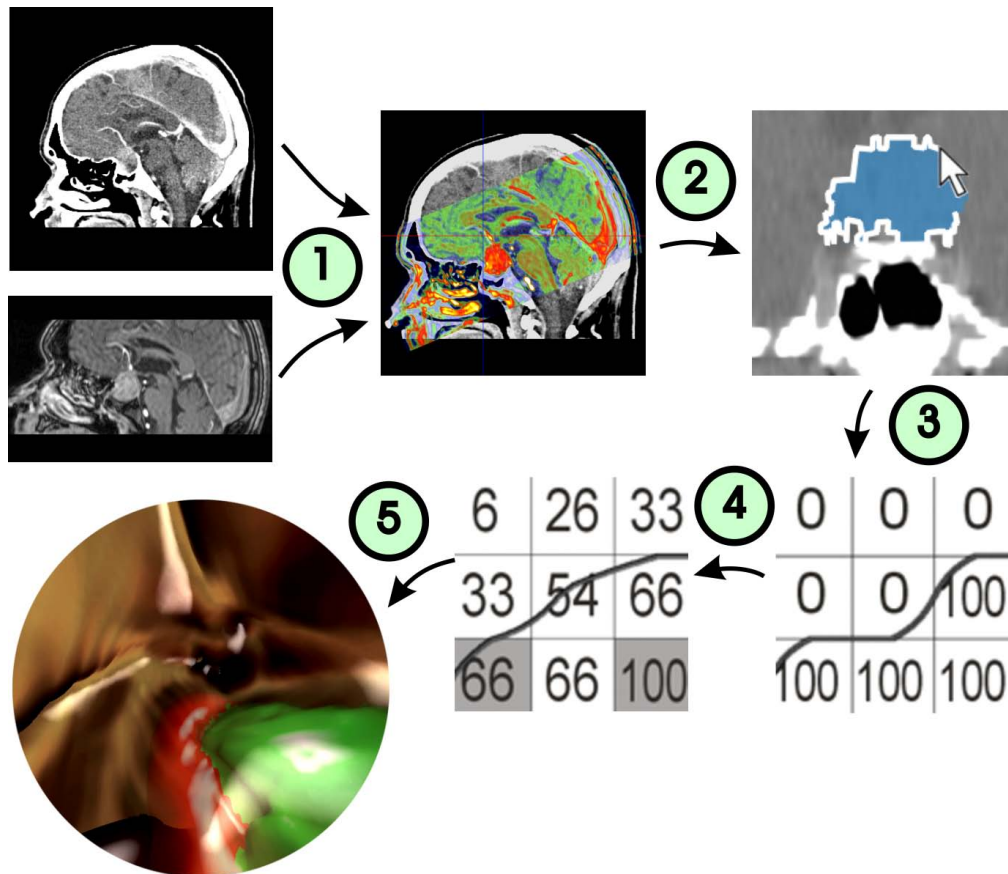
Figure 5.1: The rendering pipeline for background objects. Step 1: The CT and MRA data are registered; Step 2: The background object is segmented; Step 3: An artificial iso-surface is generated from the binary segmentation mask; Step 4: The artificial iso-surface is smoothed; Step 5: The background object is rendered

tion and virtual endoscopy were each encapsulated into one module. Sections 5.1 and 5.2 describe the registration and segmentation modules, respectively. Section 5.3 deals with reconstruction of segmented background objects.

5.1 Registration

The process of registration of two radiological images, which is also referred to as *matching*, is the search for the definition of the transformation that matches one image to the other such that after the transformation two voxels, one in each image, which correspond in terms of image coordinates, also correspond in terms of world coordinates. Correct image registration maps each point in one image onto the corresponding point in the second image.

Mostly it is assumed that the transformation between images of different modalities is isotropic. It is assumed that neither image is skewed and that non-linear (e.g., barrel) distortion for each modality is negligible. Thus, the transformation under question can be com-

posed out of simple linear transformations, such as rotation, scaling and translation. When there is variability in geometric structures between images, e.g. distortion in the MRI image because of the variable magnetic field, non-linear transformations (e.g., warping) may be applied.

Overviews of registration techniques are provided by Maintz and Viergever [MV98] and by Pluim et al. [PMV03]. Registration techniques can be roughly divided into the following two groups:

- **Control Point Based Registration:** Control point based registration requires as an input a set of predetermined correspondences between the images. This can be provided by fiducial markers, visible in both images [CWK$^+$01] (*marker-based registration*). Alternatively, the user can manually establish the correspondence. Also there has been work in using automated identification and matching of physiologic or anatomical landmarks [SFB98] (*landmark-based registration*).

- **Voxel-Based Registration:** This method takes one image as a reference and iteratively transforms the other image until a similarity criterion is optimized. This requires a quantification of the similarity between two images.
  In the simplest case, the similarity is assessed by comparing raw gray-values. This, however, assumes that both images are recorded by the same modality using the same protocol. Even if these assumptions are met, image noise can considerably hamper the search for a global optimum.
  Another simple example is *edge-based registration*. This technique can only be used if the images contain clearly defined boundaries between regions. These edges are extracted using a special image filter (e.g., the Sobel filter). All information other than the resulting edge image is discarded. Frequently, for registration, simply the gray-values of the edge images are compared and the MSE (mean square error) is minimized. These methods suffer from the fact that only a part of the available image information is used since only gradient magnitudes are examined. Also the optimization criterion often presents a very sharp global optimum. Because of noise in contour images, the iteration often gets stuck in local optima, leading to a totally meaningless registration [KIMW00]. In order to overcome this problem, and to ensure high reliability, contour-based methods must involve global optimization techniques (e.g., genetic algorithms).
  A more sophisticated similarity criterion is *mutual information* [MCV$^+$97, VW99]. Mutual information is a concept from statistics and describes the degree of dependence of two statistical variables A and B. The mutual information, $I(A, B)$ is defined as:

$$
\begin{aligned}
I(A, B) \quad &= H(A) + H(B) - H(A, B) \\
&= H(A) - H(A|B) \\
&= H(B) - H(B|A)
\end{aligned}
\tag{5.1}
$$

where the entropy $H(A)$ is known to be a measure of the amount of uncertainty about the random variable A and is defined by:

$$
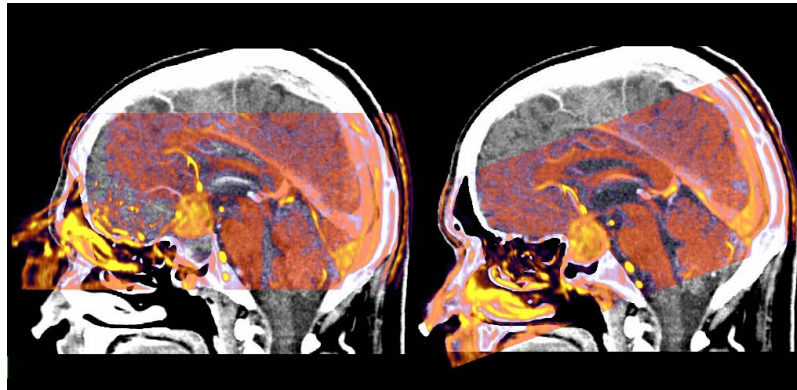H(A) = -\int p_A(a) \cdot log\, p_A(a)\, da
\tag{5.2}
$$

68

Figure 5.2: Registration: The CT and the MRA data are aligned; Left: CT and MRA images before the alignment; Right: after the alignment

where p_A is the statistical distribution of variable A. Considering the image intensity values, a and b, of a pair of corresponding voxels in the two images that are to be registered to be random variables A and B, respectively, estimations for the joint distribution $p_{AB}(a, b)$ and the marginal distributions $p_A(a)$ and $p_B(b)$ can be obtained by simple normalization of the joint and marginal histograms of the overlapping parts of both images. Registration is performed by optimizing the geometric transformation T_α defined by the registration parameter α. The mutual information registration criterion states that the images are geometrically aligned by the transformation $T_{\alpha*}$ for which $I(A, B)$ is maximal.

Capek and Wegenkittl [CW01] state that mutual information has a clearly expressed global extreme which is well-detectable in parametric space.

There are different similarity criteria serving as possible alternatives to mutual information, e.g., the one presented by Woods et al. [WMC93].

The registration module used by STEPS aligns two three-dimensional data sets by using rigid body transformations with six degrees of freedom (three translations and three rotations). The alignment is achieved by maximizing mutual information. The optimization is performed using adaptive simulated annealing.

Capek and Wegenkittl [CW01] developed this fast and robust module for voxel based registration. Intuitive user interaction has been added. This allows performing a manual pre-positioning of the dataset to be registered, which increases the probability of the global optimum being found automatically. To ensure finding the global optimum, usually only very rough manual pre-processing is required. A skilled user performs this task in less than two minutes. No manual definition of landmark points or similar is required. The automatic registration process takes approximately 10 seconds on a standard PC.

Figure 5.2 shows blended CT and MRA data in a multi-planar reconstruction (MPR), before (left) and after (right) the automatic registration process. For better discrimination, the MRA data was colored using a transfer function.

5.2 Segmentation

Medical image segmentation is the process of partitioning a medical image such that each resulting group of voxels represents a certain real-world object of interest (e.g., an organ or a vessel). Segmentation is one of the most important problems of medical image processing and visualization. There is a constantly ongoing search for methods to extract certain parts of anatomy with as little user interaction needed as possible.

Today, a high level of automation of the segmentation process is in many cases still associated to its specialization onto a specific task, like e.g. segmentation of the ventricles of the heart [NW03b, NW03a, NW03c], vessels, the liver, the prostate or the brain [BNH$^+$05]. Only simple segmentation methods (e.g. thresholding, region growing) are all-purpose tools. These universally applicable segmentation methods still require user-interaction and post-processing to get reasonable results, but are fast and simple to apply.

5.2.1   Automatic Segmentation

Automatic approaches usually rely on models describing the usual appearances of certain anatomic structures. These models are extracted from training sets and mirror the variability of object appearance in these training sets expressed in statistical measures. Important examples are Active Shape Models which encode object shapes, and their successors, Active Appearance Models [CET98], encoding object shapes and textures. These segmentation methods usually start off by placing the average shape encountered in the training set into the volume. Then the object is scaled, translated and deformed, using modes of variation extracted from the training set, thereby always retaining a 'legal' shape, until the thereby synthesized object matches the underlying volume data.

However, methods based on appearance models become problematic as soon as tumors are involved. Tumors often grow and deform the neighboring anatomy in a chaotic way, hardly describable by a simple statistical model. The wide range of possible tumor shapes in fact makes the description of variability an intractable task.

Since this is often the case in the presented project, STEPS relies on semi-automatic segmentation.

5.2.2   Semi-Automatic Segmentation

There is a wide variety of semi-automatic algorithms, including the live-wire technique, level-set methods, deformable models [TPBF87] and seed-based region growing. For STEPS, the so-called *watershed-from-markers* technique is used: The watershed algorithm is a classical algorithm in mathematical morphology originally described for 2D data sets. Structures and their boundaries are detected using a simple idea: The gradients of the image can be interpreted as a height field similar to a landscape with ridges of mountains and valleys. Keeping this picture, the algorithm is equivalent to flood the landscape step by step with water up to a predefined level. The emerging basins correspond to the segmented objects and their boundaries.

In general, the algorithm is universally applicable to segmentation problems but works best in datasets with low noise and for structures with clear boundaries i.e., if surrounding tissue

differs clearly from tissue of the object of interest, or if contrast agent is present.

The original watershed algorithm starts from local minima without taking specific objects of interest into account and therefore suffers from over-segmentation. The watershed-from-markers technique improves this by adding simple user interaction: Markers are placed into the volume, each marker related to either an object of interest or the background. This additional input adds information about object connectivity to the algorithm and reduces the probability of generating incorrect boundaries.

Felkel et al. [FWB01] developed an extremely memory efficient and fast implementation of the watershed-from-markers algorithm, extended to 3D, with acceptable performance even for large data sets. The increased speed (the calculation takes about 25 seconds on average on a P4-3000 MHz) allows multiple passes of editing the markers and running the algorithm until the result is satisfactory. Minor adaptations can be (and usually are) carried out manually, in a paint/erase fashion. Manual editing is important to retain complete control over the segmentation process and to achieve a medically sound segmentation result. Manual adaptation takes about 50 percent of the time spent on segmentation, i.e., approximately 15 minutes per patient.

User interaction in the segmentation module takes place via a MPR (multi-planar reconstruction, see also section 7.1). Objects of interest can therefore be marked on arbitrary cross-sections through the volume. This presents a drastic improvement to segmentation carried out solely on original slices, as performed by Talala et al. [TPK$^+$00] for segmentation of the ICAs in CTA images. They state that partial volume effects sometimes made manual segmentation difficult because it was not always easy to distinguish between the bony structures and the contrast medium in the ICAs, especially when the direction of the vessel was not perpendicular to the CT slice. Using the MPR, the impact of this problem can be significantly reduced.

3D previews of all segmented objects can be viewed at any time during segmentation, using the real-time volume rendering library presented by Mroz and Hauser [MH01].

5.3 Reconstruction of Segmented Background Objects

There are two kinds of background objects in virtual endoscopy applications: The first are those which can be represented by an isosurface in the original data volume. Examples are bones and, if CTA or MRA has been used to record the data, also blood vessels. These objects can be extracted using thresholding or, equivalently, iso-surfacing in the original data. The second group of background objects are those which cannot be extracted using simple thresholding (e.g., tumors). For these objects, a surface must be constructed by editing the data volume.

5.3.1 Filtering of Binary Objects

Gibson [Gib98a] presented *elastic surface nets*, an approach which constructs a smooth triangle mesh around a binary segmented object: A vertex is positioned in the center of each cell which is adjacent to at least one voxel inside and at least one voxel outside the binary

segmented object. In each of a series of relaxation steps, each vertex is moved slightly towards the average position of vertices in the neighboring cells, with the constraint that no vertex is allowed to cross a cell boundary. After this smoothing phase, a triangle mesh is created by connecting the resulting vertex positions. This technique is a good choice if polygon rendering is used to display background objects. This algorithm was further refined in subsequent work [dBVP$^+$00, NBW$^+$05].

If, as is the case with STEPS, visualization is performed directly on the voxel values, rather than via an intermediate data structure (e.g., a polygon mesh), artificial isosurfaces can be constructed through *voxelization*. Trivial voxelization techniques assign a value v_1 to the voxels surrounding the segmented object, with $v_1 < v_{iso}$, v_{iso} being the iso-value defining the isosurface, and a value v_2 with $v_2 > v_{iso}$ to all voxels belonging to the segmented object. The resulting isosurface, however, suffers from heavy aliasing, since the binary classification results in a discontinuous inside-outside function with an unbounded frequency spectrum, and trilinear interpolation has only a limited filtering effect. Thus a considerable fraction of the isosurface is positioned exactly on cell boundaries, visibly revealing the underlying voxel-structure (see Figures 5.6 through 5.10). To improve image quality, the isosurface should be faired, i.e., low-pass filtered [SK98]. One way to do this is to simply apply a low-pass filtering kernel (e.g., Gaussian) to the volume data.

One problem with the application of filter kernels is that its effects on the object shape cannot easily be controlled. Thin parts of objects might fall victim to overfiltering and disappear, while more massive parts require a certain minimum amount of filtering to gain the required smoothness.

A straightforward solution to the problem of object shapes being significantly altered by filtering is to confine voxels inside the segmented object to values $\geq v_{iso}$ and other voxels to values $\leq v_{iso}$, thereby maintaining the original inside-outside classification of voxels and therefore principal object shapes. This, however, can again lead to aliasing, because, with increasing amount of filtering, voxel values near the object boundary tend to acquire the value v_{iso}. If values are confined to be $\geq v_{iso} + \delta$ or $\leq v_{iso} - \delta$ with an arbitrary value δ, a large amount of filtering yields a quasi-binary classification: voxels near the boundary acquire either the value $v_{iso} + \delta$ or the value $v_{iso} - \delta$, restoring aliasing as it was before filtering.

A more sophisticated solution would be to confine filtering to only those voxels which are within a bounded region around the object boundary. This *filter region* contains all voxels whose distances to the real object boundary do not exceed a certain maximum. The aliasing effect is reduced, but the result is still not satisfying, as illustrated in Figures 5.3 and 5.4. The figures show the real boundary of the object. The filter region is bounded by the dotted curves. The real object boundary is located exactly in its center. Ideally, the object should be voxelized such that the resulting isosurface is as similar to the real object boundary as possible, with errors occurring only due to subsequent discretization of the intensity function to a set of voxel values. Figure 5.3 shows this ideal case: The intensity linearly decreases from v_2 to v_1 along any straight line outwards, as indicated by the red curve. Thus the center of the filtered region, which is, in the continuous case, equal to the real object boundary, acquires the value $v_{iso} = (v_1 + v_2)/2$.

If the filter region is convolved with a symmetric filter kernel, the object shape is sig-
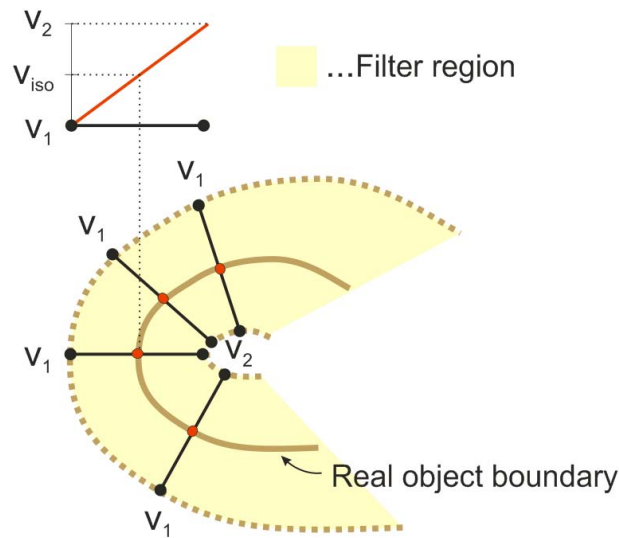
Figure 5.3: Filtering is applied only to the voxels inside the filter region, delimited by the dotted curves. Ideally, voxel values should increase from v_1 to v_2 linearly as illustrated by the red curve.

nificantly altered. This can be shown by imagining the filter kernel being centered exactly on the real object surface. As Figure 5.4 shows, the kernel obtains significantly more support from outside the object than from the inside, resulting in a voxel value that causes the voxel on which the kernel is centered to be classified as *outside the object*, rather than *on the surface*, as would be desired. Thus, the convexity of convex object parts is reduced. For concave parts, it is analogous. The object shape will tend towards spherical, constrained only by the boundaries of the filter region.

This problem can be reduced by applying a distance-based approach. If voxel values linearly decrease with increasing distance from the inner boundary of the filter region, the desired case depicted in Figure 5.3 can be well approximated.

Distance field based approaches to voxelization were presented by Sramek [Sra94], Jones [Jon96] and other authors. Gibson [Gib98b] showed how distance fields can be used to improve the smoothing of elastic surface nets.

Inspired by this, a simple solution would be to create a signed distance field around the boundary of the binary segmented object and select voxel values according to distance. This, however, does not result in a smoothed iso-surface, because, in the vicinity of the iso-surface, voxels are assigned either a small negative distance value (if they are inside the object) or a small positive distance value (if outside the object), restoring the quasi-binary classification resulting in blocking artifacts.

73
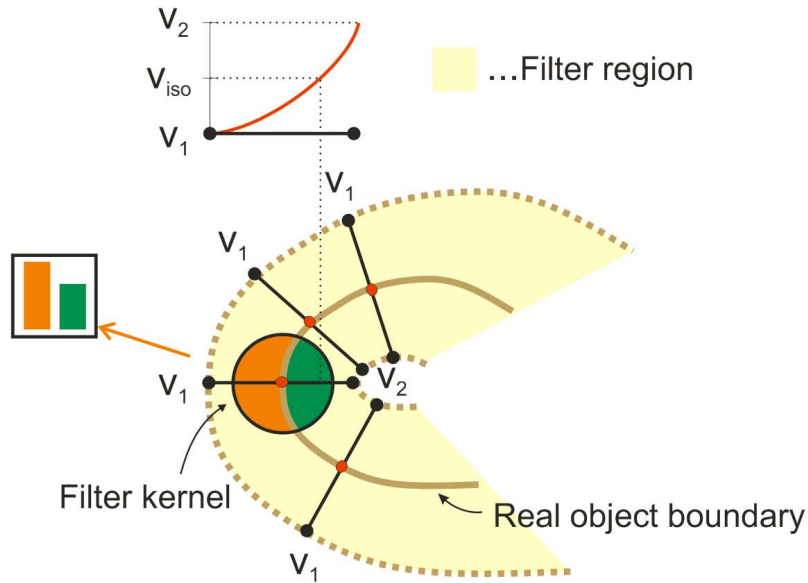
Figure 5.4: Due to the convex shape of this part of the surface, the filter, centered on the real object boundary gets more support from outside the object. Therefore the isosurface is moved towards the inside.

5.3.2 Object Reconstruction as Performed by STEPS

This section develops the algorithm used for object reconstruction in three steps. Algorithm 1 implements the basic idea, Algorithms 2 and 3 present optimizations.

Algorithm 1 The technique used by STEPS smoothes the boundary of a binary segmented object by generating a *reference object* through erosion of the original object and assigning final voxel intensity values indirectly proportional to the distance from the reference object (see Algorithm 1). The surface of the reference object can be seen as an approximation of the inner boundary of the filter region as introduced in Section 5.3.1 (see Figures 5.3 and 5.4). For the erosion, a spherical kernel is used. The amount of filtering can be controlled by selecting the radius r of the erosion kernel.

The assigned voxel values range between v_1 and v_2, with $v_2 > v_1$ and the iso-value $v_{iso} = (v_1 + v_2)/2$. All voxels inside the reference object acquire the value v_2. Each voxel positioned at a distance d with $d < 2 \cdot r$ from the reference object acquires a value v with:

$$v = v_1 + (1 - \frac{d}{2 \cdot r}) \cdot (v_2 - v_1) \tag{5.3}$$

Voxels positioned at a distance greater than $2 \cdot r$ are assigned the value v_1.

This basic algorithm suffers from the problem that object shape characteristics are significantly altered. Features of the same magnitude as r or smaller are removed from the object with a high probability. To prevent this, the algorithm is adapted as follows:

Algorithm 1 Smoothing a binary segmented object

1: Algorithm input:
   $O_{original}$: original (unsmoothed) object
   $r > 0$: extent (radius) of object erosion
2: initialize the reference object $O_{reference}$ as equal to the original object $O_{original}$
3: generate a spherical erosion kernel K with radius r
4: **for** each volume cell located at the boundary of $O_{original}$ **do**
5:   Place K such that C_{kernel}, the center of K, coincides with the center of the cell
6:   **for** each voxel V covered by K **do**
7:     **if** $V \in O_{reference}$ **then**
8:       remove V from $O_{reference}$
9: **for** each voxel V inside $O_{reference}$ **do**
10:   $voxelValue_V \leftarrow v_2$
11: generate spherical distance transform mask M with the radius $(2 \cdot r)$
12: **for** each pair V_1, V_2 of neighboring voxels with $V_1 \in O_{reference}$ AND $V_2 \notin O_{reference}$ **do**
13:   Center M on the midpoint P of the line segment connecting V_1 and V_2
14:   **for** each voxel V covered by M, with $V \notin O_{reference}$ **do**
15:     $voxelValue_V \leftarrow$
      $MAX(voxelValue_V,$
      $v_1 + (v_2 - v1) \cdot (1 - (distanceBetween(P,V) + 0.5)/(2 \cdot r + 1)))$

Algorithm 2 The reference object is initialized to be equal to the original object. Again, the reference object is eroded using a spherical kernel of radius r. This erosion can be seen as iteratively peeling off the outer layer of the reference object. After each removal of a layer, it is checked, whether the object shape has significantly changed, i.e., features have been lost. This is the case if parts (i.e., voxels) of the reference object were removed in a peeling step although they are not adjacent to the object remaining after this peeling step. Those parts (voxels) are detected and returned to the reference object. The adapted technique is described in Algorithm 2.

The iterative peeling is performed by combining the erosion with a distance transform, i.e., each eroded voxel is assigned the distance to the nearest part of the boundary of the original object. After the erosion, the algorithm checks for each eroded voxel V with a distance
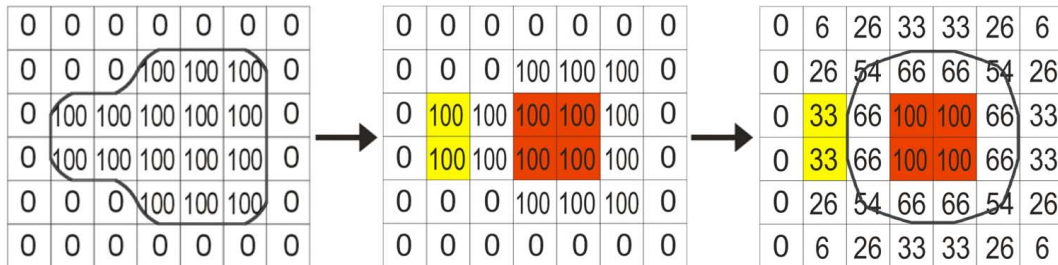
Figure 5.5: If Algorithm 1 is applied, important object features may be smoothed out. Here: $v_1 = 0$, $v_2 = 100$, $v_{iso} = 50$

value d_V, whether at least one of its neighbor voxels was assigned a distance value d_N with $d_N > d_V$. If that is not the case, the voxel V is added to the reference object and assigned the voxel value $voxelValue_V$ with:

$$voxelValue_V = v_1 + (v_2 - v_1) \cdot \frac{r + d_V}{2 \cdot r} \qquad (5.4)$$

Logically, the impact that these new reference-object voxels have on voxel values of nearby voxels that are not inside the reference object is reduced. Equation 5.3 must be adapted accordingly (see Algorithm 2).

According to Algorithm 2, the erosion kernel carrying the distance information is centered on voxel boundaries, more specifically, on the boundary between any pair of neighboring (6-neighborhood) voxels of which exactly one is part of the original object. These points are also targeted by the smoothed object boundary. Since, on the other hand, voxel values are assigned to voxel centers, the reference object is actually a set of voxel centers. Hence the distances from the reference object to the smoothed surface will, on average, be approximately $r + 0.5 \cdot vSize$, with $vSize$ being the side-length of a voxel, provided that voxels are cube-shaped (as in *isotropic* volumes). For anisotropic volumes, using the inter-voxel distance within slices still provides a sound approximation.

Equation 5.4 assumes that the average distance is approximately equal to r. A more accurate result can be acquired using Equation 5.5:

$$voxelValue_V = v_1 + \frac{v_2 - v_1}{2} \cdot (1 + \frac{d_V}{r + 0.5}) \qquad (5.5)$$

See Figure 5.5 for an example: If the outer layer of voxels is removed from the original object depicted in the left image, only the four voxels marked red in the center image will remain. The two leftmost voxels of the original object, marked yellow in the center image, have been peeled off and are not adjacent to any of the four voxels that remain. These two voxels thus constitute a feature of the object shape which will be lost, if the final reference object consist only of the four voxels remaining after erosion, as would be the case if Algorithm 1 was used. The final reconstructed object would, in this example, be approximately a circle (see right image, Figure 5.5). A significant feature (the bulging on the left hand side of the original object) would be lost.

If Algorithm 2 is used, the two leftmost voxels are returned to the reference object. Thus the feature can be retained, despite smoothing, as illustrated in Figure 5.6.

As an additional measure of security, the in-out classification of voxels with respect to the object can be artificially maintained. This is done in a postprocessing step: All voxels that are inside the original segmented object and acquired a value smaller than the iso-value v_{iso} are detected and their values are set to $v_{iso} + \delta$. Analogously, voxels outside the original object whose values are greater than v_{iso} are assigned the value $v_{iso} - \delta$. This, logically, compromises the smoothing effect, but still yields significantly smoother results compared to non-filtered objects.

Algorithm 3 Algorithm 2 may lead to significant errors, if a part of the original object is approximately cone-shaped. From the thin end of such a structure towards the wide end (the base of the cone), the distance to the boundary of the original object increases, but only slightly, since the smallest distances are to the side-walls of the cone. Erosion using a
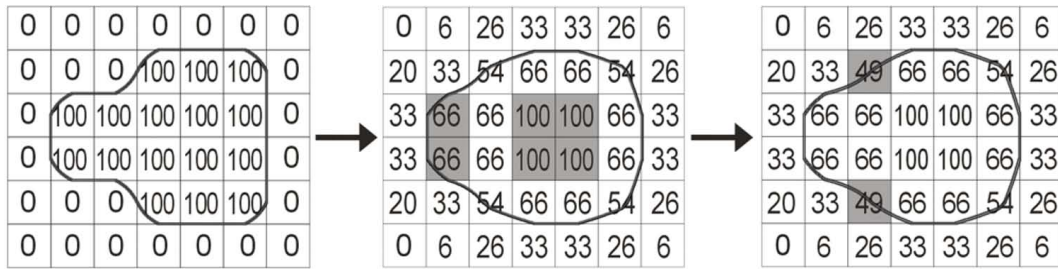
Figure 5.6: Surface smoothing: $v_1 = 0$, $v_2 = 100$, $v_{iso} = 50$; Left: isosurface initialized; Center: the isosurface has been smoothed using Algorithm 2, r=1. Voxels marked gray are part of the reference object; Right: The in-out classification of voxels is restored. The voxels marked gray were set to a value smaller than v_{iso}, because they were not part of the original object.

sufficiently large kernel will demolish large parts of the cone-shaped structure. Since most voxels inside the cone have a neighbor with a (slightly) larger distance to the object surface, algorithm 2 will not notice the significant shape change. This problem can be solved by adding to the reference mask any voxel V with a distance value d_V, which does not have a neighbor whose distance to the object boundary exceeds $d_V + d_{delta}$ for some $d_{delta} > 0$. Algorithm 3 illustrates this improved method.

5.3.3 Evaluation

Figure 5.7 compares results of the two algorithms presented here with a non-filtered object on part of a cross section of a segmented object. The voxel values resulting from object reconstruction are encoded in red color. White indicates the iso-surface.

The left image shows an object whose boundary was not smoothed. Voxels belonging to the binary segmented object are indicated by the light red color. Black parts of the image represent voxels outside the binary segmented object. A significant portion of the iso-surface (white) is located exactly on voxel boundaries, resulting in blocking artifacts. For the generation of the center image, Algorithm 1 was used, with a erosion kernel with radius $r = 2$. As could be expected, the smoothing is tremendous, but details are lost. The image on the right hand side was generated using Algorithm 2, also with an erosion kernel with radius $r = 2$. Smoothing of homogeneous regions is significant. Yet, important features of the object shape are retained to a large extent.

Figure 5.8 compares 3D results of algorithms 1 and 2 with an unfiltered object. It shows that algorithm 2 with maintenance of the in-out classification of voxels retains surface features and still smoothes the object significantly.

Figure 5.9 compares the results of applications of Algorithm 2 with maintenance of in-out classifications of voxels with different values for the radius r of the erosion kernel.

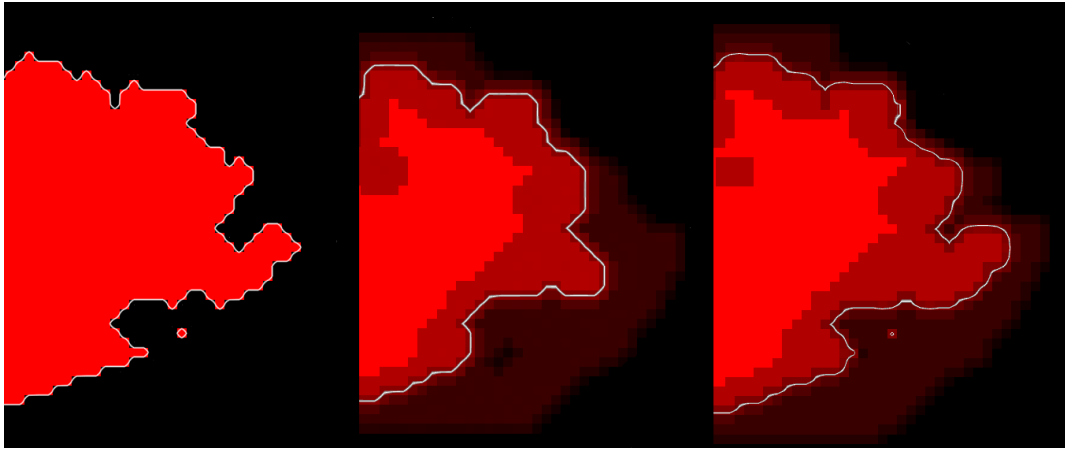Figure 5.10 provides result images for Algorithm 3 using different values for the parameter d_{delta}.

Figure 5.7: Comparison of smoothing techniques. The red color indicates voxel value magnitude, white indicates the iso-surface. Left: No smoothing; Center: Algorithm 1 (r=2) was used for smoothing; Right: Algorithm 2 (r=2), with maintenance of in-out classification of voxels, was used
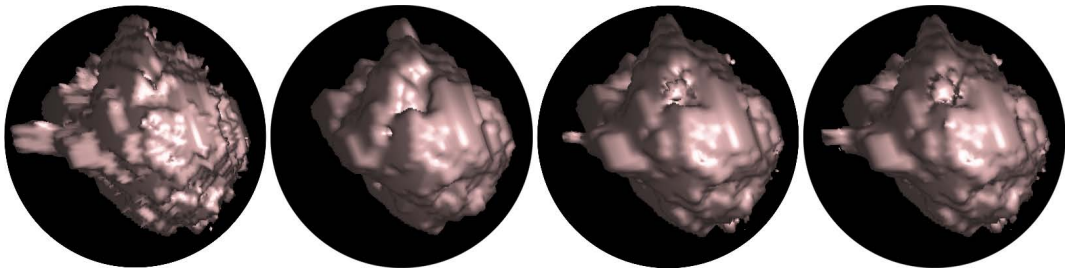

Figure 5.8: Left: No smoothing; Center left: Algorithm 1 (r=2) was used for smoothing; Center right: Algorithm 2 (r=2) was used, without maintenance of in-out classification of voxels; Right: Algorithm 2 (r=2) was used, with maintenance of in-out classification of voxels
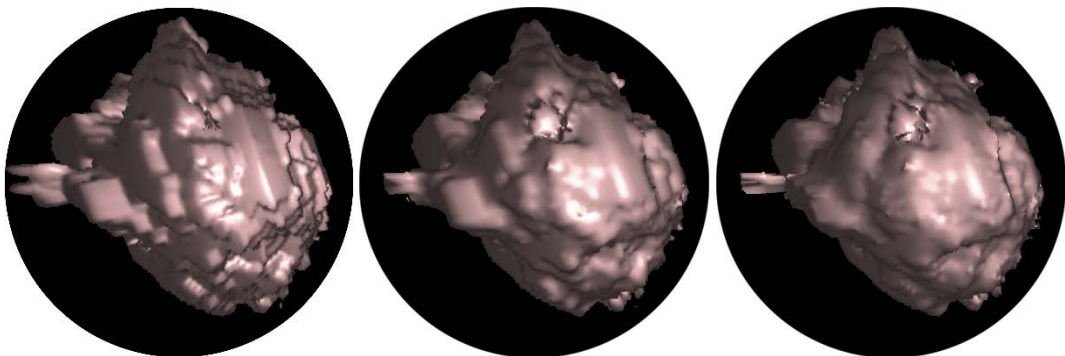
Figure 5.9: Comparison of filtering intensities: For each of these images, Algorithm 2 with the maintenance of in-out classifications of voxels was used, with different values for the radius r of the erosion kernel. Left: $r = 1$; Center: $r = 2$; Right: $r = 3$.

Algorithm 2 Algorithm for reconstruction and smoothing of a binary segmented object, constrained to retain the principal object shape

---

1: Algorithm input:

$O_{original}$: original (unsmoothed) object

$r > 0$: extent (radius) of object erosion

$\delta > 0$: delta-value for maintenance of in-out classification of voxel centers

2: initialize the reference object $O_{reference}$ as equal to the original object $O_{original}$

3: generate a spherical erosion kernel K with radius r

4: augment K with distance information (the distance to the center of the kernel)

5: **for** each pair V_1, V_2 of neighboring voxels with $V_1 \in O_{original}$ AND $V_2 \notin O_{original}$ **do**

6:     Place K such that C_{kernel}, the center of K, coincides with the midpoint of the line segment connecting V_1 and V_2

7:     **for** each voxel V covered by K **do**

8:       **if** $V \in O_{original}$ **then**

9:         adjust the distance value d_V stored for voxel V:

        $d_V \leftarrow MIN(d_V, distanceBetween(V, C_{kernel}))$

10:         **if** $V \in O_{reference}$ **then**

11:           remove V from $O_{reference}$

12: **for** each voxel V inside $O_{reference}$ **do**

13:     $voxelValue_V \leftarrow v_2$

14:     $d_V \leftarrow r + 0.5$ {the remaining voxels in the reference object obtain max distance}

15: **for** each voxel V with $V \in O_{original}$ AND $V \notin O_{reference}$ **do**

16:     **if** no neighbor voxel N of V was assigned a distance value $d_N > d_V$ **then**

17:       add V to $O_{reference}$

18:       $voxelValue_V \leftarrow v_1 + \frac{v_2 - v_1}{2} \cdot \left(1 + \frac{d_V}{r+0.5}\right)$

19: generate distance transform masks for possible radii between 1 and $(2 \cdot r)$

20: **for** each pair V_1, V_2 of neighboring voxels with $V_1 \in O_{reference}$ AND $V_2 \notin O_{reference}$ **do**

21:     Center a distance transform mask M with the radius $r_M = d_{Vref} \cdot 2$ to the midpoint P of the line segment connecting V_1 and V_2

22:     **for** each voxel V covered by M, with $V \notin O_{reference}$ **do**

23:       $voxelValue_V \leftarrow$

      $MAX(voxelValue_V,$

      $v_1 + (v_2 - v1) \cdot (1 - (distanceBetween(V, P) - d_{V1} + r + 1)/(2 \cdot r + 1)))$

24: {this (optional) last part of the algorithm ensures the maintenance of the in-out classification of voxel centers}

25: **for** each voxel V **do**

26:     **if** $V \in O_{original}$ AND $voxelValue_V < v_{iso}$ **then**

27:       set $voxelValue_V \leftarrow v_{iso} + \delta$

28:     **if** $V \notin O_{original}$ AND $voxelValue_V < v_{iso}$ **then**

29:       set $voxelValue_V \leftarrow v_{iso} - \delta$

Algorithm 3 Improved algorithm for reconstruction and smoothing of a binary segmented object, constrained to retain the principal object shape

---

1: Algorithm input:

    $O_{original}$: original (unsmoothed) object

    $r > 0$: extent (radius) of object erosion

    $d_{delta} > 0$: minimum voxel-to-voxel increase of the distance to the object surface

2: initialize the reference object $O_{reference}$ as equal to the original object $O_{original}$

3: generate a spherical erosion kernel K with radius r

4: augment K with distance information (the distance to the center of the kernel)

5: **for** each pair V_1, V_2 of neighboring voxels with $V_1 \in O_{original}$ AND $V_2 \notin O_{original}$ **do**

6:    Place K such that C_{kernel}, the center of K, coincides with the midpoint of the line segment connecting V_1 and V_2

7:    **for** each voxel V covered by K **do**

8:      **if** $V \in O_{original}$ **then**

9:        adjust the distance value d_V stored for voxel V:

            $d_V \leftarrow MIN(d_V, distanceBetween(V, C_{kernel}))$

10:      **if** $V \in O_{reference}$ **then**

11:        remove V from $O_{reference}$

12: **for** each voxel V inside $O_{reference}$ **do**

13:    $voxelValue_V \leftarrow v_2$

14:    $d_V \leftarrow r + 0.5$ {the remaining voxels in the reference object obtain max distance}

15: **for** each voxel V with $V \in O_{original}$ AND $V \notin O_{reference}$ **do**

16:    **if** no neighbor voxel N of V was assigned a distance value $d_N > d_V + d_{delta}$ **then**

17:      add V to $O_{reference}$

18:      $voxelValue_V \leftarrow v_1 + \frac{v_2 - v_1}{2} \cdot \left(1 + \frac{d_V}{r+0.5}\right)$

19: generate distance transform masks for possible radii between 1 and $(2 \cdot r)$

20: **for** each pair V_1, V_2 of neighboring voxels with $V_1 \in O_{reference}$ AND $V_2 \notin O_{reference}$ **do**

21:    Center a distance transform mask M with the radius $r_M = d_{Vref} \cdot 2$ to the midpoint P of the line segment connecting V_1 and V_2

22:    **for** each voxel V covered by M, with $V \notin O_{reference}$ **do**

23:      $voxelValue_V \leftarrow$

      $MAX(voxelValue_V,$

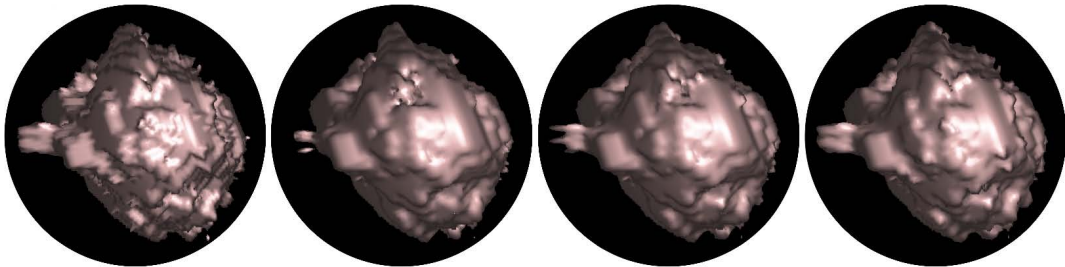 $v_1 + (v_2 - v1) \cdot (1 - (distanceBetween(V, P) - d_{V1} + r + 1)/(2 \cdot r + 1)))$

Figure 5.10: Results of Algorithm 3 (r=2); Left: no smoothing; Center Left: $d_{delta} = 0.1$; Center Right: $d_{delta} = 0.3$; Right: $d_{delta} = 0.5$.

Chapter 6

STEPS - Rendering Techniques

To be applicable, a virtual endoscopy system must provide instant reaction to user input. This results in the need for high frame rates and thus fast visualization. A review of existing techniques for accelerated visualization is given in chapter 4.

For STEPS, two optimized first-hit ray casting techniques, both without special hardware acceleration, are applied. Entirely CPU-based algorithms are applied for reasons of hardware independence. Furthermore, the rendering techniques used by STEPS work directly on the data volumes obtained from the radiological modality or generated during object reconstruction. No intermediate object representations (e.g., polygon meshes) are used, ensuring the highest possible amount of flexibility. This, for example, allows interactive adjustment of iso-values.

One of the applied ray casting techniques is used to depict the foreground (the inner walls of the investigated cavities, an isosurface in the original CT data) and one to visualize the pre-segmented background objects. In this chapter, the generation of the virtual endoscopic images will be explained.

Since the foreground always covers the complete image and thus each pixel has to be processed, an optimized image-order (pixel by pixel) ray casting technique is applied [NWF$^+$04]. The technique achieves interactive frame rates by exploiting inter-pixel coherency and accelerating memory access and empty space traversal. Rays are tracked efficiently from cell-boundary to cell-boundary [AW87] in order to avoid missing thin structures.

Background visualization is performed on a synthetic background volume which is generated by reconstructing objects of interest from binary segmentation masks. For objects which can be represented by an isosurface in the original MRA or CT data (e.g., the ICAs in the MRA data set), the segmentation result comprises the objects and its surroundings - the original values of voxels inside the segmented object are simply transferred to the background volume and the object boundaries can be extracted during the virtual endoscopy by interactively adjusting the iso-value for the object. Other objects are reconstructed by generating synthetic isosurfaces. The boundaries of these objects are defined by the binary segmentation masks and the amount of smoothing applied.

Background objects cover a varying number of pixels, depending on their shapes and their positions relative to the viewing frustum. Therefore the background is generated by applying the optimized object-order algorithm *cell-based first-hit ray casting* [NFW$^+$04, NMHW02].

For each object an isosurface is displayed, the threshold can be adapted individually for each object.

In each frame, the two resulting images (foreground and background) are merged using a z-buffer. The difference of the z-values between foreground and background determines foreground opacity.

6.1   Foreground Rendering

The foreground is rendered using image-order first-hit ray casting. To achieve interactive frame rates, several acceleration techniques are applied:

Exploitation of Pixel-Space Coherence: To speed up foreground rendering during phases of interaction, the image is subdivided into tiles of 8×8 pixels and a ray is cast for each corner of a tile. If the colors obtained from the cast rays are sufficiently similar, the remaining pixels of the tile can be computed from the corner pixels using bilinear interpolation. Otherwise the tile is recursively subdivided and further rays have to be cast for the sub-tiles [Lev90b]. Using this approach, rays have to be cast for only 15-35 percent of all pixels in a typical endonasal view. This strategy occasionally leads to artifacts, which are, however, hardly noticed during interaction. This optimization is not used when rendering the final quality image when the viewing frustum remains still.

Acceleration of Memory Access: Retrieving volume data along an arbitrarily oriented ray usually exhibits low utilization of the CPU cache. To reduce the number of accesses to the data volume, a binary acceleration volume, the *void volume*, consisting of 1 bit per voxel, is used. It marks volume cells which do not contain the object's surface. During ray traversal, marked cells can be simply skipped.

Space Leaping: A second binary acceleration volume, the *leap volume*, marks cells whose distances from the isosurface exceed a value d. Whenever a marked cell is encountered during ray casting, the ray can be advanced by a distance of d without missing the surface. The leap volume approximates the benefits of distance fields (see Section 4.1.2) at much lower memory cost.

The acceleration volumes have to be recomputed after a change of the iso-threshold. The computation, however, is fast: the void volume is generated trivially by checking the voxel values for each cell and the leap volume is calculated by eroding the void volume. During threshold adjustment the acceleration volumes are not used resulting in a slight drop of the frame rate (by approximately 10 percent). The time that it takes for the user to switch between threshold adjustment and any other interaction is usually sufficient to recalculate the acceleration volumes.

In the vicinity of the isosurface rays are tracked efficiently from cell-boundary to cell-boundary [AW87] in order to avoid missing thin structures.

6.1.1 Display of Surface Rigidity

A smoothly shaded isosurface as generated by first-hit ray casting gives no information apart from shape. All structures, regardless of the types of tissue they represent, are displayed in the same way and the same color. The information given can be enhanced by adding meaningful texture to the surface. Information about tissue rigidity, for example, helps the surgeon in a variety of ways. First of all, the intuitive mapping between the 3D view and reality is enhanced. In real endoscopy it is easy to differentiate between soft and rigid tissue - if there are not sufficient visual hints, a structure can be tested for rigidity quite easily by checking the deformability. This can not yet be done in STEPS. However, showing surface rigidity as a texture reduces that disadvantage.

Furthermore, if the simulation is suspended (see Section 7.4.4), the texture can help identify why endoscope rotation is prevented in a certain direction. The texture also gives information about isosurface reliability: the more rigid the displayed structure is, the steeper is the gradient in the region of the isosurface, implying a smaller change of the surface as a response to iso-value adaptation. On the other hand, parts of the surface which are far from rigid structures, can be expected to change significantly, if the iso-value is slightly changed. STEPS can indicate tissue rigidity in various ways, namely volume rendering, double-iso-surfacing and simple color-mapping. Each of these has its advantages and disadvantages.

Volume Rendering

Volume rendering is performed by extending each foreground viewing ray a few steps beyond the isosurface, using a transfer function to accumulate a pixel color. This accumulated color is then blended with the color assigned by foreground iso-surfacing. The extended ray is stopped either after reaching the user-defined maximum number of traversal steps, after reaching full opacity, after encountering a data value lower than the foreground iso-value, or after entering a background object. Volume rendering is the most flexible (the transfer function can be adapted by the user) but slowest alternative, increasing rendering cost by 20 to 80 percent, depending on the maximum number of accumulation steps.

Volume rendering provides valuable information about bony structures, as illustrated in Figure 6.1. Figure 6.2 shows a lateral view of the sphenoid sinus enhanced by volume rendering. The sellar floor is displayed in the top left region of the image.

Since, in this application, volume rendering is used for the sole purpose of visualizing bony structures, STEPS provides a simplified user interface for adjusting the transfer function. Opacity is zero for all intensity values below the foreground iso-value and then increases linearly up to 0.2. The size of the interval in which opacity increases can be adapted using a slider. Within the same interval, the color changes from a user-defined value (default: red) to white. A second slider is used to adjust the transparency of the isosurface.

Double Iso-Surfacing

Double iso-surfacing is performed by defining a second foreground iso-value v_{iso2}, representing the transition between soft tissue and bone. Similar to volume rendering, rays are extended by a limited number of steps and stopped, when a background object is entered.
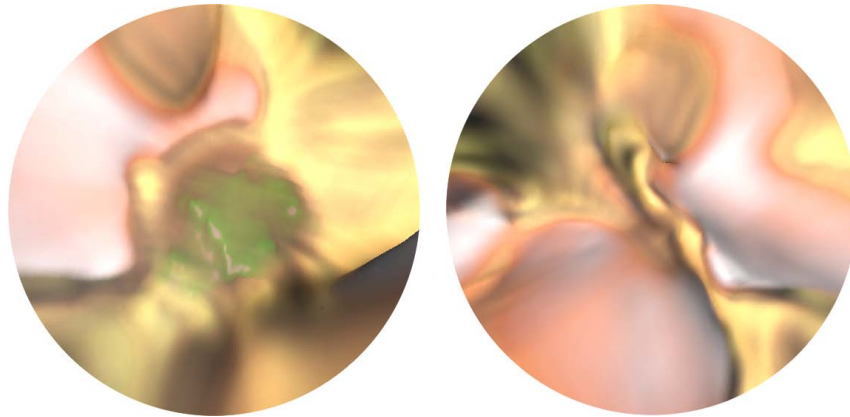
Figure 6.1: Volume rendering: Left: The tumor (green) is located beyond a thin bony structure, the sellar floor; Right: The bone structure of an incomplete sphenoidal septum

An isosurface intersection is performed as soon as a data value larger than v_{iso2} is encountered. Double iso-surfacing gives a good impression of the proximity of bone and works faster than volume rendering (speed decreases by about 15 to 60 percent), but generates less intuitive images.

Color Coding

The third method, simple color-coding, is the fastest one with a speed decrease of about 10 percent: Tissue rigidity is mapped to surface color. The color scale ranges from red (soft tissue) to white (massive bone). Again each foreground ray is elongated by a limited number of steps. After each step, the tissue density at the current ray position is sampled using trilinear interpolation. The calculated value is decreased by $c \cdot d$, with c being a positive constant and d the distance to the foreground isosurface. This emphasizes rigid tissue that is very close to the isosurface. The largest value encountered for each ray is used to calculate the texture color. In order to add to the understanding, the number of sampling steps and the constant c, the sensitivity to distance can be adjusted by the user.

Comparison

Figure 6.3 shows example images for all three methods and a slice of original CT data showing part of the scene depicted in the 3D renderings. The number of ray steps beyond the isosurface was 20 for the volume rendering and 10 for the other two methods. Currently we consider simple color mapping and volume rendering the most effective methods. Color mapping provides an intuitive visual result with little computational overhead. Volume rendering is more expensive, but gives a better impression of distances between bones and the surface. Both techniques, especially volume rendering, in fact, were accepted very well by the neurosurgeons.

One reason that volume rendering and double iso-surfacing are significantly slower than color coding is that they produce images of lower homogeneity than color coding. There is
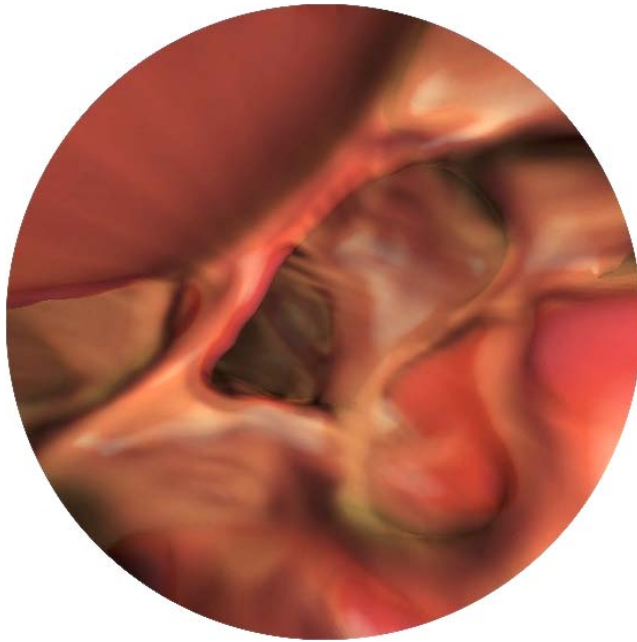
Figure 6.2: Volume rendering: Display of bony structures behind the semi-transparent surface of the sphenoid sinus

more information, leading to high-frequency discontinuities in the image. Therefore there is less inter-pixel coherency and the tile-based rendering acceleration technique described at the beginning of section 6.1 becomes less effective. Typically, if volume rendering is used, about half of all pixels acquire their final colors through ray casting, and half through interpolation. In the current implementation tiles of 4×4 pixels are used for volume rendering, rather than 8×8 pixels.

6.1.2   Depth Shading

For the endonasal approach, an endoscope with a relatively small diameter (4mm or less) must be used to maximize maneuverability. Thin endoscopes, however, have the disadvantage of reduced light transport capabilities. The range of light emitted by the endoscope is therefore very limited, as can be seen in Figure 3.7.
STEPS uses depth shading to simulate this phenomenon and to enhance depth perception. The left image in figure 6.4 was rendered without using depth shading. Due to the shaded surface, depth is anticipated, but only very vaguely. By darkening the scene with increasing distance from the eye point (Figure 6.4, center and right images), depth is communicated much more effectively.
The user can interactively adjust the intensity of depth shading.
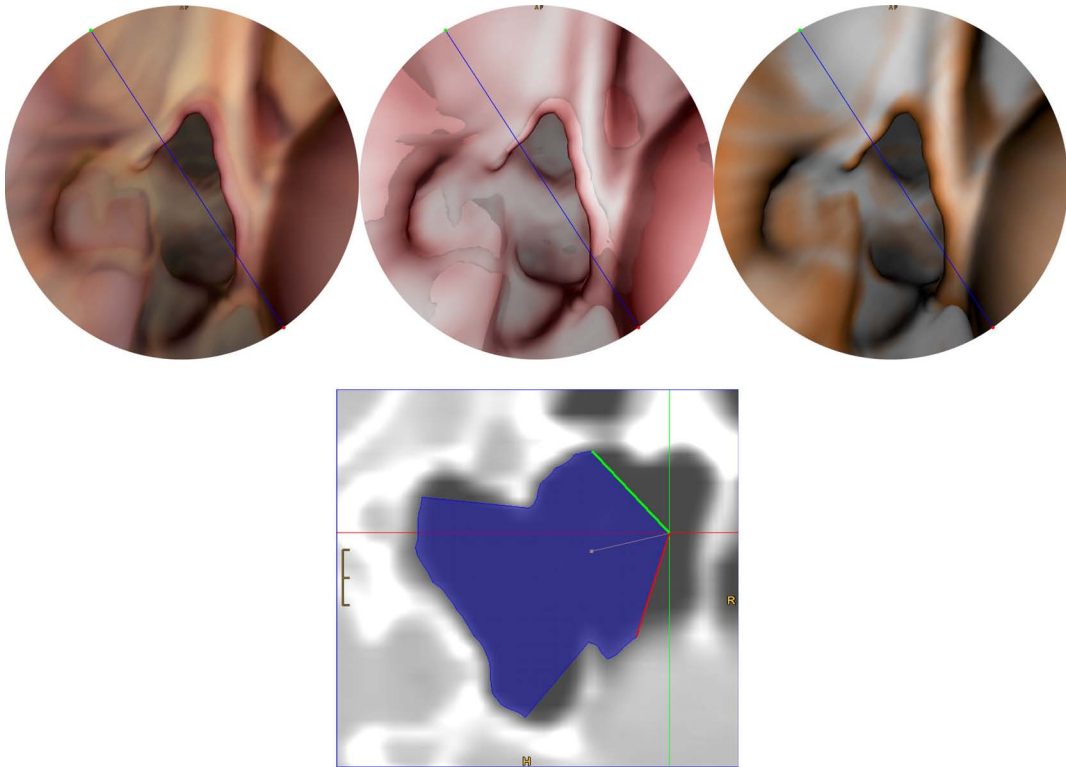
Figure 6.3: Displaying surface rigidity: Top Left: volume rendering; Top Center: double iso-surfacing; Top Right: color-coding; Bottom: section through original CT data, the blue lines in the 3D images indicate the intersection with this CT slice (see Section 7.2)

6.2 Background Rendering

Background rendering is performed using the *Cell-Based First Hit Ray Casting* technique [NMHW02, NFW$^+$04]. This algorithm works in an object-order fashion and is therefore well suited especially for the display of background objects. Section 6.2.1 briefly describes the basic method. Section 6.2.2 deals with rasterization, the core part of the algorithm. Acceleration strategies are introduced in Sections 6.2.3 through 6.2.7.

6.2.1 Cell-Based First-Hit Ray Casting

Cell-based first-hit ray casting is roughly described in Algorithm 4: The volume extracted from the CT scan of the patient's head is divided into cubic bricks, referred to as *macro-cells*. These bricks form the leaves of a min-max-octree, implemented as an integer-array as proposed by Wilhelms and van Gelder [WG90]. This octree can be efficiently traversed to find all macro-cells containing a part of an isosurface in order of increasing distance from the eye-point. Each of those macro-cells is then projected to the image plane. The projection is rasterized to the screen, scan line by scan line. For each pixel which is covered by the projection and has not yet been assigned a color, a local ray segment is tracked inside the
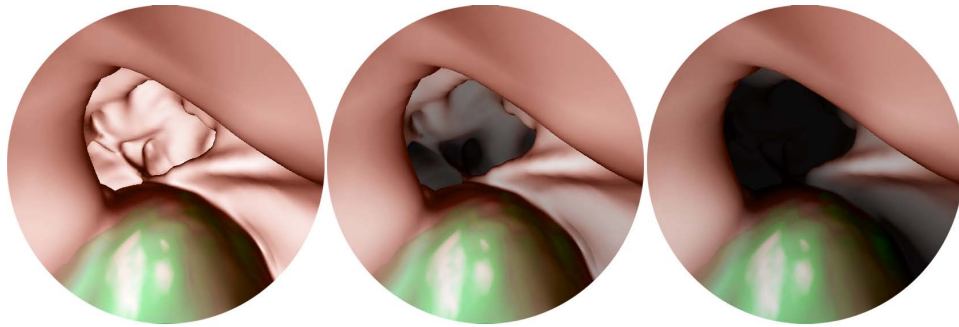
Figure 6.4: Left: no depth shading; Center: low intensity depth shading; Right: high intensity depth shading

macro-cell using the fast voxel traversal technique by Amanatides and Woo [AW87]: The ray position is efficiently propagated from one cell-boundary to the next, therefore quickly identifying all cells pierced by the ray. This traversal algorithm requires costly calculations solely for ray initializations. When a cell which contains a part of an isosurface is encountered, an intersection test is performed. If an intersection exists, the surface normal at the intersection point is calculated and the pixel is shaded accordingly. Some optimizations of the algorithm were proposed [Neu01]:

Macro-cell trimming: Instead of the complete macro-cell, only the smallest cuboid containing the isosurface inside the macro-cell is projected, decreasing the number of local ray segments having to be tracked. Also, local ray segments are confined to this smallest cuboid. This latter optimization was partly discarded to allow for a more effective one described in section 6.2.5.

Screen regions: Small rectangular regions of the screen which have already been completely filled, can be skipped by rasterization.

The main principle of acceleration in the original cell-based first-hit ray casting algorithm [Neu01] is the reduction of ray traversal distances. Sections 6.2.3, 6.2.4, 6.2.6 and 6.2.7 will explain, how pixel-space coherency can be exploited for further speed-up. Sections 6.2.5 and 6.2.8 will introduce optimization techniques which increase the number of ray steps tracked, but still speed up rendering by finding compromises satisfying the trade-off between the ray traversal time saved and the cost associated with saving it. Figure 6.5 shows an overview of the algorithm and points out which steps of the original method are adapted to allow the proposed optimizations.

6.2.2 Rasterization of Macro-Cell Projections

After a macro-cell has been projected to the image plane, a rasterization process, traversing the projection scan line by scan line, determines the set of pixels, for which local ray segments have to be tracked through the macro-cell. This process is explained by Algorithm 5. Depending on the view point, three faces of the current macro-cell are visible at most. Each projected macro-cell vertex is assigned a *visibility coefficient* indicating how many of the

Algorithm 4 Cell-Based First-Hit Ray Casting

1: Algorithm input:
   min-max octree encoding the background volume
2: process root of min-max octree
3: **for** the current node **do**
4:    **if** the node is a leaf node (represents a macro-cell) **then**
5:       project vertices of according macro-cell
6:       rasterize the macro-cell (see Algorithm 5)
7:    **else**
8:       arrange child nodes in correct order
9:       **for** $i = 0$ to 7 **do**
10:         **if** min value (of child node i) $\leq v_{iso}$ AND max value (of child node i) $\geq v_{iso}$
             then
11:           process child node i (go to line 2)
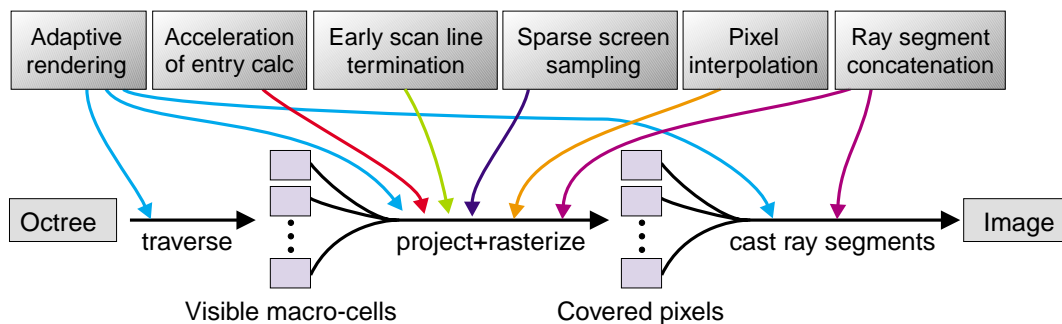
Figure 6.5: Optimization strategies for cell-based first-hit ray casting. Most affect the rasterization of macro-cell projections.

faces adjacent to the vertex are visible. If the visibility coefficient is 1 or 2, the vertex is part of the outline of the projection. Those vertices are interconnected by *silhouette lines*. Vertices with visibility coefficients of 2 or 3 define the end points of *frontier lines*. Figure 6.6 shows an example projection, illustrating silhouette lines, frontier lines and visibility coefficients.

Most acceleration techniques introduced in this chapter affect the rasterization process (see overview in Figure . Algorithms 6 and 7 explain the changes of the rasterization algorithm due to the optimization strategies.

6.2.3 Acceleration of entry point calculation

The starting point of a local ray segment (the entry point of a ray into the macro-cell) is determined by intersecting the ray with the macro-cell. If it is known, through which face (on which side) the ray enters the macro-cell, only a ray-plane (instead of ray-cuboid) intersection must be calculated. Using a simple algorithm (see algorithm 6), which determines the entry face during rasterization can therefore save valuable rendering time:

The macro-cell face that projects to the ray launched through the first pixel of a scan line
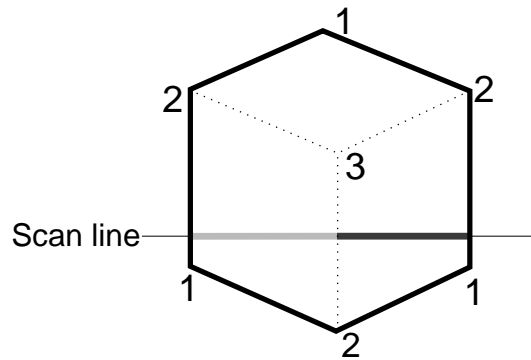
Figure 6.6: Silhouette lines (solid), frontier lines (dotted), visibility coefficients and an example scan line
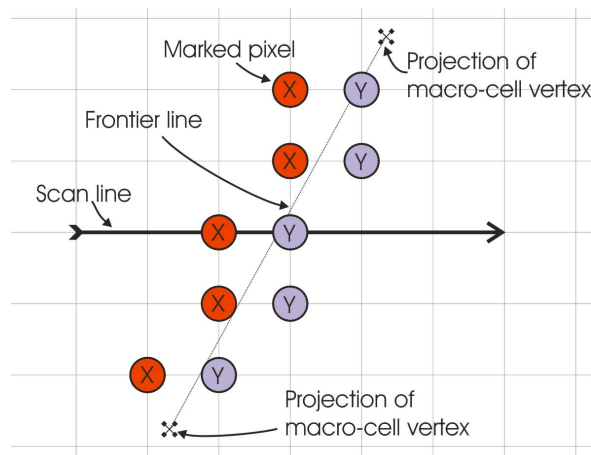

Figure 6.7: The pixels next to the rasterized frontier line are marked as frontier pixels. The macro-cell face projecting to each marked pixel is stored. In this example, the frontier line separates the projections of the x-face and the y-face of the macro-cell.

is determined during the rasterization of the silhouette lines. Additionally, the frontier lines are rasterized to determine, when, during processing of the scan line, the entry face changes. Figure 6.7 shows an example: The dotted line is a frontier line, separating the *x-face* (the visible face whose vertices have a common x-coordinate) from the *y-face*. During rasterization, the pixels adjacent to the frontier line are marked with the faces projecting to them (here either x or y). These pixels are called *frontier pixels*. During rasterization of a scan line, as soon as a frontier pixel is reached, the face projecting to it is used as the ray entry face from that moment on, until the next frontier pixel is encountered or the scan line is finished.

The overall rendering time saved due to the faster intersection tests amounts to about 8 percent on the average.

Algorithm 5 Rasterization of macro-cell projection. Scan lines are processed from left to right.

1: Algorithm input:
   world- and screen-coordinates of the vertices of the projected macro-cell
2: determine set F_v of visible macro-cell faces
3: depending on F_v, determine set L_s of silhouette lines
4: rasterize L_s using the Bresenham algorithm. This reveals the left and right boundaries of the macro-cell projection for each scan line.
5: **for all** scan lines intersecting the macro-cell projection **do**
6:   $x_1 \leftarrow$ left boundary of macro-cell projection in this scan line
7:   $y_1 \leftarrow$ right boundary of macro-cell projection in this scan line
8:   **for** $i = x_1$ to x_2 **do**
9:     **if** pixel i of the scan line has not been finalized **then**
10:      calculate ray direction for pixel i
11:      intersect the ray with the macro-cell to find the entry point p_{entry}
12:      track local ray segment inside the macro-cell, starting at p_{entry}
13:      **if** an intersection with the iso-surface has been found **then**
14:        finalize pixel i

6.2.4 Early Scan Line Termination

Early scan line termination (EST) is a heuristic measure yielding a reduction of the number of ray segments tracked. Based on an analysis of the isosurface geometry inside the macro-cell, a decision is made, whether rasterization of a scan line can be stopped as soon as a ray segment did not intersect the isosurface.

Depending on the geometry of the isosurfaces inside the macro-cell, scan lines can be either rows or columns of pixels and can be processed in either direction (vertical scan lines, for example, can be processed top-down or bottom-up). Very often, processing of a scan line can be stopped as soon as a local ray through a pixel of this scan line did not intersect an object's surface. This optimization depends heavily on the direction of scan line processing: In the case of the macro cell depicted in Figure 6.8, scan lines should be processed from left to right to allow effective application of early scan line termination. In the example depicted in Figure 6.9, processing should occur bottom-up. If scan lines are processed from left, early scan line termination will lead to errors. The parts of the iso-surface whose normals face in direction of scan line origin (here: towards the left), will be lost.

The principal (average) direction of surface normals inside a macro-cell determines the direction of scan line processing. This, of course, does not render early scan line termination completely safe. Therefore, after deciding upon a direction in which scan lines are processed, an analysis of the isosurface geometry inside the macro-cell [Neu01, NMHW02] is performed to generate a *critical region* in which early scan line termination is forbidden, to reduce the probability of errors. Efficient algorithms for this task are based on heuristics and lack the accuracy to remove all errors. Most of the remaining errors are, however, removed by ray segment concatenation (see Section 6.2.5) or by tracing rays backwards, if a macro-cell is entered on the far side of the isosurface (see Figure 6.12).

Unfortunately, early scan line termination is in general less effective when visualizing only

Algorithm 6 Rasterization of macro-cell projection, with accelerated entry point calculation. Scan lines are processed from left to right.

1: Algorithm input:
   world- and screen-coordinates of the vertices of the projected macro-cell
2: determine set F_v of visible macro-cell faces
3: depending on F_v, determine set L_s of silhouette lines
4: depending on F_v, determine set L_f of frontier lines
5: rasterize L_s using the Bresenham algorithm. This reveals the left and right boundaries of the macro-cell projection for each scan line. This also identifies the macro-cell faces projecting to the left boundary of the macro-cell projection for each scan line
6: rasterize L_f using the Bresenham algorithm. This reveals the set of frontier pixels, i.e., pixels which constitute the frontiers between the projections of macro-cell faces)
7: **for all** scan lines intersecting the macro-cell projection **do**
8:    $activeFace \leftarrow$ macro-cell face projecting to the left boundary of the macro-cell projection in the current scan line
9:    $x_1 \leftarrow$ left boundary of projection in this scan line
10:    $y_1 \leftarrow$ right boundary of projection in this scan line
11:    **for** $i = x_1$ to x_2 **do**
12:       **if** pixel i of the scan line is a frontier pixel **then**
13:         $activeFace \leftarrow$ macro-cell face projecting to pixel i
14:       **if** pixel i has not yet been finalized **then**
15:         calculate ray direction for pixel i
16:         intersect the ray with $activeFace$ to find the entry point p_{entry}
17:         cast local ray segment inside the macro-cell, starting at p_{entry}
18:         **if** an intersection with the iso-surface has been found **then**
19:           finalize pixel i

background objects, compared to its application for generating the foreground for a virtual endoscopy system. Its benefit is further compromised by sparse screen sampling (see Section 6.2.6). Therefore, in the current system, early scan line termination is only applied to macro cells close to the eye point (see section 6.2.8).

To increase the effect of early scan line termination, it might be a feasible idea to subdivide a macro-cell such that each part exhibits one principal surface orientation. This has not yet been tested.

6.2.5 Ray Segment Concatenation

Often a ray moves along the surface of an object for a long distance, and through many macro-cells, until it finally intersects the isosurface, if it does that at all. The actually tracked portion of such a ray consists of a considerable amount of local ray segments, each having to be initialized. Ray segment initialization, including for example entry point calculation and the calculation of parameters used for the fast voxel traversal [AW87], is, summed over all ray segments during one frame, a costly part of the algorithm.

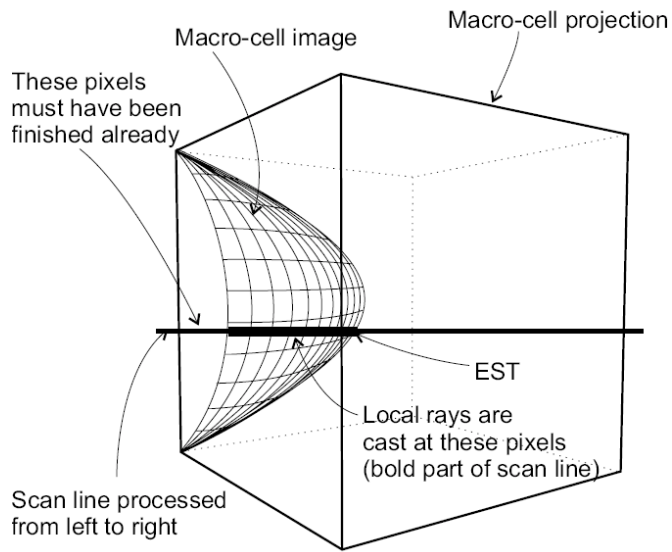It is more efficient to concatenate consecutive ray segments, wherever possible: Whenever

Figure 6.8: Early scan line termination: In this example all scan lines can be terminated early.
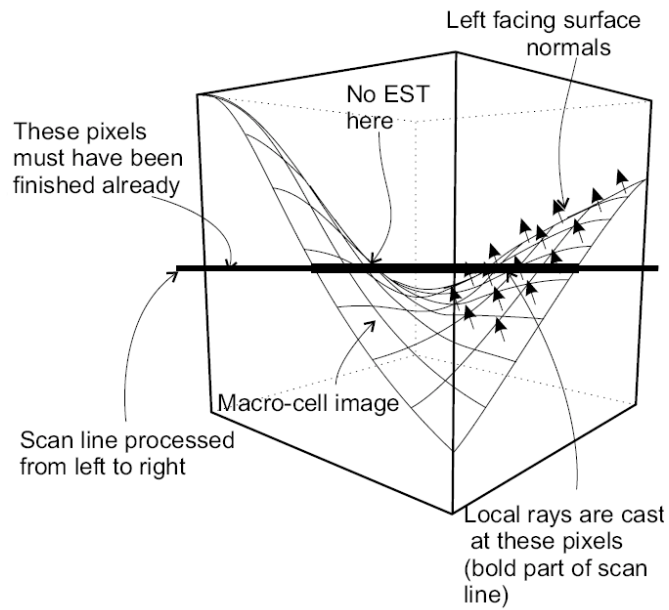

Figure 6.9: The scan line marked in this image cannot be terminated due to left-facing surface normals. Since most surface normals point upwards, the macro cell should be rasterized bottom-up.
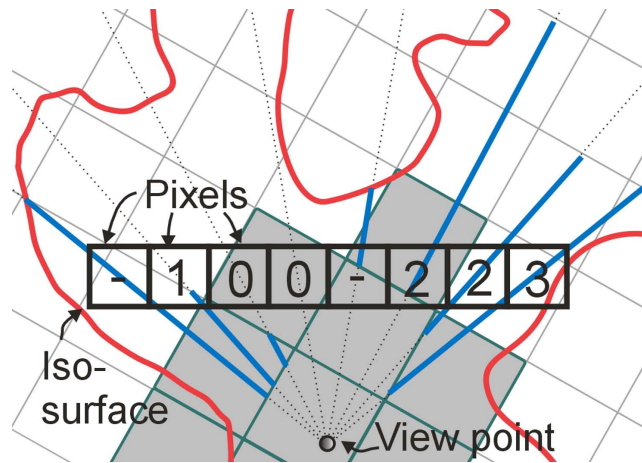
Figure 6.10: Ray segment concatenation and *concat* values.

a ray leaves the macro-cell, the algorithm checks whether the macro-cell now entered by the ray contains a part of the isosurface. In the positive case, the ray segment is tracked further, otherwise ray tracking is stopped.

This algorithm requires some simple extensions of the data structure: So far, the information whether a macro-cell contains a part of an isosurface or not is only coded in the min-max values at the leaves of the octree, where neighborhood information cannot be easily retrieved. An additional bit-volume is introduced, with one bit per macro-cell, indicating, whether the macro-cell contains a part of an isosurface. If the bit is set, the macro-cell will be entered by local ray segments arriving at its boundary.

Another extension to the data structure is needed to avoid ray segments being tracked more than once: For each pixel, a value *concat* is stored. It indicates the number of macro-cells having been traversed in advance for this pixel. Whenever a ray segment is stopped without having intersected the isosurface, the value *concat* of the according pixel is set. The new value of *concat* is set to 0 when the ray segment is launched and increased by 1 whenever a new macro-cell, or, more precisely, a cuboid remaining after macro-cell trimming, is entered.

Figure 6.10 illustrates ray segment concatenation and the calculation of the *concat* values with a 2D-example. Here it is assumed that macro-cell trimming has not been performed and the entire macro-cells are projected. Squares filled with gray color correspond to macro-cells which have already been processed. The viewing plane consists of eight pixels, the numbers indicate the current *concat* value of each pixel ("-" means the pixel has acquired its color already). A viewing ray (dotted line) connects the eye point to the center of each pixel. The bold portions of the viewing rays are the parts which have actually been tracked up to this point in time.

Ray segment concatenation causes the rasterization algorithm to change slightly: For each pixel that is still empty and inside the projection of the currently processed macro-cell, first the *concat* value is checked. Only if it is zero, a local ray segment is started, otherwise *concat* is decreased by 1. Algorithms 7 and 8 outline the rasterization process with ray seg-
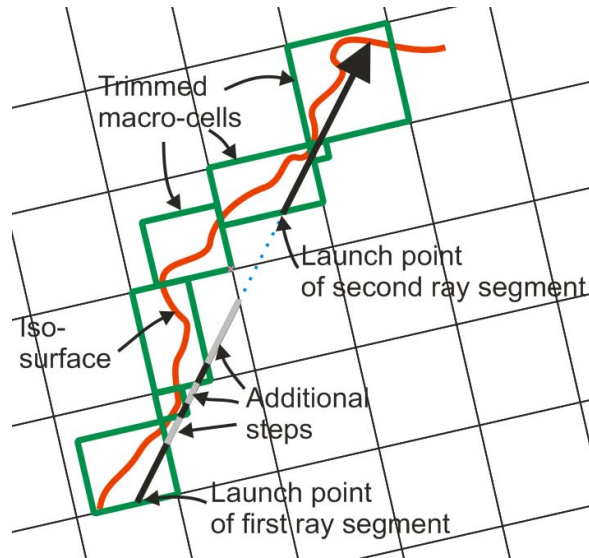
Figure 6.11: Increased ray-tracking distance caused by ray segment concatenation

ment concatenation used.

A drawback of ray segment concatenation is that now ray segments must be tracked all the way to the boundary of the current macro-cell in order to find the entry point to the neighboring macro-cell. Ray tracking can no more be confined to the portion of the macro-cell remaining after trimming. Also, it has to be noted, that not all ray segments that are concatenated to a long ray segment would, if ray segment concatenation was not applied, be tracked. Some would be dropped due to macro-cell trimming or early scan line termination. Still, the impact of this problem is clearly outweighed by the performance gain brought about by ray segment concatenation: The number of ray segments is reduced by about 59 percent. Despite an increase of tracked ray steps by about 6 percent, rendering time decreases by about 17 percent on average.

Figure 6.11 shows the effect of ray segment concatenation on ray tracking distance, with macro-cell trimming having been performed. Bold parts of the depicted ray are actually tracked. The ray depicted in this example passes through six trimmed macro-cells. Without ray segment concatenation, therefore, six local ray segments would have to be launched. Due to concatenation, in this example, number of tracked ray segments is reduced from six to two. The downside is the increased number of tracked ray steps: The parts of the ray which are colored gray would not be tracked in the original algorithm.
A possible alternative to the technique presented in this section would be to keep on tracking the ray, regardless of whether the neighboring macro-cell contains a part of an isosurface, or not. The advantage would be that there would be only one ray segment per pixel at most. Rays which miss the isosurface, however, would have to be tracked all the way to the end of the volume, resulting in massive overhead.

94

6.2.6 Sparse Screen Sampling

The number of local ray segments which fail to intersect the isosurface has already been significantly reduced by applying ray-segment concatenation, macro-cell trimming and early scan line termination. Still, there remains a considerable number of ray segments cast in vain, especially when the eye point is very close to a visualized background object. There are two main reasons for this: Background objects can be arbitrarily small and arbitrarily rough, therefore surface normals may vary at high frequencies, resulting in a large variety of gradient directions inside a macro-cell. This can prohibit early scan line termination completely or confine it to only a small number of scan lines. The second main reason is the considerable magnification induced by perspective rendering, resulting in empty spaces inside a macro-cell being projected to large portions of the screen. The projection of a macro-cell can in fact cover the complete screen, if the eye point is inside or very close to the macro-cell. Traversing big empty regions pixel by pixel is computationally expensive. Applying a sparse sampling strategy can significantly reduce this problem. Reduction of sampling frequency, however, is always associated with some amount of quality reduction, which should be bounded, keeping the following things in mind:

- The quality of the visual output is heavily affected, if pixels defining parts of the outline of a visualized object are not computed correctly.

- The information loss associated with an error in the image is directly proportional to the relation of the size of the erroneous region to the sizes of the projections of structures holding the main image information. A small erroneous region could, for example, completely hide an object whose projection is very small - but it would do only little harm to the information given by a big projection of an object.

In the background rendering algorithm applied by STEPS, sampling frequency is determined by two parameters:

- $skip_{column}$ is the number of pixels skipped by the rasterization process after a local ray segment did not intersect the isosurface

- $skip_{row}$ is the number of scan lines skipped, after all local ray segments of a scan line have missed the isosurface

As soon as an intersection with the isosurface has been found, and therefore a pixel has been assigned its color and finalized, the pixels that were skipped between the previous and the current ray segments have to be processed to find the real object boundary. Also, for the same reason, scan lines skipped between the previously processed scan line and the current one must be processed, if during rasterization of the current scan line intersections with the isosurface have been found.

Due to ray segment concatenation, the probability of parts of objects being lost due to skipping pixels is initially very low. It can be further reduced by tracking ray segments backwards, as illustrated in Figure 6.12. Complete objects whose projections never intersect the sparse sampling grid, might, however, be lost.

In order to keep information loss at a minimum, values $skip_{column}$ and $skip_{row}$ can be adapted to the size of the projection of the macro-cell (see also section 6.2.8). If a macro-cell is projected to a large portion of the screen, one can expect that important structures
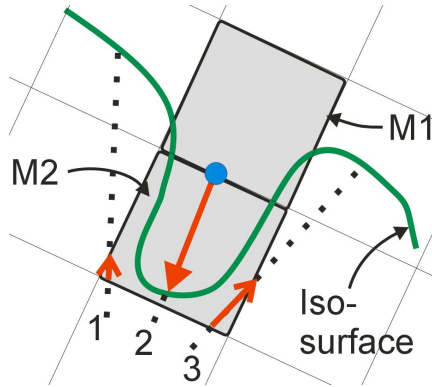
Figure 6.12: Tracking ray segments backwards: During processing of macro-cell M2, rays 1 and 3 encounter no intersection, therefore ray 2 is erroneously skipped. Macro-cell M1 is processed after M2. The entry point of ray 2 into M1 is inside the object - the ray must be traced backwards.

inside this macro-cell will appear large in the final image. Therefore, for near macro-cells, a coarser sampling grid can be applied.

For macro-cells that are far from the view point, the skip values should be kept low in order to find all important objects. This restriction is bearable: As pointed out above, sparse sampling is effective especially for macro-cells close to the eye point - larger skip values for far macro-cells would not significantly reduce rendering times. For the current implementation, $skip_{column}$ and $skip_{row}$ were set to 2 and 1, respectively, for near macro-cells and to 1 and 1, respectively, for far macro-cells. The probability of important objects being lost is therefore extremely low. Sparse screen sampling saves another 23 percent of rendering time on average.

6.2.7 Pixel Interpolation

The number of local ray segments can further be reduced by exploiting the well-known heuristic that a pixel whose neighbors all have similar color can be expected to have a color similar to its neighbors.

The following strategy is applied: Whenever a local ray segment has hit the isosurface and the according pixel is assigned its final intensity i_1, rasterization skips one pixel and the next local ray segment is cast for the next but one pixel in the currently processed scan line. If the resulting intensity i_2 fulfills the similarity criterion

$$\mid i_1 - i_2 \mid < t_{sim}$$

with the predefined threshold t_{sim}, the pixel that was skipped acquires its color through linear interpolation. Otherwise, a local ray segment is cast also for this pixel. After a scan line has been finished, one scan line is skipped and the next but one scan line is processed. After that, rasterization returns to the scan line that was skipped.

As pointed out in the previous section, it is important to correctly compute pixels depicting the outline of the object. Therefore, now ray segments are cast for each pixel that is still

empty and has exactly one vertical neighbor that has been assigned its final color. If both vertical neighbors are filled, the similarity criterion is used to decide whether a ray segment is cast, or linear interpolation is applied.

This strategy saves an average of 8 percent of overall rendering time. A pseudo-code implementation of the rasterization process including sparse sampling, pixel interpolation and ray segment concatenation is given in Algorithm 7.

6.2.8 Adaptive rendering

The algorithm described so far computes all macro-cells in the same way. Only the sampling frequency parameters introduced in section 6.2.6 differ between macro-cells, depending on the sizes of the macro-cell projections. This section shows some more examples of how the size s_p of a macro-cell projection can be used to increase flexibility of the algorithm in order to maximize performance:

Early scan line termination The performance gain due to early scan line termination is not for free: Some computation has to be performed for each macro-cell in order to identify the set of scan lines for which early termination is applicable. The reward is a reduced number of ray segments having to be tracked. A reduction of per-pixel-cost is achieved at the expense of increased per-macro-cell-cost. If the number of ray segments can be reduced only slightly or not at all (because it is found that early scan line termination is not applicable), processing of the macro-cell becomes more expensive, instead of cheaper. This is more likely to happen if a macro-cell is projected to only a small region of the screen, resulting in only few ray segments in the first place.

Therefore it is more efficient to confine the application of early scan line termination to macro-cells near the eye-point. Since, due to the octree traversal scheme, macro-cells are processed in order of increasing distance from the eye point, the application of early scan line termination can simply be turned off at some point during the traversal, optimally as soon as the projection size s_p of a processed macro-cell is below a predefined threshold t_{est}.

Macro-cell size Both large and small macro-cells have their advantages and disadvantages: A small macro-cell size leads to a large number of macro-cells, which increases the time needed to traverse the octree and perform computations required for each macro-cell, e.g., rasterization. On the other hand, small macro-cells offer a more compact and more accurate representation of the visualized object and, despite the balancing effect of macro-cell trimming, generally yield fewer local ray segments, especially for smaller objects.

To exploit the advantages of both small and large macro-cells, small macro-cells are used near the eye-point, where the small macro-cell size can effectively reduce the number of ray segments, while in regions far from the eye point the usage of larger macro-cells is more efficient. An arbitrary change of macro-cell size would result in the need to completely rebuild the min-max octree, which is not possible at interactive frame rates. Flexibility in this matter is therefore restricted to treating the sub-volumes represented by nodes at differing levels of the octree as macro-cells.

The following strategy is used: From the moment when a macro-cell with $s_p < t_{macro}$

for a predefined threshold t_{macro} has been encountered, sub-volumes represented by nodes of the second lowest level (instead of the lowest level) of the octree are used as macro-cells, increasing the macro-cell size by factor 2 in each dimension. Using $t_{macro} \leq t_{est}$ makes sense: Early scan line termination is more effective for small macro-cells, because in a small macro-cell the chance that the isosurface exhibits one principal orientation is bigger.

For the results reported in Section 6.2.9, macro-cell size was $4 \times 4 \times 4$ voxels initially and extended to $8 \times 8 \times 8$ voxels for far objects.

Sparse space sampling The fast voxel traversal algorithm applied for tracking a ray segment detects each single cell inside the macro-cell that is intersected by the ray to check for an intersection with the isosurface. This level of accuracy is important for parts of the isosurface that are close to the view point. Otherwise, disturbing visual artifacts would be likely to appear. For distant objects, sampling frequency along the ray can be reduced significantly without creating disturbing image errors. Thus, from the moment when s_p is smaller than a predefined threshold t_{samp} for the first time, the resolution inside the macro-cell is coarsened by combining eight cells to one. This can be done efficiently by manipulating the parameters of the fast voxel traversal algorithm.

Applying these adaptive rendering strategies saves about 20 percent of rendering time on average.

The parameter s_p should be an efficiently computable estimate of the size of the projection of the macro-cell. Since the projection size is indirectly proportional to the distance between the eye point and the center of the macro-cell, simply calculating the (squared) distance would be a feasible measure. To allow for interactive adjustment of the opening angle of the viewing frustum without loss of performance, s_p should be normalized with respect to the opening angle.

STEPS uses a different method: The (squared) length of the diagonal of the smallest bounding box of the projection is calculated and normalized with respect to the (squared) distances between each pair of vertices whose projections define the side lengths of the bounding box. This normalization has to be done for two reasons: The first reason is that due to macro-cell trimming (and the flexible macro-cell sizes introduced in this section) sub-volumes of various sizes are projected. The second reason is that this normalization helps to compensate for the fact that the rotational position of the macro-cell relatively to the viewing frustum also has significant impact on the size of the bounding box.

6.2.9 Results

The rendering algorithm was tested on two different systems: an Intel P4 1900 and an Intel P4 3000 single processor machine. All timings given were measured using a $512 \times 512 \times 201$ data set of a human head. Table 6.2.9 gives timings acquired during a complete virtual endonasal investigation, giving minimum, maximum and average performance of background visualization for a circular image with a diameter of 429 pixels. For the investigation two background objects were loaded from binary masks: the tumor and the internal carotid

	P4 1900		
	min (fps)	max (fps)	av. (fps)
Tumor	6.1	19.4	11.0
Tumor and Vessels	4.5	15.4	8.6
	P4 3000		
Tumor	11.1	32.3	18.6
Tumor and Vessels	8.0	25.1	15.1

Table 6.1: Minimum, maximum and average frame rates (in frames/second) during a virtual endonasal examination

artery. The examination started inside the nose - with the background objects still far away, naturally, the highest frame rates were achieved - and ended just in front of the sellar floor, where background objects cover a large fraction of the screen and the lowest frame rates were measured.

Due to the object-order nature of the algorithm, performance is highly dependent on the sizes of the object projections. The performance drops with increasing number of objects and with decreasing distance of the objects to the view point due to the bigger amount of pixels having to be processed.

6.3   Background Volume Generation

For visualization of the selected background objects, a synthetic volume is generated from all segmentation masks prior to the virtual endoscopic procedure. In this background volume, each background object is represented by an isosurface. Thresholds are either fixed or can be adapted by the user individually for each object. This is because there are two kinds of objects: Those which can be represented as an isosurface in a part of the original volume and those that cannot.

An example for objects of the first group is the ICA. Since a contrast agent is used during data acquisition, the artery is imaged at higher intensity than most of its surroundings. Still, an isosurface in the original CT volume with the correct threshold, representing the vessel boundary, would also include parts of bones and, of course, other large vessels of the head. In the MRA volume, at least other vessels would be included. The best way to get a clean high-quality image of the artery is to manually create a segmentation mask defining an object containing the vessel and its low-intensity surroundings and to copy the values of the voxels contained by the segmented object from the original MRA volume to the background volume. Since cell-based first-hit ray casting allows interactive threshold adjustment, the user can then, during the virtual investigation, complete the segmentation of the artery by selecting the correct threshold. The default value of the threshold is an empirically determined average.
With this method, the vessel does not have to be reconstructed from a binary mask, the appearance of the vessel boundary as recorded by the MRA is maintained.
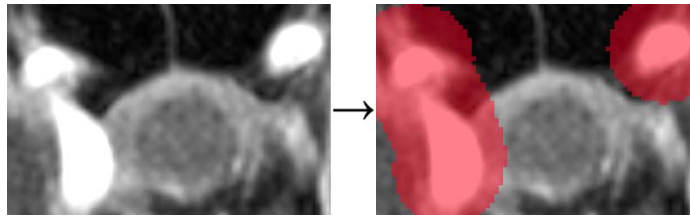
99

Figure 6.13: Manual segmentation of the vessel: The vessel is only roughly segmented (red) in the MR-Angio data set. The underlying intensity values are transferred to the background volume. Segmentation is completed using interactive threshold adjustment during virtual endoscopy.

The second group consists of those segmented objects whose boundaries cannot be represented by isosurfaces in the original data volume. Examples are the tumor, the pituitary gland and the optic nerve. For each of those objects, an artificial isosurface must be constructed in the background volume. This is achieved by applying the technique outlined in Section 5.3.

In order to prevent interferences among overlapping objects, the calculated values are written to the background volume using a maximum-operator: If a voxel obtains more than one value greater than zero (for two or more objects), the largest of these values is finally assigned. Each voxel belongs to one object at most - to the one which determines the voxel value. Rendering neighboring objects with different thresholds can, however, still result in artifacts. To keep these artifacts at an insignificant level, the fixed threshold for reconstructed objects is set within the interval of reasonable thresholds for the vessel. Also, for object reconstruction, a similar data range as encountered in the MR-Angio data is used. Therefore, in relation to the data range, thresholds usually differ only slightly.

6.4   Image Fusion

In each frame during a virtual endoscopy investigation, two images are generated, one depicting the foreground and one showing background objects of interest. The two images are combined using a weighted average operator, yielding the impression of the foreground being semi-transparent. A z-buffer is used to determine those regions of the image where the background is not covered by the foreground. This allows previewing the effect of opening the sellar floor (see section 7.4.5). The z-buffer is also used to enhance depth perception: The distance between foreground and background at each pixel determines foreground opacity (see Figure 6.14). This drastically increases the visual effect of transparency, enhances the association between foreground and background and facilitates navigation.
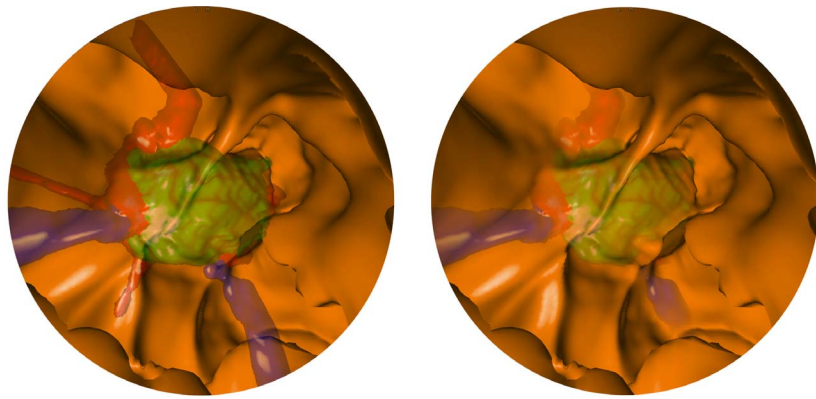
Figure 6.14: Left: simple blending, constant foreground opacity; Right: improved depth perception, foreground opacity z-buffer-dependent

Algorithm 7 Rasterization of macro-cell projection, with accelerated entry point calculation, early scan line termination (EST), ray segment concatenation, sparse sampling and pixel interpolation

1: Algorithm input:
   world- and screen-coordinates of the vertices of the projected macro-cell
2: determine ideal direction for scan line rasterization (here: left to right)
3: generate critical region in which EST may not be performed
4: process the scan lines intersecting the macro-cell projection in the following order: 0,2,1,4,3,6,5,8,7,... {to allow for vertical pixel interpolations and efficient skipping of single scan lines ($skip_{row} == 1$)}
5: **for all** scan lines intersecting the macro-cell projection **do**
6:    $activeFace \leftarrow$ macro-cell face projecting to the left boundary
7:    **for** $i = leftBoundary$ to $rightBoundary$ **do**
8:       **if** pixel i of the scan line is a frontier pixel **then**
9:          $activeFace \leftarrow$ macro-cell face projecting to pixel i
10:       **if** both scan lines above and below the current one are already finished **then**
11:          **if** vertical neighbors of pixel i are similar **then**
12:             interpolate and finalize pixel i
13:       perform Algorithm 8 for pixel i
14:       **if** pixel i is finalized **then**
15:          **if** $pixelsSkipped == skip_{column}$ **then**
16:             {in that case the pixels were skipped due to sparse sampling}
17:             **for** each skipped pixel **do**
18:                $concat \leftarrow concat+1$ {because it was reduced when the pixel was skipped}
19:                perform Algorithm 8
20:          **else**
21:             {$pixelsSkipped == 1$, one pixel was skipped, hoping for interpolation}
22:             **if** horizontal neighbors of skipped pixel are similar **then**
23:                interpolate and finalize skipped pixel
24:             **else**
25:                for the skipped pixel: $concat \leftarrow concat + 1$ {because it was reduced when the pixel was skipped}
26:                perform Algorithm 8 for skipped pixel
27:          skip 1 pixel, if the skipped pixel is a frontier pixel, adjust $activeFace$
28:          $pixelsSkipped \leftarrow 1$
29:          $concat \leftarrow concat - 1$ for skipped pixel
30:       **else**
31:          **if** pixel i is not inside the critical region (for EST) **then**
32:             $concat \leftarrow concat - 1$ for the remaining pixels of this scan line
33:             terminate scan line
34:          **else**
35:             skip $skip_{column}$ pixels, if any of the skipped pixels is a frontier pixel, adjust $activeFace$, $pixelsSkipped \leftarrow skip_{column}$
36:    A complete scan line has been processed. If the two neighboring scan lines of the scan line that is next in the queue have already been processed, and there are no finalized pixels in either of them, the scan line next in the queue can be skipped.

Algorithm 8 Initialization and launching of a local ray segment from pixel i through macro-cell M, if ray segment concatenation is used

---

1: Algorithm input:
   coordinates of pixel i
   M: currently processed macro-cell
   $activeFace$: the face of M projecting to pixel i
2: **if** pixel i has not yet been finalized **then**
3:     **if** $concat_i > 0$ **then**
4:         $concat_i \leftarrow concat_i - 1$
5:     **else**
6:         calculate ray direction for pixel i
7:         intersect the ray with $activeFace$ to find the entry point p_{entry}
8:         track local ray segment inside macro-cell M, starting at p_{entry}. Stop tracking the ray as soon as an intersection with the iso-surface has been found, or a macro-cell which does not contain a part of the iso-surface is entered
9:         **if** an intersection with the iso-surface has been found **then**
10:             finalize pixel i
11:         **else**
12:             set $concat_i \leftarrow$ number of trimmed macro-cells traversed by the ray, apart from the macro-cell that is currently processed

---

Chapter 7

STEPS - The User Interface

This chapter described the user interface of STEPS. It introduces navigation aids, helping the user to find the correct route, explains the different modes of camera control and how transsphenoidal pituitary surgery is simulated by STEPS.

7.1   Display

The screen is separated into four parts (see Figure 7.1): the 3D-view displaying the simulated endoscopic image and a multi-planar reconstruction (MPR) consisting of three *section views* displaying raw CT data and selected background objects on three mutually orthogonal cross-sections of the volume. The current eye point is the intersection point of the three cross-sections. The cutting planes defining the cross-sections can be arbitrarily rotated around the eye point by the user. The section views are linked to each other - the intersection between any pair of cross-sections is displayed in both section views.

7.2   Navigation Aids

Especially for an inexperienced surgeon, it is often hard to find the correct route through the nasal cavities to the sellar floor. Therefore navigation aids are needed. In physical endoscopy, navigation aids can be provided by a navigation system which tracks the position of the endoscope camera or a pointing device and displays it in a MPR. In virtual endoscopy, this can be achieved more easily and more accurately, because the exact camera position is always known. The MPR displayed by STEPS provides a feasible means of increasing the understanding of the relationship between original data and reconstructed 3D images.

The user can link any of the section views to the 3D view as illustrated in Figure 7.1. If a section view is linked, the intersection of its cutting plane with the viewing frustum is displayed in both the 3D view and the section view. The boundaries of the intersection are marked in different colors (e.g., blue and green for the red intersection) in both images, adding to the understanding of the interconnection between 3D view and section view. This tremendously improves navigation, since structures seen in the 3D view are mapped to their
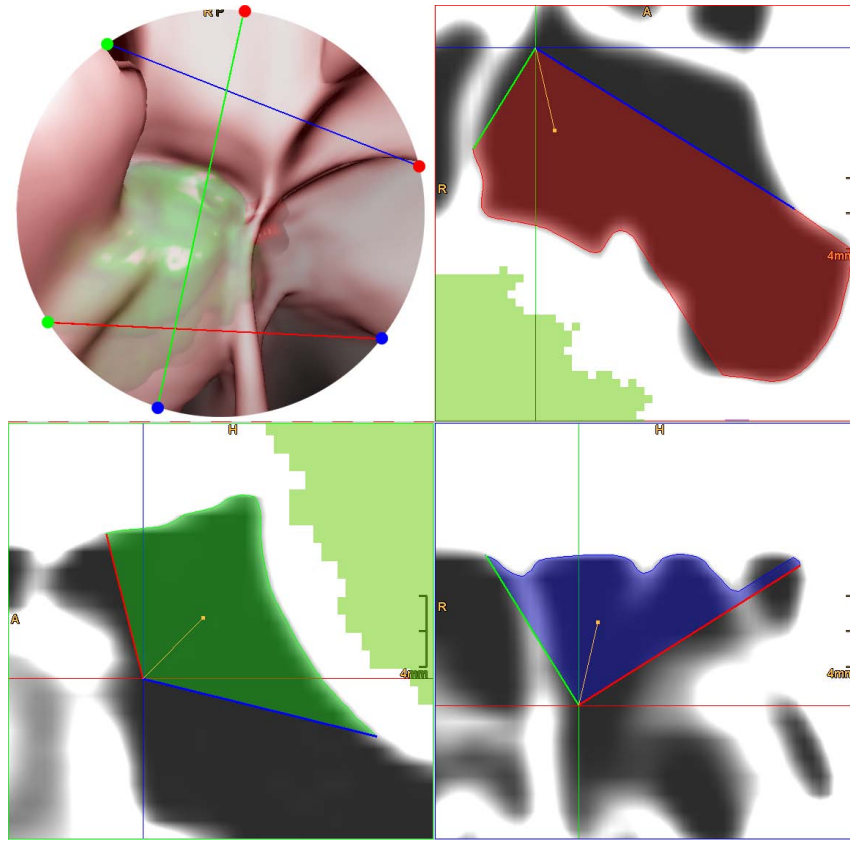
Figure 7.1: The 3D view (top left) and the three section views are linked to each other. Note the septum on the left hand side of the 3D view, rendered incompletely due to a too high threshold, as confirmed by the blue section view (bottom right).

locations in the CT data in a very intuitive way. This technique also serves as an instrument to find a suitable threshold for the foreground isosurface.

Additionally to the linked views, navigation is enhanced by communicating the principal orientations of the row and column vectors of the viewing plane. Figure 7.2 shows an example. Near the left boundary of the 3D image, three letters are displayed. The well-noticeable *H (head)* signals that the vector V_{left} from one pixel in the image to its left neighbor in 3D is almost parallel to the patient's longitudinal axis and pointing towards the patient's head. The opposite direction would be denoted by an *F (feet)*. Letters displayed in darker color indicate that the vector points only slightly into the according direction. Other directions are *R (right), L (left), A (anterior - towards the front)* and *P (posterior - towards the back)*. Analogously the letters on the top of the image indicate the orientation of the vector pointing from the bottom to the top of the image in 3D.

The remaining navigation aids are designed to guide the user to the target, the reconstructed sphenoid sinus. If a target point has been specified by the user during simulation, specifying the location at which the sellar floor should be opened (see Section 7.4.1), the
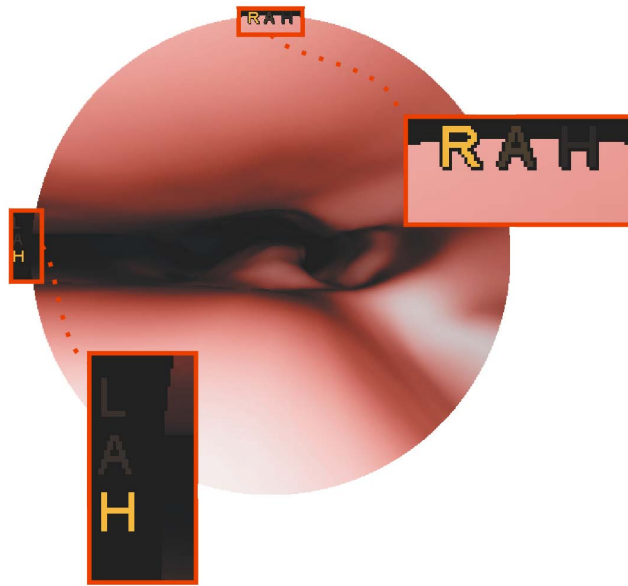
Figure 7.2: The orientations of the row and column vectors of the viewing plane are indicated by small letters on top and on the left side of the 3D image

direction to the target point is indicated by a green cross (see Figure 7.3, left image). Otherwise, or additionally, the image fusion strategy enhancing depth perception (see Section 6.4) can be turned off (or the minimum transparency of the foreground can be set to a value greater than zero, respectively), making background objects visible, regardless of how far away they are. The background objects clearly mark the direction to the target (see Figure 7.3, right image).

7.3 Control

As pointed out in the introduction, STEPS can be used for various purposes - as a training device, for preoperative planning and for intraoperative support. Therefore the application provides two different modes of operation. The modes differ in the way user interaction is implemented.
The first mode of operation is the *free flight mode*. In this mode, the user is allowed to move freely, with the single restriction that the view point is not allowed to cross an isosurface. This mode is used for preoperative planning and intraoperatively - to get familiar with a patient's anatomy, view landmarks from different viewing angles, and plan where to open the sellar floor. The second mode of operation provides a simulation of the procedure. It is described in section 7.4.
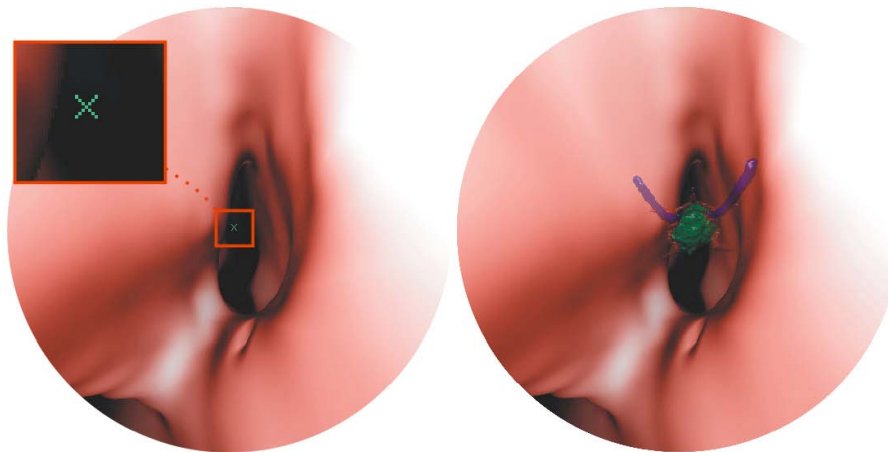
Figure 7.3: The direction to the target can be indicated by a green mark (left image) or the background objects (right image)

7.4 Simulation

The user can decide to either move freely inside the patient's head or to simulate the endonasal transsphenoidal procedure. With STEPS, the process of advancing the endoscope through the narrow nasal cavities and finding the correct way to the sphenoid sinus can be simulated as well as the process of widening the sphenoid ostium, the removal of sphenoid septa and one of the most crucial parts of the actual surgical operation, the process of opening the sellar floor.

Simulation is used mainly for training purposes but can also provide information about accessibility and the risks and obstacles to expect, which makes it useful also for preoperative planning. For the simulation, it is important to model the degrees of freedom and constraints experienced when working with a rigid endoscope as accurately as possible. The user should be confronted with similar challenges as in the real procedure and guided by haptic feedback.

In STEPS, the virtual endoscope is maneuvered using a force-feedback joystick (WingMan Force 3D, Logitech, http://www.logitech.com, see Figure 7.4).

7.4.1 Camera Control

The simulation starts with the user selecting a *start point* p_{start} and a *target point* p_{target} in the CT volume. The start point defines the position where the endoscope is entered into the patient and is usually located inside one of the patient's nostrils. The target is on the sellar floor. After selection of the two points, the eye point p_{eye} is automatically moved to p_{start} and the viewing vector points towards p_{target}. The user can move forwards (into the viewing direction) and backwards using joystick buttons and rotate the virtual endoscope by moving the joystick handle.

The endoscope is modeled as the line segment connecting p_{start} and p_{eye}. It can be rotated only around p_{start} (see Figure 7.6).
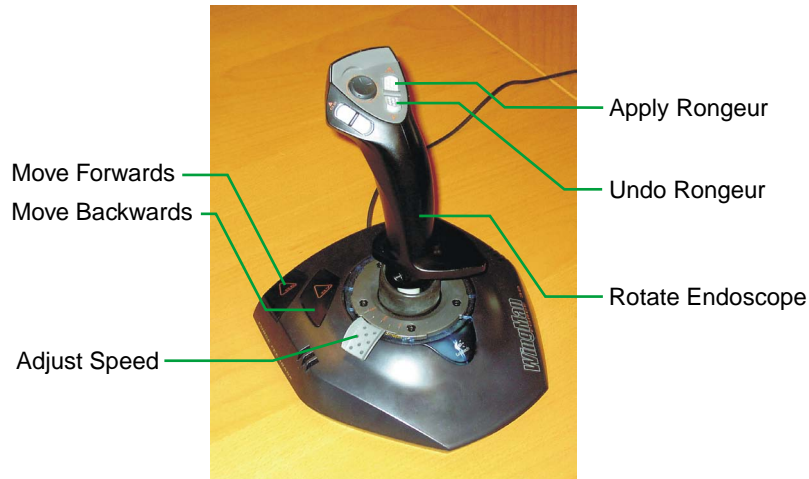
Figure 7.4: The joystick and the mapping of operations

7.4.2 Collision Detection and Implicit Deformation

On its way to the sellar floor the endoscope traverses a number of cavities in the skull. These cavities are separated from each other by bony or cartilaginous walls, so-called *septa*. Bones form barriers for the movement of the endoscope. Soft (i.e., deformable) tissue (e.g., mucosa and the cartilaginous part of the nasal septum) also poses resistance to movement, but can be deformed to some extent. In fact, soft tissue must be deformed to gain access to the sphenoid ostium.

STEPS currently simulates soft tissue deformation in an implicit way. For performance reasons, the deformation is not modeled geometrically but manifests itself only in relieved constraints of movement and in force-feedback. Volume adjustments resulting in optically or even physically correct deformation of the isosurface are computationally expensive and not straightforward. Implicit deformation as performed by STEPS is less costly and yields equivalent results as long as it does not affect the eye point.
STEPS uses two thresholds to define the data range of soft tissue:

- a soft threshold t_{soft}, equal to the foreground iso-value, and

- a firm threshold t_{firm} representing the boundary between soft tissue and bony structures,

with $-1024HU < t_{soft} < t_{firm} < 3071HU$, HU being short for *Hounsfield Units*. Apart from the eye-point, the virtual endoscope may be pushed into soft tissue, i.e., implicitly deform it. No part of the virtual endoscope is allowed to be moved to intensity greater than t_{firm}. Because p_{eye} may not cross the foreground isosurface, tissue deformation can only be performed via endoscope rotations and not by advancing the endoscope. In the vicinity of p_{eye}, geometric deformation would therefore be of significant advantage to prevent artificial limitations of endoscope movement and to increase the overall realism of the simulation. A
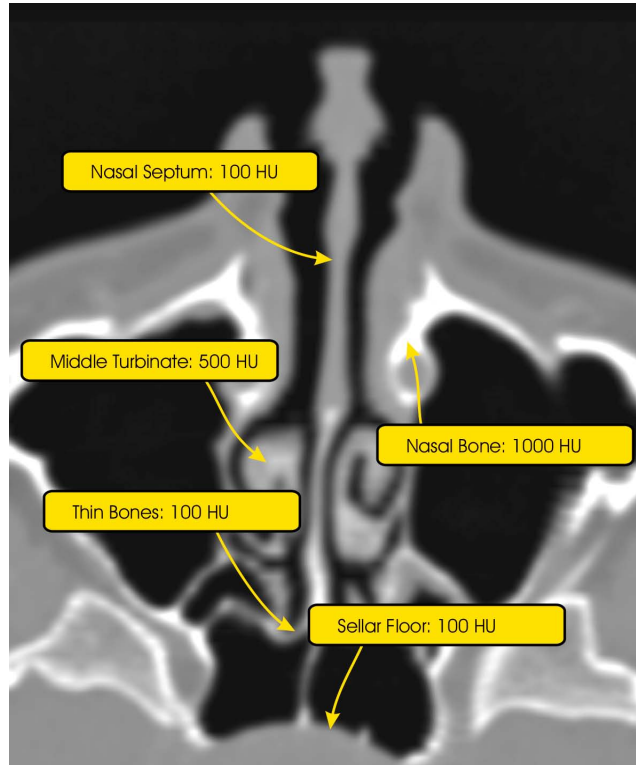
Figure 7.5: The CT values of bony structures vary significantly: Bones encasing the sphenoid sinus are measured at the same or even a smaller data level than, for example, the outer, cartilaginous part of the nasal septum.

suitable model is currently being developed. Preliminary results are given in Section 7.4.9.

Because bones encasing the sphenoid sinus are very thin, they are, due to partial volume effects, captured by the CT at a lower intensity level than more massive bones (see Figure 7.5). The intensity level of these non-deformable barriers is in fact similar to the intensity level of larger deformable structures, as experienced near p_{start}, like the cartilaginous nasal septum. Therefore, the deformability of virtual structures cannot be made dependent of CT-intensity values alone, but also spatial information must be taken into account. In other words, the value t_{firm} must change along the path of the endoscope. The same holds, to a lesser extent, also for the threshold t_{soft}. To achieve a behavior similar to reality, the value t_{firm} is defined as a function of space, the threshold $t_{firm}(V)$ used at voxel V depending on the distance d_{sv} between V and p_{start}. Currently the following step-like function is used, values are in HU:

$$t_{firm}(V) = max(f(d_{sv}), t_{soft} + 1) \tag{7.1}$$

with

$$f(d_{sv}) = \begin{cases} 450 & d_{sv} \leq \frac{8}{10} d_{st} \\ -450 & d_{sv} \geq \frac{9}{10} d_{st} \\ -\frac{1}{d_{st}} 9000 \, d_{sv} + 7650 & \text{otherwise} \end{cases} \tag{7.2}$$

where d_{st} denotes the distance between p_{start} and p_{target}. The formula is motivated as follows: Soft tissue that must be deformed is usually encountered within the first 80 percent of the path of the endoscope. The transition through the ostium takes place after about 90 percent of the traveling distance. The formula might have to be adapted for patients for whom those parameters differ significantly.

So when p_{eye} is inside the sphenoid sinus, the tip of the endoscope experiences a small threshold t_{firm} compared to the part of the endoscope that is inside the nose. This means that during the sphenoid phase parts of the endoscope close to p_{start} can still deform tissue, while movement of the tip of the endoscope is bounded by almost any visible structures, which is realistic, because the bones encasing the sphenoid sinus are usually only covered by a very thin layer of mucosa. As a result, freedom of movement during the sphenoid phase is confined by the size of the (enlarged) ostium, as is the case also in reality.

The threshold used for foreground visualization, $t_f = t_{soft}$, is usually selected such that soft tissue, as seen during the nasal phase, and thin bones, as seen especially during the sphenoid phase, are depicted. The intensity levels are similar, but there may not be an iso-value that is equally suitable for both. For this reason, the user can select to make t_f, and therefore t_{soft}, a linear function of the distance between p_{start} and p_{eye}. Thus t_{soft} might slightly change in time, with p_{eye} moving.

In the simulation, actions which would cause the eye point to move across the visualized foreground-isosurface are not allowed. To prevent the eye-point from jumping past thin walls, the path from the previous to the current eye point is sampled after each movement. If an intersection with the foreground isosurface defined by t_{soft} is found, the endoscope is returned to its previous position. Sampling is performed by casting a viewing ray, such that any visible structures are encountered with a very high probability. To prevent further forbidden collisions, this process is repeated for a number of points evenly distributed along the endoscope, with the only difference that for points other than the eye-point the threshold t_{firm} is used instead of t_{soft}.

Collision detection is only performed for the center axis of the virtual endoscope and not for its complete diameter. This is compensation for the fact that the surgeon in reality has an increased amount of freedom by performing slight intuitive movements.

7.4.3 Force-Feedback

If a part of the virtual endoscope is inside soft tissue, force-feedback is exerted with the goal to press the endoscope back into regions of lower image intensity. The direction and intensity of force-feedback are calculated using the algorithm illustrated in Figure 7.6:

The sorted set of intersection points of the virtual endoscope with the foreground iso-surface is a sequence of, alternately, *entry points* (of the endoscope into tissue denser than t_{soft}) and *exit points* (out of tissue denser than t_{soft}). An *intersection segment* is a segment of the virtual endoscope that is associated with tension, because tissue has been deformed. An intersection segment is always confined by an entry point and an exit point, but may span more than two isosurface intersections for the following reason: Inside soft tissue there
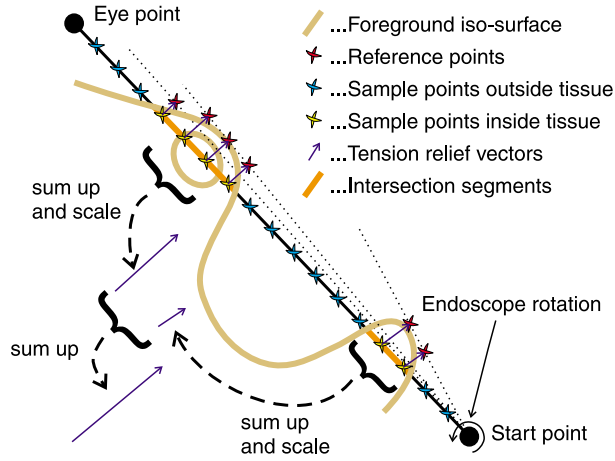
Figure 7.6: Calculation of the tension relief vector after endoscope rotation

may be, due to noise, small connected sets of voxels whose values are smaller than t_{soft}. Intersections with these regions do not break an intersection segment apart. For example, the endoscope depicted in Figure 7.6 intersects the isosurface six times (three entry points and three exit points). Still, there are only two intersection segments.

The direction into which the endoscope should be rotated in order to reduce the tension caused by the ith intersection segment S_i most effectively, is defined by the *tension relief vector* $v_{tension}^i$.

Tension relief vectors are calculated as follows: The CT data is sampled at evenly-spaced points along the virtual endoscope, from p_{start} to p_{eye}. The inter-sample distance d remains constant throughout the simulation. The λth sample point p_{sample}^λ is therefore always located at distance $\lambda \cdot d$ from p_{start}. For each λ, a reference point p_{ref}^λ is stored, indicating the last position of a λth sample point that was outside soft tissue. The ith tension relief vector is calculated as follows:

$$v_{tension}^i = c^i \frac{v}{\mid v \mid} \tag{7.3}$$

with

$$v = \sum_{\{\lambda \mid p_{sample}^\lambda \in S^i\}} (p_{ref}^\lambda - p_{sample}^\lambda) \tag{7.4}$$

and the coefficient c^i being a function directly proportional to the largest intensity value in the (slightly low-pass filtered, to reduce the impact of noise) sequence of intensity samples inside S_i and directly proportional to the distance between the center point of S_i and p_{start} (simulating torque). Finally the sum of all tension relief vectors is transformed to screen space and directly passed on to the force-feedback-joystick. The intensity of force-feedback is determined by the Euclidean norm of the resulting vector.

To further increase the quality of simulation, it is planned to apply rotations of the endoscope also to reference points with the constraint that they must not cross the foreground isosurface.
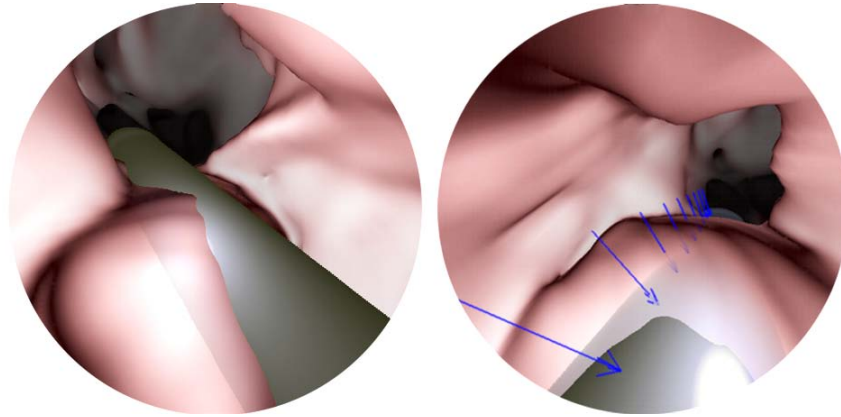
Figure 7.7: During suspended simulation, the endoscope is added to the scene. In case tissue was deformed, arrows point into the direction of the deformation (right).

7.4.4 Suspended Simulation

During simulation, the user's range of view is very limited. It can, as in real endoscopy, be enhanced by switching to an angled endoscope (see Section 7.4.6). This, however, is usually not sufficient to acquire a complete picture of the current state of the simulation, which would be useful especially for training purposes.
STEPS provides the possibility to suspend the simulation. In that case, the simulation is frozen and the application switches to a temporary free-flight mode. During suspension of the simulation, the endoscope is added to the rendered scene as an additional background object (see Figure 7.7). The user can view the current situation and assess the state of the simulation from arbitrary view points. If movement of the endoscope is blocked or endoscope rotations are prevented, suspending the simulation helps to find out the reasons.

Since currently there is no geometrical soft tissue deformation, the virtual endoscope, pushed far enough into soft tissue might make the impression of having sunk into the tissue (see Figure 7.7). This can be quite confusing and it is sometimes hard to see from which direction soft tissue has been deformed.
To give a more intuitive picture, the direction of deformation is depicted using arrows. For this, the information determining the application of force-feedback is reused: For each intersection segment a tension relief vector was calculated. This vector points, approximately, into the direction where the endoscope came from when being pushed into soft tissue. Each arrow points, in direction of the according inverted tension relief vector, to the surface of the endoscope. Arrow length corresponds to the length of the tension relief vector and is therefore proportional to the amount of force-feedback caused by the corresponding intersection segment (see Figure 7.7, right).
As soon as the simulation is continued, the endoscope is removed by restoring the original voxel values from the hard disk.

7.4.5 Simulation of surgical instruments

Surgical instruments are needed on several occasions during the procedure: The sphenoid ostium must be widened using a bone punch in order to gain sufficient access to the sphenoid sinus. Inside the sphenoid sinus, sphenoid septa might have to be removed to uncover the complete sphenoid anatomy, again using the bone punch. Finally, the sellar floor must be opened. Since the bone punch cannot be used to create, but only to enlarge an opening, a small hole is created using a drill or chisel, then the hole is widened using the rongeur.

In the simulation there is no need to use separate instruments, i.e., both drill and bone punch. Instead, STEPS simulates a bone punch with a built-in drill, which can be used also to create an opening.

The Virtual Bone Punch

The user can select the radius r and the depth d of the bone punch to use. The virtual rongeur is maneuvered using the mouse. In previous versions of the system, the punch was activated using a joystick button [NWF$^+$04]. Control of the rongeur was, however, completely delegated to the mouse, since, also in reality, the endoscope and the rongeur are two disjoint instruments. When the mouse cursor is moved over the endoscopic image, a preview is shown underneath the cursor, indicating the expected impact of the punch (see Figures 7.9 and 7.10). A mouse-click leads to activation of the rongeur at the *activation point $p_{activation}$*, i.e., the position currently rendered below the mouse cursor.

The bone punch is, logically, a rigid instrument, like the endoscope. Therefore it can be maneuvered with the same constraints as the endoscope. During the simulation, the punch preview is only shown and the punch can only be activated if the mouse cursor points to a position on the surface which can be reached from p_{start}. For testing the reachability of the rongeur, the same rules are applied as for testing the reachability for the simulation of endoscope movement (see Section 4.2.4).

Preview Generation

The preview of punch impact is generated using the following algorithm: The punch vector v_p is identified from the coordinates of the mouse cursor. It points from p_{start} (during simulation) or from p_{eye} (during free flight) to $p_{activation}$. If the simulation mode is active, the reachability is checked by testing, whether this vector intersects the isosurface defined by threshold t_{firm}.

If the result of the test is negative (i.e., there is no intersection), *n ray source points* are established. These points are evenly distributed on a *base circle* which is located on a plane perpendicular to v_p and initially centered on $p_{activation}$. The radius of the base circle is the user-specified radius r. The base circle is then projected orthogonally, in direction of $\pm v_p$, to the isosurface. The projection is displayed as a polygon in black color (see Figure 7.9). This polygon marks the *optical region of impact (OROI)*. Next, the plane containing the base circle is shifted into direction of its normal ($-v_p$) until the base circle does not intersect the isosurface any more. At the end of this procedure, it marks the beginning of the *physical region of impact (PROI)* of the punch. The PROI comprises all parts of the surface
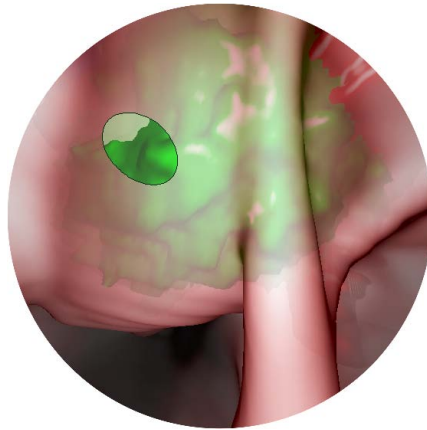
Figure 7.8: The punch preview shows whether a part of a background object is within reach. Here, the dark green color indicates that the tumor is within the region of impact of the rongeur.

that are rendered inside the OROI and whose distance d_s from p_{start} satisfies the condition $d_{sb} \leq d_s \leq d_{sb} + d$, with d_{sb} being the distance between p_{start} and the shifted base plane. In the punch preview, pixels depicting a part of the PROI are highlighted (light green color). If a part of a background object is inside the PROI, the preview shows the color of the object (see second image, Figure 7.8).

Punch Activation

The user can activate the punch with a mouse click. The punch cuts a cylindrical piece of the specified radius and depth out of foreground tissue. This is done by changing voxel values in the CT volume accordingly - the impact can therefore also be observed in the MPR. Information about each activation of the punch is stored to disk. Thus any volume manipulations can be undone, allowing, for instance, a restart of the simulation.

Figure 7.9 shows the process of virtually widening the sphenoid ostium. Figure 7.10 depicts the simulation of the process of opening the sellar floor.

7.4.6 Angled Endoscopes

In reality, the optical system of the endoscope can be adapted satisfying special needs of the surgeon. Special objectives attached to the endoscope cause the principal viewing vector to deviate from the longitudinal axis of the endoscope by a certain angle. The angle can be selected to be 12, 30 or 45 degrees for forward oblique view, 70 or 90 degrees for lateral view and 120 degrees for retrospective view.
In case of an angled objective, rotation of the endoscope around its longitudinal axis does not result in simple rotation of the endoscopic image around its center with constant view direction, but also the principal view direction is rotated around the endoscope axis. This
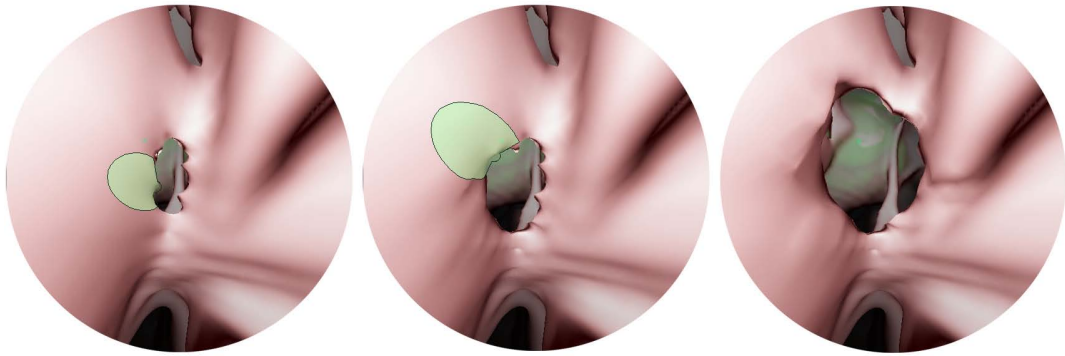
Figure 7.9: Using the virtual bone punch, the sphenoid ostium is enlarged to gain sufficient access to the sphenoid sinus
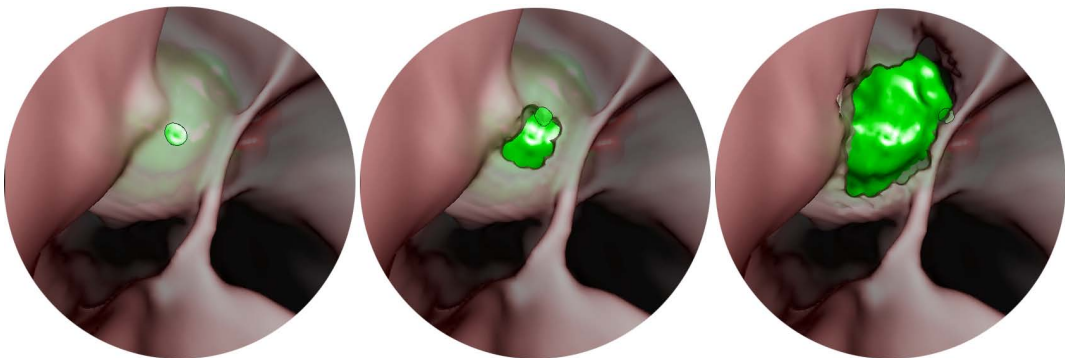
Figure 7.10: The virtual bone punch is used to open the sellar floor and uncover the tumor

tremendously enlarges the surgeon's field of view, but at the same time makes handling of the endoscope far more complicated. Laterally located structures appear to be exactly ahead of the camera. This requires a learning process, often hardly accepted, especially by older surgeons who are used to working with operating microscopes. The larger the angular deviation, the less intuitive is the visual response to movements of the endoscope. Therefore, usually the following strategy is pursued: In the beginning, a non-angled endoscope is inserted to identify the path to the point of application of surgical activity. As soon as instruments have been added and the path is laid, the endoscope can be retrieved, the objective replaced and the endoscope inserted again, next to the surgical instruments.

STEPS supports simulation of angled endoscopes and therefore helps the user get used to the difficulties arising with maneuvering an endoscope with an angled viewing axis. The angle can be selected at any time. The strategy used in the virtual case is as follows: use an angle of 0 to get to the position of interest, then change the angle and start inspecting lateral structures by rotating the endoscope around its longitudinal axis (twisting the joystick handle). The advantage of the virtual endoscope is that it does not need to be retrieved out of the patient to change its optical parameters.

7.4.7 Realistic Reaction to Endoscope Rotation

Rotating the endoscope around its longitudinal axis leads to further endoscope movements having different visual consequences. The reason is that the *up*-vector of the endoscope, which defines, what part of the rendered image will be displayed at the top of the screen, is changed. If, for example, the endoscope is twisted by 90 degrees, the up-vector ends up orthogonal to its initial direction. If now the tip of the endoscope is moved horizontally (e.g., because the endoscope is rotated horizontally by the surgeon), the image appears to move in vertical direction. This can make maneuvering quite unintuitive and difficult.

If an angled endoscope is used, the confusion is even increased.

In reality, the endoscope is inserted in correct orientation, i.e., initially, endoscope movements lead to intuitive image changes. Whenever the endoscope is twisted, it is usually returned to the correct orientation shortly afterwards. To be able to simulate this behavior, STEPS offers the possibility to select the correct orientation, i.e. the orientation in which the endoscope is entered into the patient. That is, when the according button is pressed, the current up-vector of the virtual endoscope is taken as a reference for subsequent user interaction and returning to this reference up vector yields intuitive reaction to endoscope rotations. Of course, all changes of the viewing vector must be applied also to this reference up-vector.

7.4.8 Lens Distortion

Endoscope lenses usually exhibit slight barrel distortion. STEPS simulates this by accordingly adjusting viewing ray directions. The amount of barrel distortion can be adjusted by the user. Exact calibration according to real endoscopic images is not yet provided.

The foreground image is distorted by accordingly altering ray directions. The background image must be distorted in a post-processing step, since cell-based first-hit ray casting relies on the linearity of projections. The disadvantage of postprocessing is that it is achieved via interpolation in the original image and therefore effectively reduces image resolution. Figure 7.11 illustrates the non-linearity of barrel distortion. The red, green and blue curves depict the intersections of the scene with the section views of the MPR and therefore, originally, linear.

7.4.9 Geometric Tissue Deformation

In order to increase the realism of the simulation, it is necessary to model realistic reaction of tissue to interaction with surgical instruments. Soft tissue deformation is currently being added to STEPS. This section provides preliminary results.

Many algorithms for implementation of realistic tissue deformation have been proposed to date. Most of them, however, work on some indirect representation of the rendered scene (e.g., triangle meshes or tetrahedral meshes) rather than directly on the volume. Since the rendering algorithms applied by STEPS work directly on voxel values, ensuring the maximum amount of flexibility and hardware independence, the goal is to also implement tissue deformation by directly adjusting voxel values. This turned out a difficult and demanding
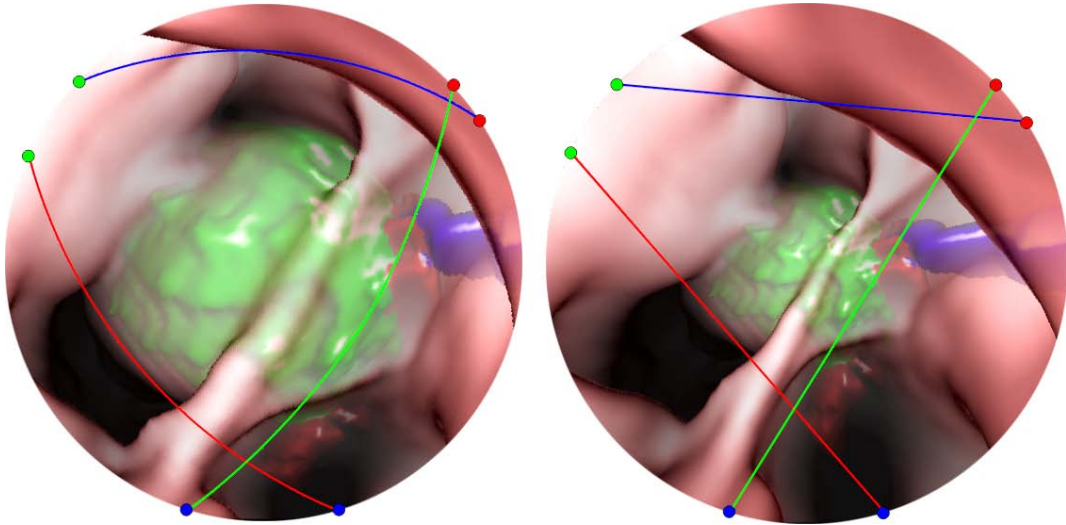
Figure 7.11: Barrel Distortion to simulate real endoscopic cameras; Left: strong (exaggerated) barrel distortion; Right: undistorted; The red, green and blue curves illustrate the non-linearity of barrel distortion.

task. Still, first results are encouraging.

The technique used is based on the Chainmail algorithm introduced by Gibson [Gib97]. In the original algorithm, the deformable tissue is represented as a three-dimensional lattice of interconnected (chain) elements. Each element is connected to its top, bottom, left, right, front and back neighbors. Constraints are defined to ensure a minimum and maximum distance between a pair of linked neighbors, both parallel and orthogonal to the link. Therefore, if one element is moved, its linked neighbors will, to some extent, adopt the movement.

In the adapted algorithm [Drä05] used by STEPS, each voxel, regardless of whether its value is smaller or greater than the current foreground iso-value, is associated with a chain element centered at the voxel coordinates. As soon as the eye point of the endoscope crosses the isosurface, the eight voxels enclosing the cell inside which the endoscope intersects the isosurface, are attempted to be moved such that the endoscope does not intersect the isosurface anymore. The chain mail neighborhood constraints are applied to determine the global effects of this deformation. The constraints are influenced by voxel values: the higher voxel values are, the more rigidly the represented tissue behaves. If the global deformation violates a constraint, the deformation is invalid and must be, to some extent, reversed or completely canceled.
After the deformation of the chain structure, each new voxel value is calculated through interpolation between the voxel values associated with the surrounding chain elements.

Figure 7.12 shows an example deformation.

117
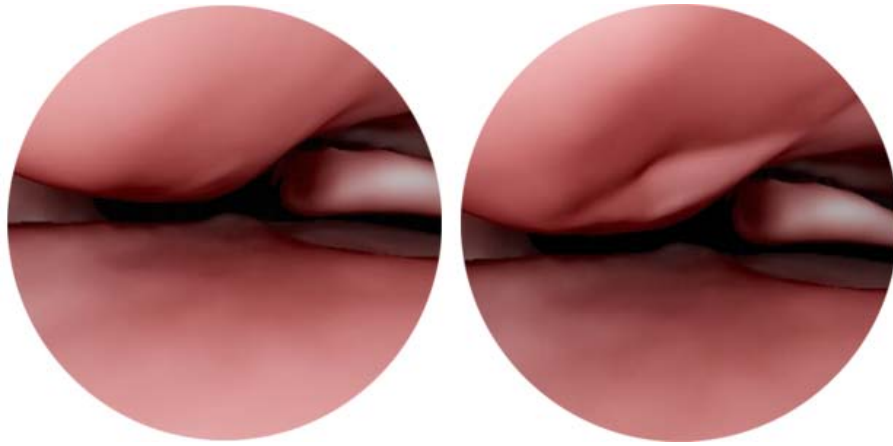
Figure 7.12: Geometric tissue deformation: The endoscope thrusts soft tissue aside; Left: original; Right: deformed

Chapter 8

STEPS - Discussion

STEPS has been developed in close cooperation with neurosurgeons and tested within the scope of a medical research project [For03, WFD$^+$04]. This chapter deals with problems and questions brought up during development and evaluation and assesses the value of the application for training and preoperative planning.

8.1 Time Consumption

Preprocessing of objects is time-consuming. Registration usually takes no more than 3 minutes, even if manual preprocessing is required. Segmentation, however, is still a tedious task, because usually much manual work must be done. Segmentation of the ICAs is generally performed quickly, since the boundaries do not have to be specified exactly. Segmentation of the tumor, the gland and the optic nerves, however, takes more effort. All in all, the time needed for segmentation amounts to approximately 30 minutes per patient.

The virtual endoscopy is then usually set up within 3 minutes, including reconstruction of all objects. The time needed for preoperative investigation depends heavily on the surgeon's experience and intention as well as state and anatomy of the patient, but can be estimated roughly as 30 minutes.

STEPS attempts to keep above a minimum frame rate during interaction by rendering images of adaptive size and scaling them to fit the screen. The minimum frame rate is adjusted by the user. Typically, a circular image with a diameter of 500 pixels can be rendered at 10 fps on a P4-3000 MHz.

8.2 Visualization

Feedback from the medical doctors concerning the visualization was predominantly positive. The surgeons observed a good correlation of the intraoperatively perceived and the virtually reconstructed anatomy (see Figure 8.1) [WFD$^+$04]. Rendering quality and speed is considered sufficient.
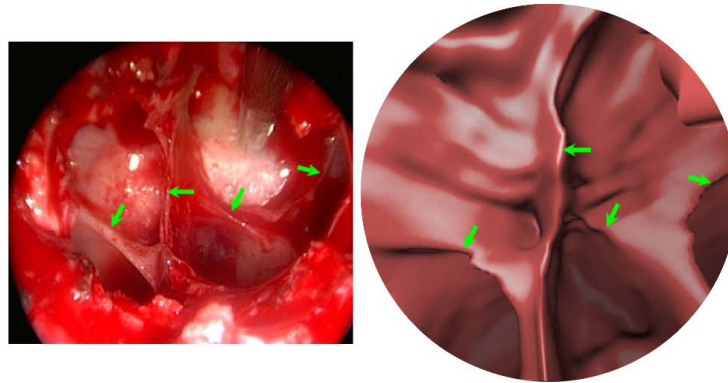
Figure 8.1: Comparison of a real and a virtual endoscopic image of a multi-septated sinus. For better orientation, the septa are marked by green arrows. The images are courtesy of the Institute of Neurosurgery, Medical University Vienna.

8.2.1 Background Visualization

There is an ongoing discussion among people associated with this project, whether background objects reconstructed from binary segmentation masks should be smoothed, or not (see Section 5.3).

Smoothing alters object appearance as communicated by the segmentation mask created by the physician. Smoothing is always associated with a low-pass filtering process and therefore reduces the amount of information given. This can, in a medical application, lead to severe problems. On the other hand, smoothing makes objects look more realistic. This increases the acceptance among physicians and their patients.

Therefore, if the virtual endoscopy system is used for training purposes, in our opinion, background objects can and should be smoothed to some extent. In the case of preoperative planning there must be the possibility to add unsmoothed objects.

If the physician chooses to smooth objects, the additional error introduced by smoothing must be constrained, and the maximum error must be communicated to the physician. Also, physicians must (and usually do) keep in mind that the image provided by virtual endoscopy does not depict reality, but a snap-shot reconstructed from discretized scalar data.

In the worst case, the errors induced by scalarization, discretization, recording, segmentation, (possibly) smoothing and reconstruction add up. The result is an approximation which must not be mistaken as a completely accurate representation. Therefore, an assumption like the one that cutting the sellar floor at a certain position is safe, because in the virtual endoscopy a vessel is missed by a few pixels is not at all justifiable, regardless of whether objects were smoothed.

In STEPS, the user can select, for each reconstructed background object, whether it should be smoothed, and adjust the amount of smoothing applied.

The way some objects (e.g., the ICAs, see Section 6.3) are reconstructed in STEPS, by passing the original MRA data almost through the complete rendering pipeline and handing the crucial decision, the definition of the boundaries, i.e., the adjustment of the iso-value, to the physician, helps the physician correctly interpret the visual result, its relation to the data and its expressiveness. Also these objects do not have to be reconstructed from a binary

mask, the appearance of the object boundaries as recorded by the MRA is maintained. The question, whether smoothing should be applied, does not arise for these objects.

The segmentation result can always be cross-checked in the MPR in which the voxels belonging to a visible background object are highlighted (see Figure 7.1). Therefore this way of segmenting is not any more error-prone than performing the segmentation solely upon original MRA data.

8.2.2 Foreground Visualization

A well-known problem of iso-surfacing based on intensity values as obtained from a CT is that isosurfaces generally do not entirely correspond to boundaries as observed in real endoscopy. Optimal threshold values differ between structures, even between different structures seen in the same image. Due to partial volume effects, a small variation of the threshold can have a quite drastic effect on image appearance. A slightly wrong threshold can result in a certain structure appearing quite differently from its real shape as seen during the real endoscopic procedure.

Adding an additional dimension to user interaction by providing interactive threshold adjustment significantly reduces the impact of this problem. The increased amount of control over the visual outcome handed to the user helps them get a feeling for the relation between raw data and resulting images and makes them aware of the fact that what they see is not equal to reality, but an approximation. On the other hand it is worth noting that it takes some time and training for medical users to get used to the provided amount of control. However, according to our experience, after this learning phase, they are convinced of the advantages, as pointed out above, and they also have a better appreciation and understanding of the strategies and processes underlying the visual feedback.

8.3 Application as a Training Device

STEPS was recognized to be beneficial as a training tool. The traditional way to train transsphenoidal pituitary surgery is cadaver training (see Figure 8.2). Advantages of virtual endoscopy over cadaver training include lower cost and effort, unbounded reproducibility, an improved learning curve by utilizing navigation and perception aids which can to some extent replace an instructor, and the possibility to suspend the training simulation and view the situation from perspectives unreachable for a real endoscope. These advantages make virtual endoscopy a valuable addition to cadaver training.

Still, training as currently offered by STEPS suffers from some drawbacks: A force-feedback-joystick is a cheap but not ideal solution to the problem of modeling haptic feedback as experienced in real life. Near-exact simulation would require using some highly sophisticated haptic feedback device which comes at very high cost. However, although force-feedback as applied by STEPS is not completely realistic, it manages to give the user a very intuitive notion of soft tissue deformation, which substantially increases the value of the simulation. Currently only a part of the simulation is enhanced using force-feedback. There is no force-feedback triggered by the virtual bone punch and by tissue encounters

Figure 8.2: Neurosurgeon performing a transsphenoidal pituitary procedure on a cadaver. The image is courtesy of the Institute of Neurosurgery, Medical University Vienna.

during advancement of the endoscope.

The lack of geometric tissue deformation reduces the maneuverability of the virtual endo-scope. It, for instance, sometimes prevents the virtual endoscope from being retreated. The reason is that if the endoscope intersects the visible isosurface (see Figure 7.6), pulling it in direction of the inverted viewing vector, towards p_{start}, would at some point cause p_{eye} to cross the isosurface, which is forbidden. In real endoscopy, this restriction does not exist. A possible fix to this problem would be to use the reference points used for the calculation of force-feedback to define a path back to the nose, which leads around the deformed structure. Retreating along this path would, however cause rotations of the endoscope, not triggered by the user.

This has not yet been implemented. It might also not be needed, if geometric tissue defor-mation turns out to solve this problem.

8.4 Application for Preoperative Planning

The applicability of STEPS in preoperative planning was assessed in a series of 22 patients [WFD$^+$04]. The system was found useful in a variety of ways:

12 of 22 patients had a microadenoma. With these patients, visualization of the position of the pituitary gland and the tumor was found useful for the planning of an individually tailored approach to the tumor minimizing possible damage to the gland. In 10 of 22 pa-tients STEPS was recognized to facilitate planning of the operative removal of the tumor,

thus minimizing the risk of damage to the ICAs. STEPS aids the surgeon in assessing the dangers associated with removal of sphenoid septa overlying an ICA. The depth-enhanced semi-transparent display provided by STEPS allows intuitive spatial association of the segmented objects with foreground landmarks (e.g., septa), which also helps the surgeon to preoperatively determine landmarks in the decision as to where to open the sellar floor. Of the 22 patients studied, 15 had more than one sphenoid septum. In all of these patients, STEPS helped successfully identify the individual anatomy of the sphenoid sinus chambers, facilitating selection of the opening site. STEPS was also found to be useful for the decision which nostril to use as an entry point for the endoscope.

Chapter 9

A Virtual Transsphenoidal Procedure

This chapter illustrates the basic steps of a virtual transsphenoidal intervention, from the insertion of the endoscope into the nose to the resection of the tumor. The patient is a 49 year old male with a pituitary macroadenoma. Two background objects were segmented: the tumor and the ICAs, respectively.

9.1 Nasal Phase

Inspection of the patient's radiological data resulted in the decision to approach the adenoma through the patient's right nostril. Figure 9.1 shows that nostril from outside.

The endoscope is inserted into the nostril and advanced along the nasal septum (see Figure 9.2).

If structure and shape of the anatomy fail to give the user sufficient information for finding the correct route, additional navigation aids can be used. For instance, the background objects can be displayed by disabling the depth-dependency of foreground transparency (see Section 7.2). Figure 9.3 depicts the tumor (green) and the ICAs (red).

Soon the endoscope passes the middle nasal turbinate (see Figure 9.4), an important reference landmark for the surgeon.

After passing the middle nasal turbinate, the superior nasal turbinate becomes visible. Laterally to the superior nasal turbinate, the spheno-ethmoid recess and its opening to the sphenoid sinus, the sphenoid ostium are located (see Figure 9.5).

The sphenoid ostium in its natural state is too small to be traversed by the endoscope (see Figure 9.6).

The ostium must be enlarged beyond the endoscope diameter in order to increase the maneuverability of the endoscope during the sphenoidal phase of the procedure. Figure 9.7 shows the enlarged virtual ostium. The result of enlarging the ostium can be seen also in the
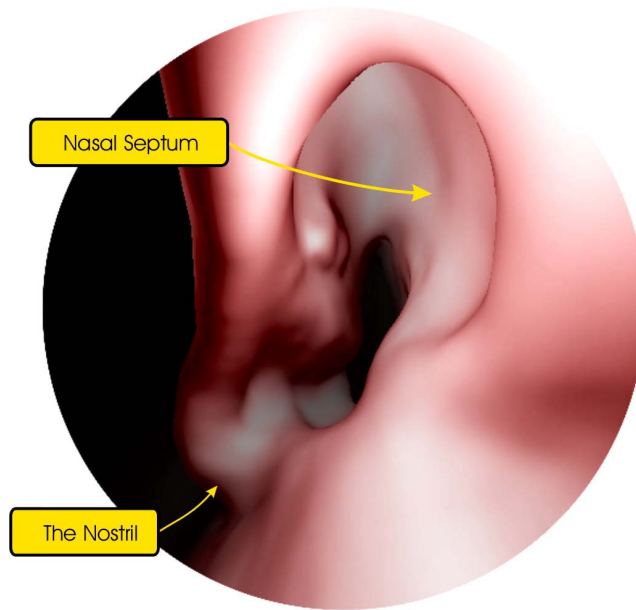
Figure 9.1: The endoscope is inserted into the nostril

MPR (see CT section in Figure 9.6).

9.2 Sphenoidal Phase

After the endoscope has traversed through the enlarged ostium, the interior of the sphenoid sinus becomes visible (see Figure 9.8). The tumor is occluded by the sellar floor straight ahead of the endoscope and by the intersinus septum separating the left and right chambers of the sphenoid sinus. If the procedure had been carried out through the other nostril, the camera would now be located on the far side of the septum.

The septum is removed using the virtual bone punch, facilitating access to the complete macroadenoma. The sellar floor is then opened, revealing the tumor. Figure 9.9 depicts the scene after this crucial part of the virtual intervention.

9.3 Sellar Phase

Now that full access to the tumor is gained, it can be removed. In reality, it is divided from healthy tissue and removed through the generated hole and, further, through the enlarged sphenoid ostium and the patient's nasal cavity and nostril.
In STEPS, the tumor can be resected using the virtual surgical instrument (see Figure 9.10).
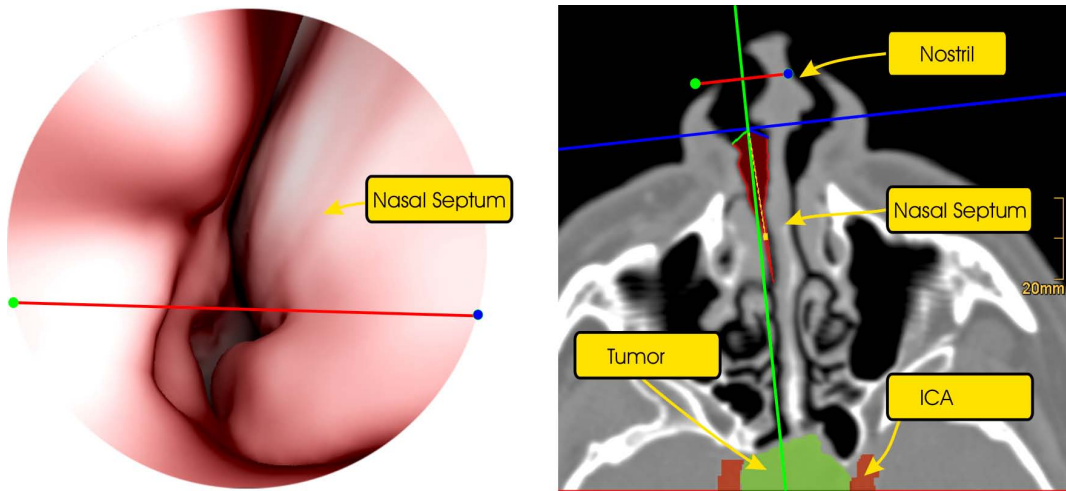
Figure 9.2: The endoscope is advanced along the nasal septum through the nasal cavity
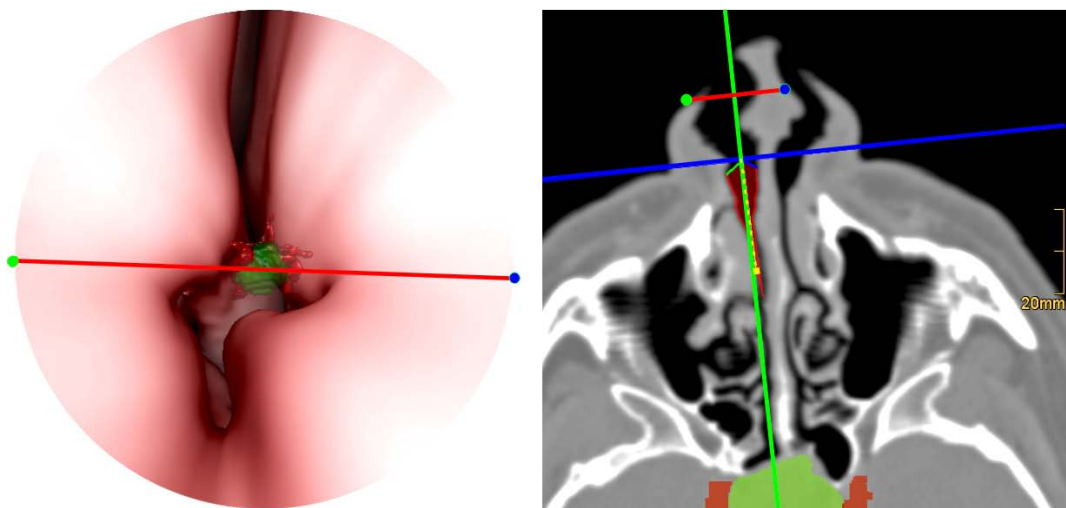
Figure 9.3: Background objects can be shown to facilitate navigation
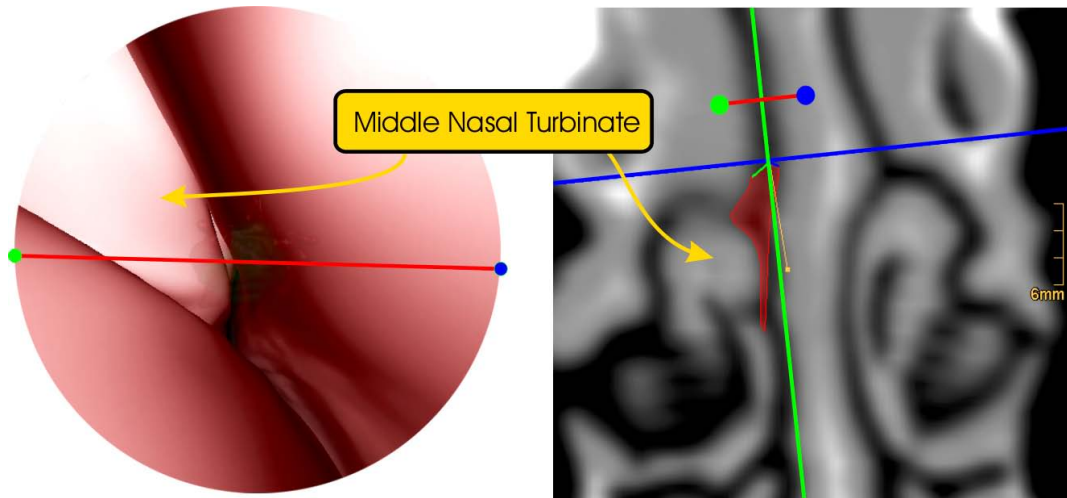
Figure 9.4: The middle nasal turbinate serves as a reference landmark
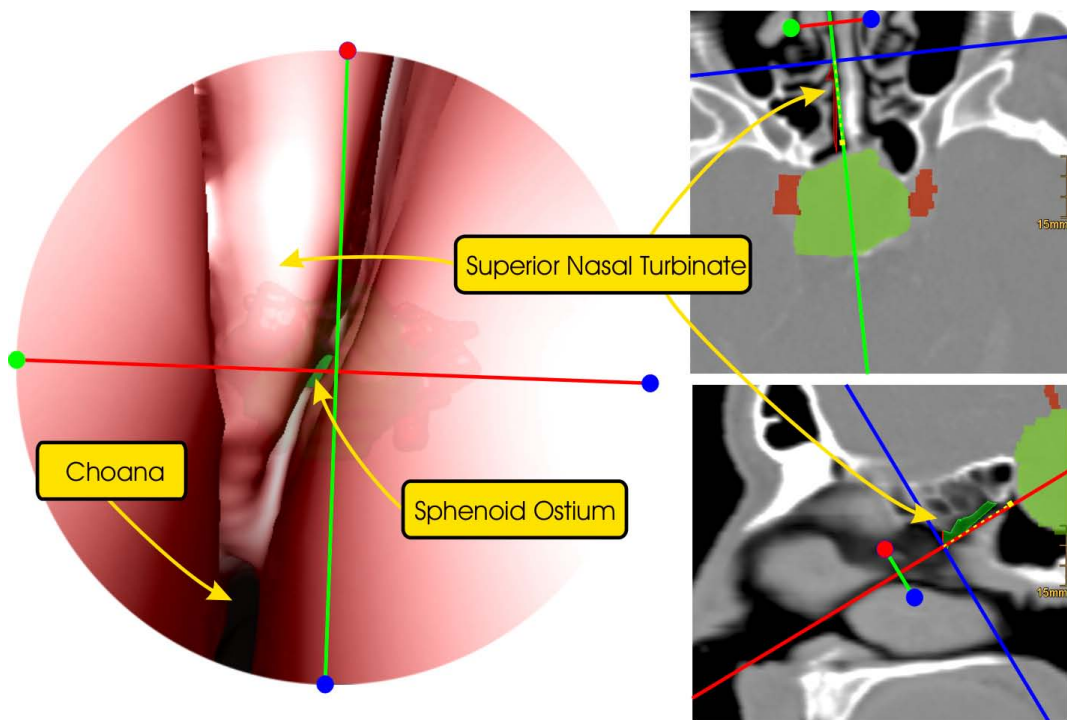
Figure 9.5: The superior nasal turbinate and the sphenoid ostium become visible
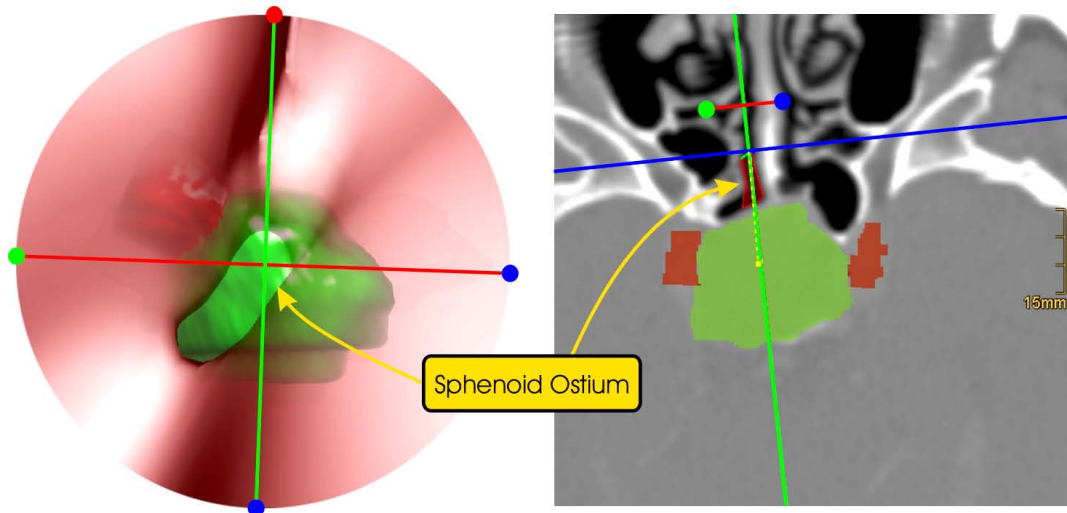
Figure 9.6: The ostium is too small to be traversed
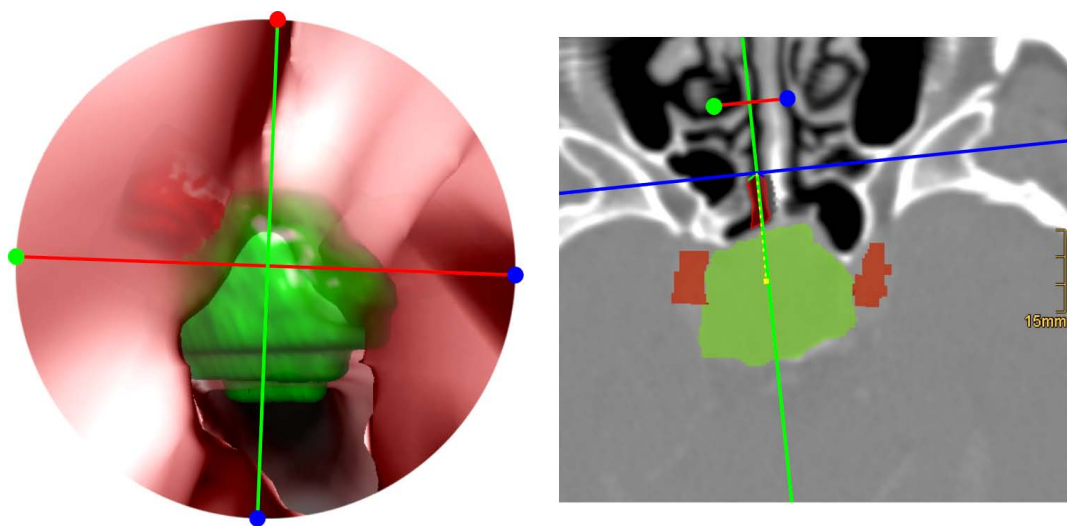
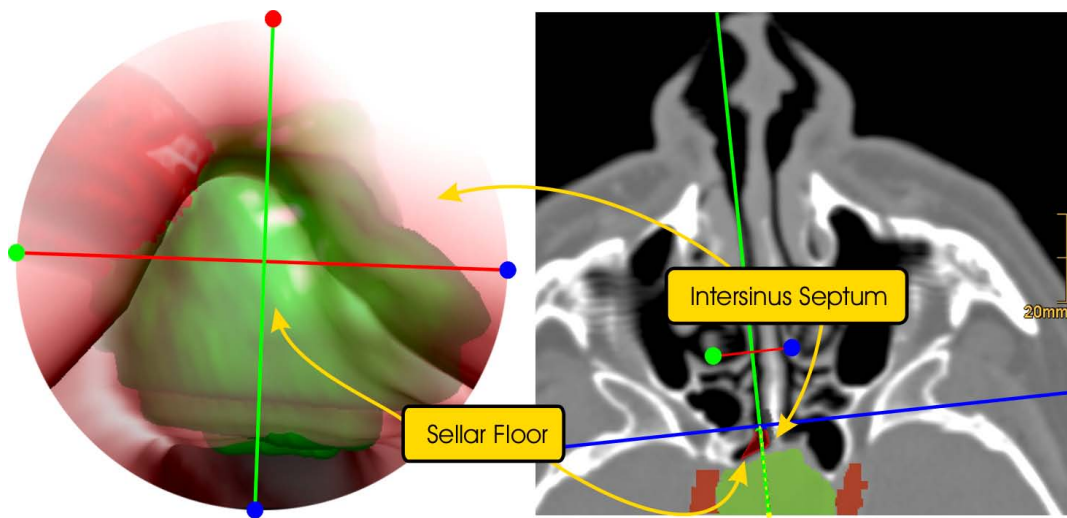Figure 9.7: Using the bone punch, the ostium is enlarged

Figure 9.8: Inside the sphenoid sinus, the tumor is occluded by the sellar floor and the intersinus septum
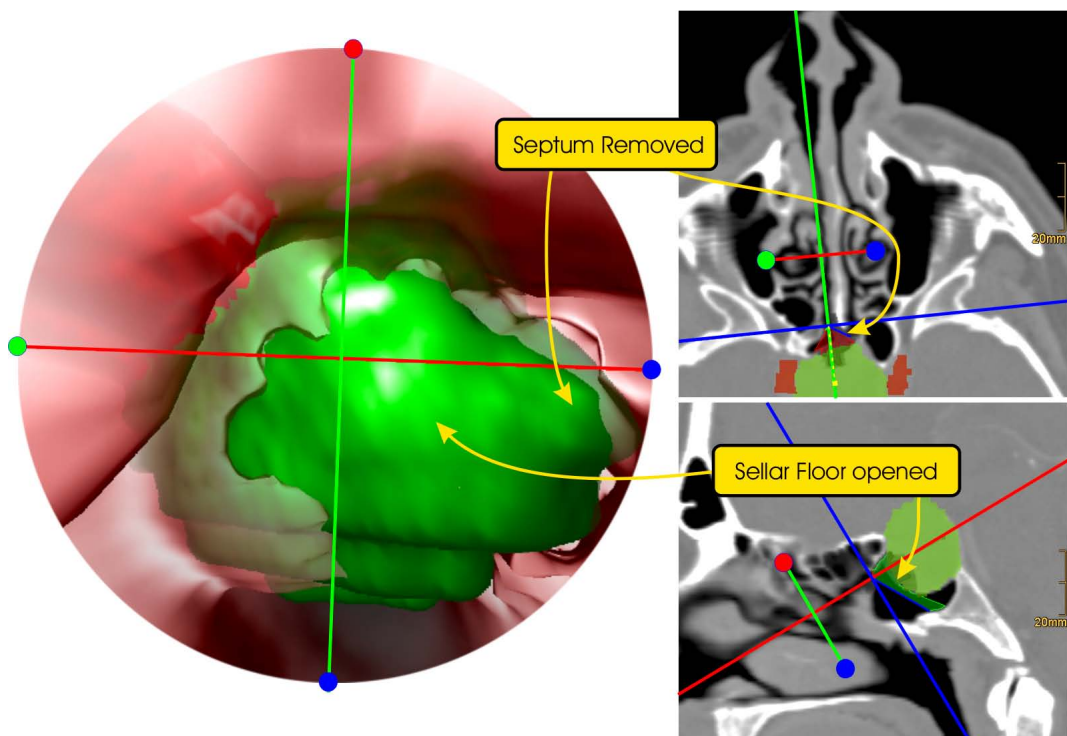
Figure 9.9: The septum has been removed and the sellar floor opened. Now the tumor can be resected.
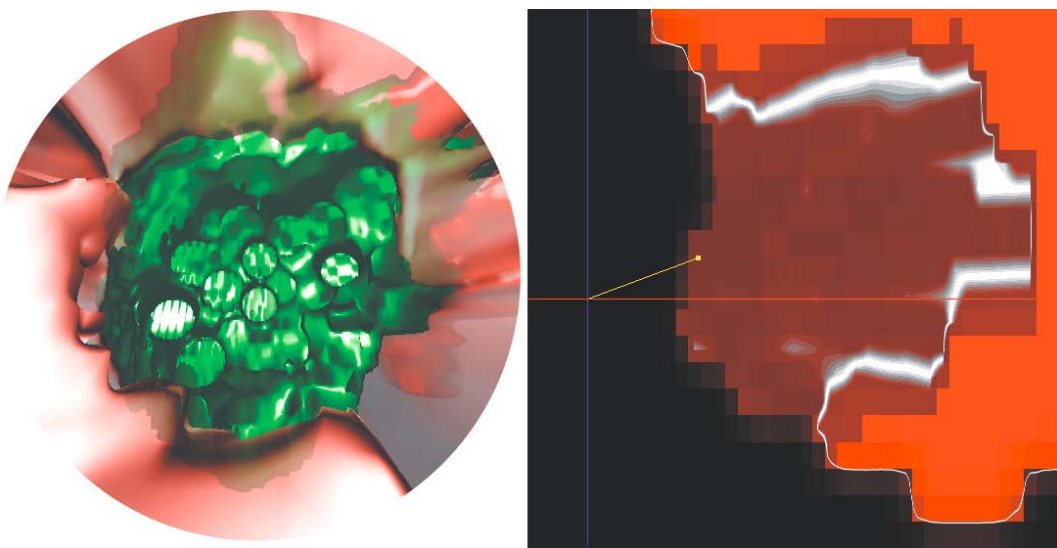
Figure 9.10: Left: Using the virtual surgical instruments, the tumor can be resected; Right: The virtual dissection is performed by adapting background volume values. Shades of red indicate background volume values, shades of gray indicate the proximity of the iso-surface depicting the tumor

Chapter 10

Conclusion and Future Work

This thesis introduced STEPS [NWF$^+$04, NWF$^+$05, WFD$^+$04] a virtual endoscopy system for pituitary surgery. The tool is used for two different purposes - preoperative assessment and training of inexperienced surgeons.

Preoperative assessment requires an optically accurate reconstruction of individual patient anatomy, including structures seen by the surgeon in real interventions and important background anatomy. STEPS uses entirely CPU-based rendering algorithms, ensuring a high amount of flexibility and hardware-independence. The surfaces of the investigated cavities and presegmented background objects of interest are rendered using iso-surfacing. The amount of visual information obtained can be enhanced by performing local volume rendering beyond surfaces, by rendering fractions of a second iso-surface, or by color-coding surface rigidity.

Currently a second version of the tool is being developed. It trades part of the flexibility of the current system for increased rendering quality and speed, using GPU-based ray casting. This new version will again depict a freely adjustable iso-surface and volume rendering applied from the iso-surface through the complete remaining volume. First results are presented in a recent publication [Sch05].

In conjunction with this new rendering style, experiments are currently being conducted to test, whether the segmentation of background objects of interest can be replaced by the application of special transfer functions. This, however, only seems possible, if a *fused* data volume, containing both CT and MRA data is rendered. More-dimensional transfer functions are being tested for their applicability.

A further hardware-extension planned for the future is the adaptation of the rendering software for display on stereo-displays. User studies have shown that screens generating the impression of real depth of field are accepted well by physicians and improve the mental reconstruction of the depicted anatomy.

The second field of application of STEPS is training of inexperienced neurosurgeons. For training purposes, a simulation of the procedure must be provided, presenting to the surgeon conditions and constraints of movement as experienced in real life. Finding the correct path from the nostril to the sphenoid sinus should therefore present similar difficulties to the surgeon as it does in the physical procedure. Additionally, the simulation of conducting basic surgical activity, including the sphenoidotomy, the removal of sphenoid septa, the opening of the sellar floor, and, possibly, even the removal of the tumor, provides the possi-

bility to embed the training process into a framework similar to a complete real procedure. This helps the novice in pituitary surgery gain some basic amount of routine which can then be further enhanced by training on cadavers.

In the current version of the software, the virtual endoscope is maneuvered using a force-feedback joystick during simulation. Compared with other existing endoscopic simulations, this is an inaccurate approximation of reality, but still works out fine at comparably very low hardware cost. The user-interface of a real endoscope can never be accurately modeled by means of a joystick, but the basic feeling can be provided. Therefore hands-eye coordination can be trained, also, as a valuable aspect, in relation to the visual response gained.

There has been a discussion about replacing the force-feedback joystick with a more elaborate haptic feedback device. Yet, even with the most sophisticated feedback device, the fact that the virtual procedure is performed inside a volume obtained from a medical modality and hence on a discrete approximation of reality, prevents a completely accurate simulation. Thus the training effect would probably not be notably increased, compared to the current system. So, the cheap solution using the joystick still seems a good choice.

Future improvement of the simulation offered by STEPS will include the optimization of the model used for geometric soft tissue deformation. This, providing more accurate response to physical tissue interaction, is expected to significantly improve the realism of the simulation.

A further planned extension of STEPS is to attach it to a tracking system and use it intra-operatively as a navigation device and to aid in strategic decisions during surgical activity, e.g., the decision where to open the sellar floor. The endoscopes, however, use optics with adjustable intrinsic parameters. Therefore, determination of the physical position and orientation of the endoscope are not sufficient to match a virtual to a reality image. The opening angle of the virtual camera must be adapted to account for changes of the focal parameters of the real camera. As soon as this has been accomplished, a combination of the real and the virtual image can be displayed. A possible feasible set-up would be to render background objects or volume rendered anatomy additionally to the real endoscopic image.

In the future, the tool shall be adapted for use in other fields of application. An example from neurosurgery is *third ventriculostomy*. The interior of cerebral ventricles will be depicted using surface rendering. In the background, important blood vessels (particularly the basilar artery) will be rendered, signaling the dangers associated with the procedure. The already existing virtual cutting tool can be used to simulate the ventriculostomy.

Aside from slight adaptations, the framework of STEPS can be used for almost any application of virtual endoscopy.

PUBLICATIONS AND AWARDS

List of publications:

- A. Neubauer, K. Bühler, R. Wegenkittl, A. Rauchberger and M. Rieger; **Advanced Virtual Corrective Osteotomy**. *Proceedings of Computer Assisted Radiology and Surgery (CARS 2005)*, June 2005.

- A. Neubauer, S. Wolfsberger, M.-T. Forster, L. Mroz, R. Wegenkittl and K. Bühler; **Advanced Virtual Endoscopic Pituitary Surgery**. *Accepted for publication in IEEE Transactions on Visualization and Computer Graphics (TVCG)* (to appear)

- A. Neubauer, S. Wolfsberger, M.-T. Forster, L. Mroz, R. Wegenkittl and K. Bühler; **STEPS - An Application for Simulation of Transsphenoidal Endonasal Pituitary Surgery**. *Proceedings of IEEE Visualization 2004*, pages 513-520, October 2004. (Best Applications Paper Award)

- S. Wolfsberger, M.-T. Forster, M. Donat, A. Neubauer, K. Bühler, R. Wegenkittl, T. Czech, J. A. Hainfellner and E. Knosp; **Virtual Endoscopy is a Useful Device for Training and Preoperative Planning of Transsphenoidal Endoscopic Pituitary Surgery**. *Minimally Invasive Neurosurgery, Issue 04, Volume 47*, pages 214-220, August 2004.

- A. Neubauer, M.-T. Forster, R. Wegenkittl, L. Mroz and K. Bühler; **Efficient Display of Background Objects for Virtual Endoscopy using Flexible First-Hit Ray Casting**. *Data Visualization 2004, Proceedings of the Joint Eurographics - IEEE TCVG Symposium on Visualization (VisSym 2004)*, pages 301-310, May 2004.

- A. Neubauer and R. Wegenkittl; **A Skeleton-Based Inflation Model for Myocardium Segmentation**. *Proceedings of Vision Interface (VI 2003)*, June 2003.

- A. Neubauer and R. Wegenkittl; **Analysis of Four-Dimensional Cardiac Data Sets Using Skeleton-Based Segmentation**. *Proceedings of Winter School of Computer Graphics (WSCG 2003)*, pages 330-337, February 2003.

- A. Neubauer and R. Wegenkittl; **Skeleton-Based Myocardium Segmentation**. *Proceedings of Visualization and Data Analysis (VDA 2003)*, pages 56-67, January 2003.

- A. Neubauer, L. Mroz, H. Hauser and R. Wegenkittl; **Cell-Based First-Hit Ray Casting**. *Data Visualization 2002, Proceedings of the Joint Eurographics - IEEE TCVG Symposium on Visualization (VisSym 2002)*, pages 77-86, May 2002.

- A. Neubauer, L. Mroz, H. Hauser and R. Wegenkittl; **Fast and Flexible Iso-Surfacing for Virtual Endoscopy**. *Proceedings of Central European Seminar on Computer Graphics (CESCG 2002)*, pages 41-52, April 2002.

- A. Neubauer and A. Kanitsar; **AlVis - Meeting the Tremendous Requirements Arising with the Visualization of Aluminium Foam Samples Investigated by High Resolution Industrial CT Modalities**. *Proceedings of Central European Seminar on Computer Graphics (CESCG 2000)*, May 2000. (Best Paper Award)

Awards:

- Best Applications Paper Award, *IEEE Visualization 2004*

- MedVis Award (Karl-Heinz-Höhne Preis) 2004

- Best Paper Award, *CESCG 2000*

ACKNOWLEDGEMENTS

time activities.

Thanks to the people working at the Institute of Computer Graphics and Algorithms at the Vienna University of Technology for interesting discussions, enjoyable common activities and, especially, for having and supporting some great kind of weird humor.

Last but not least I want to thank my family, especially my parents, Brigitte and Werner Neubauer, for their incredibly valuable support over all the years. My gratitude also goes to my sister Katrin and all my friends, simply for being there and making life more colorful.

Bibliography

[AAK97]   L. M. Auer, D. Auer, and J. F. Knoplioch. Virtual endoscopy for planning and simulation of minimally invasive neurosurgery. In *Proc. of First Joint Conference, Computer Vision, Virtual Reality and Robotics in Medicine and Medical Robotics and Computer-Assisted Surgery*, volume LNCS 1205, pages 315–318, 1997.

[AF00]   M. Ahmed and A. A. Farag. A neural optimization framework for zoom-lens camera. In *Proc. of IEEE International Conference on Computer Vision and Pattern Recognition*, pages 403–409, June 2000.

[AHR01]   T. A. A. Ayeni, D. Holmes, and R. A. Robb. Virtual angioscopic visualization and analysis of coronary aneurysms using intravascular ultrasound images. In *Proc. of SPIE Medical Imaging 2001: Physiology and Function From Multidimensional Images*, pages 44–52, 2001.

[AMKD02]   M. R. Ali, Y. Mowery, B. Kaplan, and E. J. DeMaria. Training the novice in laparoscopy. more challenge is better. *Surg Endosc*, 16(12):1732–1736, December 2002.

[AW87]   J. Amanatides and A. Woo. A fast voxel traversal algorithm for ray tracing. In *Proc. of Eurographics '87*, pages 3–10, 1987.

[Bar03a]   D. Bartz. Möglichkeiten und Grenzen der virtuellen Endoskopie. In *Proc. of Simulation und Visualisierung*, 2003.

[Bar03b]   D. Bartz. Virtual endoscopy in research and clinical practice. In *Eurographics State-of-the-Art-Reports 2003, (S4)*, 2003.

[Bau97]   R. Baumann. *Haptic interface for virtual reality based laparoscopic surgery training environment*. PhD thesis, Swiss Federal Institute of Technology, Lausanne, 1997.

[BBD+02]   J. Burtscher, R. Bale, A. Dessl, W. Eisner, K. Twerdy, R. A. Sweeney, and S. Felber. Virtual endoscopy for planning neuro-endoscopic intraventricular surgery. *Minim Invasive Neurosurg.*, 45(1):24–31, 2002.

[BDB+00]   J. Burtscher, A. Dessl, R. Bale, W. Eisner, A. Auer, K. Twerdy, and S. Felber. Virtual endoscopy for planning endoscopic third ventriculostomy procedures. *Pediatric Neurosurgery*, 32:77–82, 2000.

[BFGS86]   L. Bergman, H. Fuchs, E. Grant, and S. Spach. Image rendering by adaptive refinement. *Computer Graphics*, 20(4):29–37, August 1986.

[BFMZ94]   G. Bishop, H. Fuchs, L. McMillan, and E. J. Scher Zagier. Frameless rendering: Double buffering considered harmful. In *Proc. of SIGGRAPH '94*, pages 175–176, 1994.

[BGG98]   C. Baur, D. Guzzoni, and O. Georg. Virgy: A virtual reality and force feedback based endoscopy surgery simulator. In *Proc. of MMVR '98*, pages 110–116, 1998.

[BGL+01]   D. Bartz, Ö. Gürvit, M. Lanzendörfer, A. Kopp, A. Küttner, and W. Straßer. Virtual endoscopy for cardio vascular exploration. In *Proc. of Computer Assisted Radiology and Surgery (CARS 01)*, pages 960–964, 2001.

[BMF+03]   D. Bartz, D. Mayer, J. Fischer, S. Ley, A. del Rio, S. Thust, C. Heussel, H. Kauczor, and W. Straßer. Hybrid segmentation and exploration of the human lungs. In *Proc. of IEEE Visualization*, 2003.

[BMV+99]   U. Bockholt, W. Müller, G. Voss, U. Ecke, and L. Klimek. Real-time simulation of tissue deformation for the nasal endoscopy simulator (NES). *Comput Aided Surg.*, 4(5):281–285, 1999.

[BNH+05]   K. Bühler, A. Neubauer, M. Hadwiger, S. Wolfsberger, and R. Wegenkittl. Interactive 3D techniques for computer aided diagnosis and surgery simulation tools. In W. Hruby, editor, *Digital (R)evolution in Radiology, Vol 2*. Springer, 2005.

[Bru04]   Stefan Bruckner. Efficient volume visualization of large medical datasets. Master's thesis, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria, may 2004.

[BS99]   D. Bartz and M. Skalej. VIVENDI - a virtual ventricle endoscopy system for virtual medicine. In *Data Visualization (Proc. of Symposium on Visualization)*, pages 155–166, 1999.

[BSG+01]   D. Bartz, W. Straßer, Ö. Gürvit, D. Freudenstein, and M. Skalej. Interactive and multi-modal visualization for neuroendoscopic interventions. In *Data Visualization (Proc. of Symposium on Visualization)*, pages 157–164, 2001.

[BSSW99]   D. Bartz, W. Straßer, M. Skalej, and D. Welte. Interactive exploration of extra- and intracranial blood vessels. In *Proc. of IEEE Visualization*, pages 389–392, 1999.

[BVS+04]   S. Bisdas, M. Verink, M. Stieve, A. Lazakidou, I. Evangelou, and H. Becker. Three dimensional imaging of the nasal cavity and paranasal sinuses. In *Proc. of Special Mini-Symposium BIOHEMED' 04 within the context of the 4th WSEAS International Conference ISA' 04*, 2004.

[CAdDT01] P. Cappabianca, A. Alfieri, E. de Divitiis, and M. Tschabitscher. *Atlas of Endoscopic Anatomy for Endonasal Intracranial Surgery.* Springer, 2001.

[CCS⁺02] P. Carrascosa, R. Corti, G. Sangster, C. Capunay, G. Suinbourn, L. Schenone, A. Giordano, J. Doweck, G. Menendez, J. Valero, O. Zerbo, and J. Carrascosa. Virtual gastroscopy. Preliminary experience. *Acta Gastroenterol Latinoam*, 32(1):11–15, 2002.

[CET98] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In *Proc. of European Conference on Computer Vision 1998*, volume 2, pages 484–498, 1998.

[Che95] S. E. Chen. Quicktime vr - an image-based approach to virtual environment navigation. In *Proc. of SIGGRAPH 95*, pages 29–38, August 1995.

[cK99] H. K. Çakmak and U. Kühnapfel. The ”karlsruhe endoscopic surgery trainer” for minimally invasive surgery in gynaecology. In *Proc. of Congress on Computer Assisted Radiology and Surgery (CARS ’99)*, June 1999.

[cK00] H. K. Çakmak and U. Kühnapfel. Animation and simulation techniques for vr-training systems in endoscopic surgery. In *Proc. of EGCAS*, pages 173–185, 2000.

[CLW⁺00] D. Chen, Z. Liang, M. Wax, L. Li, B. Li, and A. Kaufman. A novel approach to extract colon lumen from CT images for virtual colonoscopy. *IEEE Transactions on Medical Imaging*, 19:1220–1226, 2000.

[CN93] T. Cullip and U. Neumann. Accelerating volume reconstruction with 3D texture hardware. In *Technical Report TR93-027, University of North Carolina at Chapel Hill*, 1993.

[CRK⁺99] M. J. Clarkson, D. Rueckert, A. P. King, P. J. Edwards, D. L. G. Hill, and D. J. Hawkes. Registration of video images to tomographic images by optimising mutual information. In *Proc. of Medical Image Computing and Computer-Assisted Intervention (MICCAI ’99)*, pages 579–588, 1999.

[CS04] J. R. Cebral and R. M. Summers. Tracheal and central bronchial aerodynamics using virtual bronchoscopy and computational fluid dynamics. *IEEE Transactions on Medical Imaging*, 23(8):1021–1033, August 2004.

[CW01] M. Capek and R. Wegenkittl. Robust and fast medical registration of 3D-multi-modality data sets. In *Proc. of Medicon ’01*, pages 515–518, 2001.

[CWK⁺01] M. Capek, R. Wegenkittl, A. König, W. Jaschke, R. Sweeny, and R. Bale. Multimodal medical volume registration based on spherical markers. In *Proc. of WSCG ’01*, page 17 ff, 2001.

[dBVP⁺00] P. W. de Bruin, F. Vos, F. H. Post, S. F. Gibson, and A. M. Vossepoe. Improving triangle mesh quality with surface nets. In *Proc. of Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2000)*, 2000.

[dDCC03]   E. de Divitiis, P. Cappabianca, and L. M. Cavallo. Endoscopic endonasal transsphenoidal approach to the sellar region. In E. de Divitiis and P. Cappabianca, editors, *Endoscopic Endonasal Transsphenoidal Surgery*, pages 91–130. Springer, 2003.

[DGS+02]   D. Dey, D. G. Gobbi, P. J. Slomka, K. J. Surry, and T. M. Peters. Automatic fusion of freehand endoscopic brain images to three-dimensional surfaces: Creating stereoscopic panoramas. *IEEE Transaction on Medical Imaging*, 21(1):23–30, 2002.

[DKW+01]   F. Dachille, K. Kreeger, M. Wax, A. Kaufman, and Z. Liang. Interactive navigation for pc-based virtual colonoscopy. In *Proc. of SPIE Medical Imaging 2001*, pages 500–504, 2001.

[DLR+96]   C. P. Davis, M. E. Ladd, B. J. Romanowski, S. Wildermuth, J. F. Knoplioch, and J. F. Debatin. Human aorta: preliminary results with virtual endoscopy based on three- dimensional MR imaging data sets. *Radiology*, 199:37–40, 1996.

[DMTR+00] F. Dammann, E. Momino-Traserra, C. Remy, P. Pereira, I. Baumann, A. Koitschev, and C. Claussen. Radiation exposure during spiral-CT of the paranasal sinuses. *RoFo 2000*, 172(3):232–237, March 2000.

[Drä05]   C. Dräger. A ChainMail algorithm for direct volume deformation in virtual endoscopy applications. Master's thesis, VRVis Research Center/Vienna University of Technology, 2005.

[dRBJ+03] Á. del Río, D. Bartz, R. Jäger, Ö. Gürvit, and D. Freudenstein. Efficient steoroscopic rendering in virtual endoscopy applications. In *WSCG*, 2003.

[DZT+89]   B. Dubuc, S. W. Zucker, C. Tricot, J. F. Quiniou, and D. Wehbi. Evaluating the fractal dimension of surfaces. In *Proc. of R Soc Lond*, pages 113–127, 1989.

[FGG+02]   A. Ferlitsch, P. Glauninger, A. Gupper, M. Schillinger, M. Haefner, A. Gangl, and R. Schoefl. Evaluation of a virtual endoscopy simulator for training in gastrointestinal endoscopy. *Endoscopy 2002*, 34:698–702, 2002.

[FMA+97]   T. Fleiter, E. M. Merkle, AJ. Aschoff, G. Lang, M. Stein, J. Gorich, F. Liewald, N. Rilinger, and R. Sokiranski. Comparison of real-time virtual and fiberoptic bronchoscopy in patients with bronchial carcinoma: opportunities and limitations. *American Journal of Roentgenology*, 169:1591–1595, 1997.

[FNS+99]   H. M. Fenlon, D. P. Nunes, P. C. Schroy, M. A. Barish, P. D. Clarke, and J. T. Ferrucci. Occlusive colon carcinoma: virtual colonoscopy in the preoperative evaluation of the proximal colon. *Radiology 1999*, 210:423–428, 1999.

[For03]   M.-T. Forster. Virtuelle Endoskopie in der transnasalen Transsphenoidalen Hypophysenchirurgie, doctoral thesis, universität wien, 2003.

[FPBAG04]   C. Fetita, F. Prêteux, C. Beigelman-Aubry, and P. Grenier. Pulmonary air-ways: 3D reconstruction from multi-slice CT and clinical investigation. *IEEE Transactions on Medical Imaging*, 23(11):1353–1364, November 2004.

[FSHM02]   A. A. Farag, C. B. Sites, S. Hushek, and T. Moriarty. Virtual endoscopy: Modeling the navigation in 3D brain volumes. In *Proc. of International Conference on Biomedical Engineering*, 2002.

[FWB01]   P. Felkel, R. Wegenkittl, and M. Bruckschwaiger. Implementation and complexity of the watershed-from-markers algorithm computed as a minimal cost forest. In *Proc. of Eurographics '01*, pages 26–35, 2001.

[GBKG04]   Sören Grimm, Stefan Bruckner, Armin Kanitsar, and Meister Eduard Gröller. Memory efficient acceleration structures and techniques for cpu-based volume raycasting of large data. In C. Silva D. Silver, T. Ertl, editor, *Proceedings IEEE/SIGGRAPH Symposium on Volume Visualization and Graphics*, pages 1–8, October 2004.

[Gib97]   S. F. Gibson. 3D ChainMail: a fast algorithm for deforming volumetric objects. In *Proc. of Symposium on interactive 3D Graphics, ACM SIGGRAPH*, pages 149–154, 1997.

[Gib98a]   S. F. Gibson. Constrained elastic surface nets: generating smooth surfaces from binary segmented data. In *Proc. of Proceedings of Medical Image Computation and Computer Assisted Surgery (MICCAI 98)*, 1998.

[Gib98b]   S. F. Gibson. Using distance maps for accurate surface reconstruction in sampled volumes. In *Proc. of IEEE Symposium on Volume Visualization*, pages 23–30, 1998.

[GM85]   A. Gagalowicz and S. D. Ma. Sequential synthesis of natural textures. *Computer Vision, Graphics, and Image Processing*, 30:289–315, 1985.

[GP01]   C. D. Gandhi and K. D. Post. Historical movements in transsphenoidal surgery. *Neurosurg Focus*, 11(4), 2001.

[GPPA04]   K. R. Ghani, J. Pilcher, U. Patel, and K. Anson. Three-dimensional imaging in urology. *BJU Int.*, 94(6):769–773, October 2004.

[GPZT98]   E. Gobbetti, P. Pili, A. Zorcolo, and M. Tuveri. Interactive virtual angioscopy. In *Proc. of IEEE Visualization*, pages 435–438, 1998.

[Gri05]   Sören Grimm. *Real-Time Mono- and Multi-Volume Rendering of Large Medical Datasets on Standard PC Hardware*. PhD thesis, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria, April 2005.

[HBH03]   M. Hadwiger, C. Berger, and H. Hauser. High-quality two-level volume rendering of segmented data sets on consumer graphics hardware. In *Proc. of IEEE Visualization*, pages 301–308, 2003.

[HCO⁺03]   M. Haliloglu, A. O. Ciftci, A. Oto, B. Gumus, F. C. Tanyel, M. E. Senocak, N. Buyukpamukcu, and A. Besim. CT virtual bronchoscopy in the evaluation of children with suspected foreign body aspiration. *European Journal of Radiology*, January 2003.

[HDL⁺01]   S. He, R. Dai, B. Lu, C. Cao, H. Bai, and B. Jing. Medial axis reformation: a new visualization method for CT angiography. *Acad Radiol.*, 8(8):726–733, August 2001.

[HEN⁺04]   J. Hochberger, K. Euler, A. Naegel, E. G. Hahn, and J. Maiss. The compact erlangen active simulator for interventional endoscopy: a prospective comparison in structured team-training courses on 'endoscopic hemostasis' for doctors and nurses to the 'endo-trainer' model. *Scandinavian Journal of Gastroenterology*, 39(9):895–902, September 2004.

[HHCL01]   T. He, L. Hong, D. Chen, and Z. Liang. Reliable path for virtual endoscopy: Ensuring complete examination of human organs. *IEEE Transactions on Computer Graphics and Visualization*, 7(4):333–342, 2001.

[HHK⁺95]   T. He, L. Hong, A. Kaufman, A. Varshney, and S. Wang. Voxel based object simplification. In *Proc. of IEEE Visualization 95*, pages 296–303, October 1995.

[HHP03]   W. E. Higgins, J. P. Helferty, and D. R. Padfield. Integrated bronchoscopic video tracking and 3D CT registration for virtual bronchoscopy. In *Proc. of SPIE Medical Imaging 2003: Physiology and Function — Methods, Systems, and Applications*, pages 80–89, 2003.

[HJR⁺96]   A. K. Hara, C. D. Johnson, J. E. Reed, D. A. Ahlquist, H. Nelson, R. L. Ehman, C. H. McCollough, and D. M. Ilstrup. Detection of colorectal polyps by computed tomographic colography: feasibility of a novel technique. *Gastroenterology 1996*, 110:284–290, 1996.

[HKW⁺95]   L. Hong, A. Kaufman, Y. Wei, A. Viswambharan, M. Wax, and Z. Liang. 3D virtual colonoscopy. In *Proc. of IEEE Symposium on Biomedical Visualization*, pages 26–32, 1995.

[HLD⁺94]   W. A. Hall, M. G. Luciano, J. L. Doppman, N. J. Patronas, and E. H. Oldfield. Pituitary magnetic resonance imaging in normal human volunteers: occult adenomas in the general population. *Ann Intern Med*, 120(10):817–820, 1994.

[HMK⁺97]   L. Hong, S. Muraki, A. Kaufman, D. Bartz, and T. He. Virtual voyage: Interactive navigation in the human colon. In *Proc. of ACM SIGGRAPH*, pages 27–34, 1997.

[HNea97]   J. Hochberger, M. Neumann, and W. Hohenberger et al. EASIE-Erlangen education simulation model for interventional endoscopy a new biotraining model for surgical endoscopy. *Biomed Tech (Berl)*, 42, 1997.

142

[HO00]   R. Hietala and J. Oikarinen. A visibility determination algorithm for interactive virtual endoscopy. In *Proc. of IEEE Visualization*, pages 29–36, 2000.

[Hop96]   Hugues Hoppe. Progressive meshes. *Computer Graphics*, 30(Annual Conference Series):99–108, 1996.

[HSS+00]   T. Himi, M. Sakata, T. Shintani, H. Mitsuzawa, M. Kamagata, J. Satoh, and H. Sugimoto. Middle ear imaging using virtual endoscopy and its application in patients with ossicular chain anomaly. *Otorhinolaryngol Relat Spec.*, 62(6):316–320, 2000.

[JBS+97]   P. Jannin, A. Bouliou, J. M. Scarabin, C. Barillot, and J. Luber. Visual matching between real and virtual image guided neurosurgery. In *Proc. of SPIE Medical Imaging*, pages 518–526, 1997.

[Jon96]   M. W. Jones. The production of volume data from triangular meshes using voxelisation. *Computer Graphics Forum*, 15(5):311–318, December 1996.

[KAP+02]   J. K. Kim, J. H. Ahn, T. Park, H. J. Ahn, C. S. Kim, and K. S. Cho. Virtual cystoscopy of the contrast material-filled bladder in patients with gross hematuria. *American Journal of Roentgenology*, 179:763–768, 2002.

[KB89]   D. Kalra and A. H. Barr. Guaranteed ray intersections with implicit surfaces. *Computer Graphics*, 23(3):297–306, July 1989.

[KcM00]   U. Kühnapfel, H. K. Çakmak, and H. Maaß. Endoscopic surgery training using virtual reality and deformable tissue simulation. *Computers & Graphics*, 24:671–682, 2000.

[KGJ95]   R. Kikinis, P. Langham Gleason, and F. A. Jolesz. Surgical planning using computer-assisted three-dimensional reconstructions. *Taylor R, Lavalle S, Burdea G, Mosges R, editors. Computer integrated surgery. Cambridge, MA: MIT Press*, pages 147–154, 1995.

[KH84]   J. T. Kajiya and B. P. Von Herzen. Ray tracing volume densities. In *Proc. of SIGGRAPH '84*, pages 165–174, 1984.

[KIMW00]   K. Kneöaurek, M. Ivanovic, J. Machac, and D. A. Weber. Medical image registration. *Europhysics News*, 31(4), 2000.

[KJY03]   J. Koloszár and Y. Jae-Young. Accelerating virtual endoscopy. In *Proc. of WSCG*, 2003.

[KM02]   O. Körner and R. Männer. Haptic display for a virtual reality simulator for flexible endoscopy. In *Proc. of 8th Eurographics Workshop on Virtual Environments*, pages 13–18, 2002.

[KPK+01]   G. C. Kagadis, V. Patrinou, C. P. Kalogeropoulou, D. Karnabatidis, T. Petsas, G. C. Nikiforidis, and D. Dougenis. Virtual endoscopy in the diagnosis of an adult double tracheal bronchi case. *European Journal of Radiology*, 40:50–53, 2001.

[KSK+99]   Y. Kato, H. Sano, K. Katada, Y. Ogura, M. Hayakawa, and N. Kanaoka. Applications of three-dimensional CT angiography (3D-CTA) to cerebral aneurisms. *Surg Neurol*, 52:113–121, 1999.

[KST98]   W. Konen, M. Scholz, and S. Tombrock. The VN project: endoscopic image processing for neurosurgery. *Comput Aided Surgery*, 3(3):144–148, 1998.

[KVM+00]   S. Kourelea, T. Vontetsianos, V. Maniatis, K. Lyberopoulos, A. Rigopulou, M. Mandeou, C. Michael, and A. Marsh. The application of virtual bronchoscopy in the evaluation of hemoptysis: comparative evaluation with real fiberoptic bronchoscopy. In *Proc. of IEEE EMBS*, 2000.

[LBS+02]   M. Lammle, A. Beer, M. Settles, C. Hannig, H. Schwaibold, and C. Drews. Reliability of MR imaging-based virtual cystoscopy in the diagnosis of cancer of the urinary bladder. *American Journal of Roentgenology*, 178:1483–1488, 2002.

[LC87]   W. E. Lorensen and H. E. Cline. Marching cubes: a high resolution 3D surface construction algorithm. In *Proc. of SIGGRAPH'87*, pages 163–169, 1987.

[LCL+02]   S. Lakare, D. Chen, L. Li, A. Kaufman, and J. Liang. Electronic colon cleansing using segmentation rays for virtual colonoscopy. In *Proc. of SPIE 2002 Symposium on Medical Imaging 2002*, February 2002.

[LE97]   D. Luebke and C. Erikson. View-dependent simplification of arbitrary polygon environments. In *Proc. of ACM SIGGRAPH*, 1997.

[Lev88]   M. Levoy. Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, 8(5):29–37, 1988.

[Lev90a]   M. Levoy. Efficient ray tracing of volume data. *ACM Transactions on Graphics*, 9(3):245–261, July 1990.

[Lev90b]   M. Levoy. Volume rendering by adaptive refinement. *The Visual Computer*, 6(1):2–7, 1990.

[LH91]   D. Laur and P. Hanrahan. Hierarchical splatting: A progressive refinement algorithm for volume rendering. *Computer Graphics*, 25(4):285–288, July 1991.

[LHB05]   C. Langer, M. Hadwiger, and K. Bühler. Interaktive diffusionsbasierte Segmentierung von Volumendaten auf Grafikhardware. In *Proc. of Bildverarbeitung in der Medizin*, March 2005.

[Lia01]   J. Z. Liang. Virtual colonoscopy: An alternative approach to examination of the entire colon. *INNERVISION*, 16(10):40–44, 2001.

[LJK95]   W. E. Lorensen, F. A. Jolesz, and R. Kikinis. The exploration of crosssectional data with a virtual endoscope. *K. Morgan, R. M. Satava, H. B. Sieburg, R. Mattheus, J. P. Christensen, editors. Interactive technology and the new paradigm for healthcare. Ohmsha: IOS Press*, pages 221–230, 1995.

[LK04]   S. Lakare and A. Kaufman. Light weight space leaping using ray coherence. In *Proc. of IEEE Visualization '04*, pages 19–26, 2004.

[LKS01]   J.-J. Lee, Y.-M. Koo, and Y. G. Shin. User-friendly environment for virtual endoscopy. In *Proc. of The Fifth Korea-Germany Joint Workshop on Advanced Medical Image Processing*, May 2001.

[LL94]   P. Lacroute and M. Levoy. Fast volume rendering using a shearwarp factorization of the viewing transformation. In *Proc. of SIGGRAPH94*, pages 451–458, July 1994.

[LSJ96]   Y. Livnat, H.W. Shen, and C.R. Johnson. A near optimal isosurface extraction algorithm using the span space. *IEEE Transactions on Visualization and Computer Graphics*, 2(1):297–306, March 1996.

[LWCK99]   W. Li, M. Wan, B. Chen, and A. Kaufman. Virtual colonoscopy powered by volumepro. Indexed at `http://www.cs.sunysb.edu/`, 1999.

[LY96]   A. Law and R. Yagel. Multi-frame thrashless ray casting with advancing rayfront. In *Proc. of Graphics Interfaces 1996*, pages 70–77, May 1996.

[LYW+97]   Z. Liang, F. Yang, M. Wax, J. Li, J. You, A. Kaufman, L. Hong, H. Li, and A. Viswambharan. Inclusion of a priori information in segmentation of colon lumen for 3D virtual colonoscopy, 1997.

[MCC+98]   A. Morra, A. Calgaro, V. Cioffi, M. Pravato, M. Cova, and R. Pozzi Mucelli. Virtual endoscopy of the nasal cavity and the paranasal sinuses with computerized tomography. anatomical study. *Radiol Med*, 96:29–34, 1998.

[MCPA02]   S. Mak, G. Cattini, U. Patel, and K. Anson. Virtual ureterorenoscopy and threedimensional pelvicalyceal anatomy as a guide to endoscopic navigation feasibility studies in a pig kidney model. *J Urol*, 167(4), 2002.

[MCV+97]   F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality image registration by maximization of mutual information. *IEEE Transactions on Medical Imaging*, 16(2):187–198, 1997.

[MH01]   L. Mroz and H. Hauser. RTVR - a flexible java library for interactive volume rendering. In *Proc. of IEEE Visualization 2001*, pages 279–286, 2001.

[MHS+02]   K. Marukawa, J. Horiguchi, M. Shigeta, T. Nakamoto, T. Usui, and K. Ito. Threedimensional navigator for retroperitoneal laparoscopic nephrectomy using multidetector row computerized tomography. *J Urol*, 168:1933–1936, 2002.

[MKS98]   M. Meißner, U. Kanus, and W. Straßer. VIZARD II, A PCI-Card for Real-Time Volume Rendering. In *Proc. Eurographics/Siggraph Workshop on Graphics Hardware*, pages 61–68, 1998.

[MS96]   C. M. Ma and M. Sonka. A fully parallel 3D thinning algorithm and its applications. *Computer Vision and Image Understanding*, 64(3):420–433, 1996.

[MV98]   J. B. A. Maintz and M. A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):1–36, 1998.

[NBW+05]   A. Neubauer, K. Bühler, R. Wegenkittl, A. Rauchberger, and M. Rieger. Advanced virtual corrective osteotomy. In *Proc. of Computer Assisted Radiology and Surgery (CARS 05)*, 2005.

[Neu01]   A. Neubauer. Cell-based first-hit ray casting. Master's thesis, Vienna University of Technology, 2001.

[NFW+04]   A. Neubauer, M. Forster, R. Wegenkittl, L. Mroz, and K. Bühler. Efficient display of background objects for virtual endoscopy using flexible first-hit ray casting. In *Proc. Eurographics/IEEE TCVG Symposium on Visualization (VisSym '04)*, pages 301–310, 2004.

[NMHW02] A. Neubauer, L. Mroz, H. Hauser, and R. Wegenkittl. Cell-based first-hit ray casting. In *Proc. Eurographics/IEEE TCVG Symposium on Visualization (VisSym '02)*, page 77 ff, 2002.

[NRB+96]   S. Napel, G. Rubin, C. Beaulieu, R. Jeffrey, and V. Argiro. Perspective volume rendering of cross-sectional images for simulated endoscopy and intraparenchymal viewing. In *Proc. of SPIE Medical Imaging 96*, pages 75–86, 1996.

[NSG90]   K. L. Novins, F. X. Sillion, and D. P. Greenberg. An efficient method for volume rendering using perspective projection. *Computer Graphics*, 24(5):95–100, November 1990.

[NSS97]   M. De Nicola, L. Salvolini, and U. Salvolini. Virtual endoscopy of nasal cavity and paranasal sinuses. *Eur J Radiol*, 24:175–180, 1997.

[NW03a]   A. Neubauer and R. Wegenkittl. Analysis of four-dimensional cardiac data sets using skeleton-based segmentation. In *Proc. of Winter School of Computer Graphics (WSCG 2003)*, pages 330–337, 2003.

[NW03b]   A. Neubauer and R. Wegenkittl. A skeleton-based inflation model for myocardium segmentation. In *Proc. of Vision Interface (VI 2003)*, 2003.

[NW03c]   A. Neubauer and R. Wegenkittl. Skeleton-based myocardium segmentation. In *Proc. of Visualization and Data Analysis 2003*, pages 56–67, 2003.

[NWF+04]   A. Neubauer, S. Wolfsberger, M.-T. Forster, L. Mroz, R. Wegenkittl, and K. Bühler. STEPS - an application for simulation of transsphenoidal endonasal pituitary surgery. In *Proc. of IEEE Visualization 2004*, pages 513–520, October 2004.

[NWF+05] A. Neubauer, S. Wolfsberger, M.-T. Forster, L. Mroz, R. Wegenkittl, and K. Bühler. Advanced virtual endoscopic pituitary surgery. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 2005.

[OK95] R. L. Ogniewicz and O. Kübler. Hierarchic Voronoi skeletons. *Pattern Recognition*, 28(3):343–359, 1995.

[PEO+04] Haigron P., Bellemare M. E., Acosta O., Goksu C., Kulik C., Rioual K., and Lucas A. Depth-map-based scene analysis for active navigation in virtual angioscopy. *IEEE Transactions on Medical Imaging*, 23(11):1380–1390, November 2004.

[PFP04] D. Perchet, C. Fetita, and F. Prêteux. Advanced navigation tools for virtual bronchoscopy. In *Proceedings SPIE Conference on Image Processing: Algorithms and Systems III - IS&T / SPIE Symposium on Electronic Imaging, Science and Technology '04, San Jose, CA*, volume 5298, January 2004.

[PMV03] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever. Mutual-information-based registration of medical images: a survey. *IEEE Transactions on Medical Imaging*, 22:986–1004, 2003.

[Pra02] A. Prando. CT-virtual endoscopy of the urinary tract. *Int Braz J Urol.*, 28:317–322, 2002.

[PSL+98] S. Parker, P. Shirley, Y. Livnat, C. Hansen, and P. Sloan. Interactive ray tracing for isosurface rendering. In *Proc. of IEEE Visualization '98*, pages 233–238, 1998.

[QWQK00] H. Qu, M. Wan, J. Qin, and A. Kaufman. Image based rendering with stable frame rates. In *Proc. of IEEE Visualization 2000*, pages 251–258, 2000.

[RC95] R. A. Robb and B. Cameron. Virtual reality assisted surgery program. *Morgan K, Satava RM, Sieburg HB, Mattheus R, Christensen JP, editors. Interactive technology and the new paradigm for healthcare*, 1995.

[Rex02] D. K. Rex. Considering virtual colonoscopy. *Reviews in Gastroenterological Disordes*, 2(3):97–105, 2002.

[RMRH01] P. Rogalla, N. Meiri, J. C. Rückert, and B. Hamm. Multisclice CT colonography. *Medical Review*, (81):15–18, 2001.

[Rob00] R. A. Robb. Virtual endoscopy: development and evaluation using the visible human datasets. *Comput Med Imaging Graph. 2000*, 24(3):125–126, 2000.

[Rog99] P. Rogalla. Virtual endoscopy: An application snapshot. *Medica Mundi*, 43(1):17–23, 1999.

[RvSH02] P. Rogalla, J. Terwisscha van Scheltinga, and B. Hamm. Springer, 2002.

[SAS+02] R. M. Summers, N. R. Aggarwal, M. C. Sneller, M. J. Cowan, B. J. Wood, and C. A. Langford. CT virtual bronchoscopy of the central airways in wegener's granulomatosis. *Chest 2002*, 121:242–250, 2002.

[SBD+00]   G. Székely, Ch. Brechbühler, J. Dual, R. Enzler, J. Hug, R. Hutter, N. Ironmonger, M. Kauer, V. Meier, P. Niederer, A. Rhomberg, P. Schmid, G. Schweitzer, M. Thaler, V. Vuskovic, and G. Tröster. Virtual reality based surgery simulation for endoscopic surgery. *Presence: Teleoperators and Virtual Environments*, 9(3):310–333, 2000.

[Sch97]   W. J. Schroeder. A topology modifying progressive decimation algorithm. In *Proc. of IEEE Visualization'97*, pages 205–212, 1997.

[Sch05]   H. Scharsach. Advanced GPU raycasting. In *Proc. of Central European Seminar on Computer Graphics (CESCG '05)*, 2005.

[SFB98]   T. Sabisch, A. Ferguson, and H. Bolouri. Automatic registration of complex images using a self organizing neural system. In *Proc. of IEEE International Joint Conference on Neural Networks*, 1998.

[SH95]   B. T. Stander and J. C. Hart. A Lipschitz method for accelerated volume rendering. In *Proc. of IEEE Visualization '95*, pages 107–114, 1995.

[Sha96]   R. Shahidi. Surface rendering versus volume rendering in medical imaging: Techniques and applications. In *Proc. of IEEE Visualization 96*, pages 439–440, 1996.

[SHE+98]   N. Suzuki, A. Hattori, T. Ezumi, A. Uchiyama, T. Kumano, A. Ikemoto, Y. Adachi, and A. Takatsu. Simulator for virtual surgery using deformable organ models and force feedback system. *Stud Health Technol Inform*, 50:227–233, 1998.

[SHK+02]   A. Schreyer, H. Herfarth, R. Kikinis, J. Seitz, J. Schlmerich, A. Geissler, and S. Feuerbach. 3D modelling and virtual endoscopy of the small bowel based on magnetic resonance imaging in patients with inflammatory bowel disease. *Invest Radiol*, (37):528–533, 2002.

[SHLJ96]   H. Shen, C. D. Hansen, Y. Livnat, and C. R. Johnson. Isosurfacing in span space with utmost efficiency. In *Proc. of IEEE Visualization '96*, pages 287–294, 1996.

[SJP+01]   R. M. Summers, C. D. Johnson, L. M. Pusanik, J. D. Malley, A. M. Youssef, and J. E. Reed. Automated polyp detection at CT colonography: feasible assessment in a human population. *Radiology 2001*, 219:51–59, 2001.

[SK98]   M. Sramek and A. Kaufman. Object voxelization by filtering. In *Proc. of IEEE Symposium on Volume Visualization*, pages 111–118, 1998.

[SKH+98]   Y. Shigematsu, Y. Korogi, T. Hirai, T. Okuda, I. Ikushima, T. Sugahara, L. Liang, Y. Ge, and M. Takahashi. Virtual MRI endoscopy of the intracranial cerebrospinal fluid spaces. *Neuroradiology*, 40(10):644–650, 1998.

[SKL+03]   D. Siablis, G. C. Kagadis, E. N. Liatsikos, C. Kalogeropoulou, T. Petsas, D. Karnabatidis, T. Voudoukis, G. C. Sakelaropoulos, G. Nikiforidis, and

G. A. Barbalias. Ureteral metallic stents: application of virtual endoscopy for ureteral patency control. *Int Urol Nephrol*, 35(3):327–330, 2003.

[SPMH99] R. M. Summers, L. M. Pusanik, J. D. Malley, and J. M. Hoeg. 3D analysis of the peritalar complex using MR imaging in live patients. In *Proc. of SPIE Medical Imaging 99*, pages 258–269, May 1999.

[SR01] M. Sharghi and I. W. Ricketts. A novel method for accelerating the visualisation process used in virtual colonoscopy. In *Proc. of IV 2001*, pages 167–172, 2001.

[SR02a] M. Sharghi and I. W. Ricketts. Accelerating visualisation in virtual colonoscopy. In *Proc. of IEEE EMBS UK & RI Postgraduate Conference on Biomedical Engineering and Medical Physics*, July 2002.

[SR02b] M. Sharghi and I. W. Ricketts. Interactive visualisation of a virtual colonoscopy by combining ray casting with an acceleration corridor. In *Proc. of 6th Annual meeting on Medical Image Understanding and Analysis*, July 2002.

[Sra94] M. Sramek. Gray level voxelization: A tool for simultaneous rendering of scanned and analytical data. In *Proc. of Spring School on Computer Graphics and its Applications*, pages 159–168, 1994.

[STO+03] N. Sanno, A. Teramoto, R. Y. Osamura, E. Horvath, K. Kovacs, R. V. Lloyd, and B. W. Scheithauer. Pathology of pituitary tumors. *Neurosurg Clin N Am*, 14(1):25–39, 2003.

[SVvG+01] I. W. O. Serlie, F. M. Vos, R. E. van Gelder, J. Stoker, R. Truyen, F. A. Gerritsen, Y. Nio, and F. H. Post. Improved visualization in virtual colonoscopy using image-based rendering. In *Data Visualization 2001, Proceedings Joint Eurographics - IEEE TCVG Symposium on Visualization 2001*, pages 137–146, 2001.

[SWB+02] E. Sorantin, G. Werkgartner, E. Balogh, A. Vilanova i Bartroli, K. Palágyi, L. G. Nyúl, and L. Ruskó. Virtual dissection and automated poly detection of the colon based on spiral CT - techniques and preliminary experience on a cadaveric phantom. *Eur. Surg.*, 34(2):143–151, 2002.

[SWM+03] P. Schultza, M. Wiorowskia, P. Milliona, F. Veillonb, and C. Debrya. Contribution of virtual endoscopy to the investigation of laryngotracheal pathological conditions. *ORL*, 65:33–38, 2003.

[SZL92] W. J. Schroeder, J. A. Zarge, and W. E. Lorensen. Decimation of triangle meshes. In *Proc. of ACM Siggraph*, pages 65–70, 1992.

[THK+00] S. Takebayashi, M. Hosaka, Y. Kubota, K. Noguchi, M. Fukuda, Y. Ishibashi, T. Tomoda, and S. Matsubara. Computerized tomographic ureteroscopy for diagnosing ureteral tumors. *J Urol. 2000*, 163(1):42–46, 2000.

[THMG04]   M. Teschner, B. Heidelberger, M. Mller, and M. H. Gross. A versatile and robust model for geometrically complex deformable solids. In *Proc. of Computer Graphics International (CGI'04)*, pages 312–319, 2004.

[THT$^+$99]   S. Takebayashi, M. Hosaka, K. Takase, N. Kubota, T. Kishida, and S. Matsubara. Computerized tomography nephroscopic images of renal pelvic carcinoma. *J Urol. 1999*, 162(2):315–318, 1999.

[TPBF87]   D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. Elastically deformable models. In *Proc. of International Conference on Computer Graphics and Interactive Techniques*, pages 205–214, 1987.

[TPK$^+$00]   T. Talala, T. Pirila, V. Karhula, E. Ilkko, and I. Suramo. Preoperative virtual endoscopy and three-dimensional imaging of the surface landmarks of the internal carotid arteries in trans-sphenoidal pituitary surgery. *Acta Otolaryngol*, 120:783–787, 2000.

[TvSGSH02]   U. Tiede, N. von Sternberg-Gospos, P. Steiner, and K. H. Höhne. Virtual endoscopy using spherical quick time-VR panorama views. In *Proc. of MICCAI*, pages 186–192, 2002.

[VG94]   D. J. Vining and D. W. Gelfand. Noninvasive colonoscopy using helical CT scanning, 3D reconstruction. *SGR Scientific Program 1994*, 70, 1994.

[VGK00]   A. Vilanova i Bartrolí, E. Gröller, and A. König. Cylindrical approximation of tubular organs for virtual endoscopy. In *Proc. of Computer Graphics and Imaging 2000*, pages 283–289, November 2000.

[Vil01]   A. Vilanova i Bartrolí. *Visualization Techniques for Virtual Endoscopy*. PhD thesis, Vienna University of Technology, 2001.

[VKG99]   A. Vilanova i Bartrolí, A. König, and E. Gröller. Viren: Virtual endoscopy system. *Machine Graphics & Vision*, 8(3):469–487, 1999.

[vRLHJ$^+$04]   B. von Rymon-Lipinski, N. Hanssen, T. Jansen, L. Ritter, and E. Keeve. Efficient point-based isosurface exploration using the span-triangle. In *Proc. of IEEE Visualization '04*, pages 441–448, 2004.

[VSA$^+$97]   D. Vining, D. Stelts, D. Ahn, P. Hemler, Y. Ge, G. Hunt, C. Siege, D. McCorquodale, M. Sarojak, and G. Ferretti. Freeflight: A virtual endoscopy system. In *Proc. of First Joint Conference, Computer Vision, Virtual Reality and Robotics in Medicine and Medical Robotics and Computer-Assisted Surgery*, volume LNCS 1205, pages 413–416, 1997.

[VSvG$^+$01]   F. Vos, I. W. O. Serlie, R. E. van Gelder, F. H. Post, R. Truyen, F. A. Gerritsen, J. Stoker, and A. M. Vossepoel. A new visualization method for virtual colonoscopy. In *Proc. of MICCAI 2001*, pages 645–654, 2001.

[VW99]   P. Viola and W. M. Wells. Alignment by maximization of mutual information. In *Proc. of 5th Int. Conf. Computer Vision*, pages 16–23, 1999.

[VWK+01]   A. Vilanova i Bartrolí, R. Wegenkittl, A. König, E. Gröller, and E. Sorantin. Virtual colon flattening. In *Data Visualization (Proc. of Symposium on Visualization)*, pages 127–136, April 2001.

[VWKG01a] A. Vilanova i Bartrolí, R. Wegenkittl, A. König, and E. Gröller. Nonlinear virtual colon unfolding. In *Proc. of IEEE Visualization '01*, 2001.

[VWKG01b] A. Vilanova i Bartrolí, R. Wegenkittl, A. König, and E. Gröller. Perspective projection through parallely projected slabs for virtual endoscopy. In *Proc. of SCCG '01- Spring Conference on Computer Graphics*, pages 287–295, April 2001.

[WCH92]   J. Weng, P. Cohen, and M. Herniou. Camera calibration with distortion models and accuracy evaluation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(10):965–980, October 1992.

[WDK+01] M. Wan, F. Dachille, K. Kreeger, S. Lakare, A. E. Kaufman, M. Wax, and J. Z. Liang. Interactive electronic biopsy for 3D virtual colonscopy. In *Proc. of SPIE Medical Imaging*, pages 483–488, February 2001.

[Wes90]   L. Westover. Footprint evaluation for volume rendering. In *Proc. of SIGGRAPH '90*, pages 367–376, 1990.

[WFD+04] S. Wolfsberger, M.-T. Forster, M. Donat, A. Neubauer, K. Bühler, J. Hainfellner, and E. Knosp. Virtual endoscopy is a useful device for training and preoperative planning of transsphenoidal endoscopic pituitary surgery. *Minimally Invasive Surgery*, (47):214–220, 2004.

[WG90]   J. Wilhelms and A. Van Gelder. Octrees for faster isosurface generation (extended abstract). *Computer Graphics*, 24(5):57–62, November 1990.

[WHM+02] P. Walshe, S. Hamilton, D. McShane, R. McConn Walsh, M.A. Walsh, and C. Timon. The potential of virtual laryngoscopy in the assessment of vocal cord lesions. *Clinical Otolaryngology & Allied Sciences*, 27(2), April 2002.

[WHVP92] T. Van Walsum, A. J. Hin, J. Versloot, and F. H. Post. Efficient hybrid rendering of volume data and polygons. In F. H. Post and A. J. Hin, editors, *Advances in Scientific Visualization*, pages 83–96. Springer, 1992.

[WMC93]   R. P. Woods, J. C. Mazziotta, and S. R. Cherry. MRI-PET registration with automated algorithm. *Computer Assisted Tomography*, 17(4):536–546, 1993.

[WOS04]   A. Wahle, M. E. Olszewski, and M. Sonka. Interactive virtual endoscopy in coronary arteries based on multimodality fusion. *IEEE Transactions on Medical Imaging*, 23(11):1391–1403, November 2004.

[WR02]   B. J. Wood and P. Razavi. Virtual endoscopy: A promising new technology. *Americal Family Physician*, 66(1):107–112, July 2002.

[WS01]   R. Westermann and B. Sevenich. Accelerated volume ray-casting using texture mapping. In *Proc. of IEEE Visualization 2001*, pages 271–278, 2001.

[WTK+99]   M. Wan, Q. Tang, A. Kaufman, Z. Liang, and M. Wax. Volume rendering based interactive navigation within the human colon. In *Proc. of IEEE Visualization' 99*, pages 397–400, 1999.

[WVH+00]   R. Wegenkittl, A. Vilanova, B. Hegedüs, D. Wagner, M. Freund, and E. Gröller. Mastering interactive virtual bronchioscopy on a low-end PC. In *Proc. of IEEE Visualization*, pages 461–465, 2000.

[XESV97]   J. Xia, J. El-Sana, and A. Varshney. Adaptive real-time level-of-detail-based rendering for polygonal models. *IEEE Transactions on Computer Graphics and Visualization (TVCG)*, 3(2):171–183, June 1997.

[YHJ+97]   S. You, L. Hong, K. Junyaprasert, A. Kaufman, S. Muraki, Y.Zhou, M. Wax, and Z. Liang. Interactive volume rendering for virtual colonoscopy. In *Proc. of IEEE Visualization '97*, pages 343–346, 1997.

[YMFS04]   J. Yao, M. Miller, M. Franaszek, and R. M. Summers. Colonic polyp segmentation in CT colonography-based on fuzzy clustering and deformable models. *IEEE Transactions on Medical Imaging*, 23(11):1344–1352, November 2004.

[YS93]   R. Yagel and Z. Shi. Accelerating volume animation by space leaping. In *Proc. of IEEE Visualization '93*, pages 62–69, 1993.

[YSW+96]   R. Yagel, D. Stredney, G. J. Wiet, P. Schmalbrock, L. Rosenberg, D. J. Sessanna, and Y. Kurzion. Building a virtual environment for endoscopic sinus surgery simulation. *Computers and Graphics*, 20(6):813–823, December 1996.

[ZKV92]   K. J. Zuiderveld, A. H. J. Koning, and M. A. Viergever. Acceleration of raycasting using 3D distance transforms. In *Visualization in Biomedical Computing II, Proc. SPIE 1808*, pages 324–335, 1992.

[ZT94]   O. Zienkiewicz and R. Taylor. *The Finite Element Method*. McGraw-Hill Book Company, 1994.