DIPLOMARBEIT

Multimodal Visualization of Anatomical and Functional Volume Data of the Human Brain

ausgeführt am Institut für Computergraphik
der Technischen Universität Wien

unter der Anleitung von Ao. Univ.-Prof. Dipl.-Ing. Dr. Eduard Gröller

durch

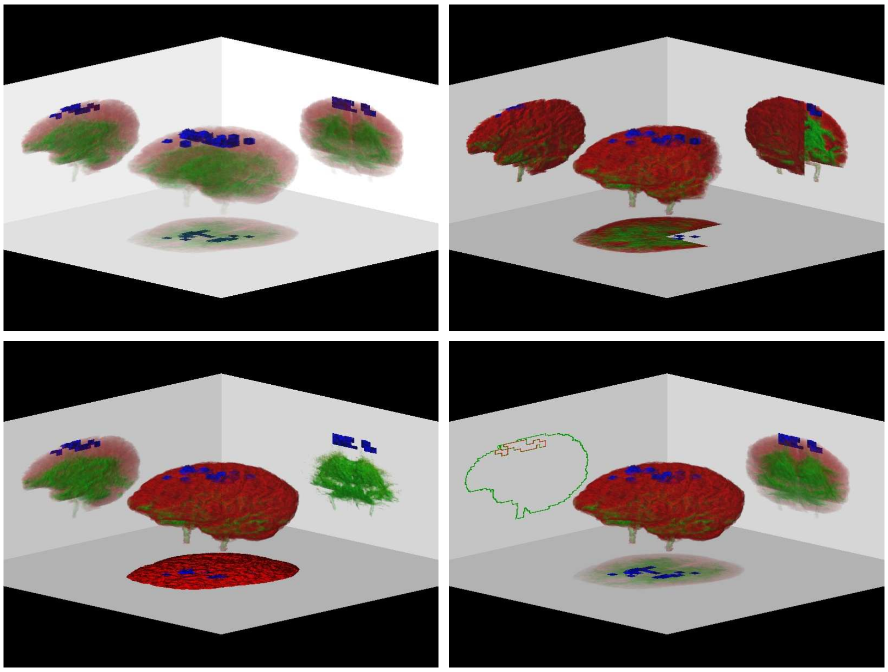
Helmut Doleisch
Aspettenstr. 34/26/2, 2380 Perchtoldsdorf

Wien, 2. Dezember 1998 Helmut Doleisch

Helmut Doleisch

Multimodal Visualization of Anatomical and Functional Volume Data of the Human Brain

(Diploma Thesis)



http://www.cg.tuwien.ac.at/research/vis/vismed/MM/

mailto:helmut@cg.tuwien.ac.at

Abstract

Multimodal visualization of functional and anatomical data of the human brain is an important field in medical volume visualization. The aim of the work presented in this thesis is to provide the user with information on the location of the functional activations in the different regions of the brain. When the approach of direct volume rendering is chosen, the visual impression of details usually suffers from accumulating the image from colors and opacities derived from the data set. After the discussion of the basic direct volume rendering approach, implemented as a multimodal ray-casting approach, three extensions to overcome the problem of such a bed visual impression are presented: *Transfer function volumes* are used for highlighting activated regions. *Multiple views* simplify the task of localizing these parts of the brain. Complementary information is brought into the visualization by the usage of *Magic Mirrors* in order to enhance the comprehensive view of the multimodal arrangement of volume-data sets.

Kurzfassung

Multimodale Visualisierung von funktionalen und anatomischen Daten des menschlichen Gehirns ist ein wichtiges Gebiet in der medizinischen Volumenvisualisierung. Das Ziel der in dieser Diplomarbeit vorgestellten Arbeit ist es, den Benutzer mit Information bezüglich der lokalen Lage von funktionalen Aktivitäten in verschiedenen Regionen des Gehirns zu unterstützen. Wenn ein "Direct Volume Rendering"-Ansatz gewählt wird, leidet der klare dreidi-

mensionale Eindruck meist unter der Aufsummierung von Farb- und Transparenzwerten aus dem Datensatz. Nach einer Vorstellung eines grundlegenden Verfahrens des "Direct Volume Rendering", implementiert in Form eines multimodalen "Raycastings", werden drei Erweiterungen präsentiert, um dieses Problem des schlechten dreidimensionalen Eindrucks bei Verwendung eines "Direct Volume Rendering"-Ansatzes zu lösen: "*Transfer function volumes*" werden verwendet, um aktivierte Regionen zu markieren. *Multiple views* vereinfachen die Lokalisierung dieser Regionen des Gehirns. Zusätzliche Information zur Visualisierung wird durch die Verwendung von *Magic Mirrors* eingebracht, um die Darstellung der multimodalen Anordnung der Volumendatensätze noch zu verbessern.

Contents

Glossary

CT	(X-ray) Computed Tomography
EEG	Electro EncephaloGraphy
FID	free-induction decay (responce signal)
FLASH	Fast Low Angle SHot
fMRI	functional Magnetic Resonance Imaging
MDEFT	Modified Driven Equilibrium Fourier Transform
MEG	Magneto EncephaloGraphy
MRI	Magnetic Resonance Imaging
NMR	Nuclear Magnetic Resonance
PD	proton density
PET	Positron Emission Tomography
RF pulse	radio-frequency pulse
SNR	signal to noise ratio
SPECT	Single Photon Emission Computed Tomography
TE	echo delay time (in MR scanning)
TR	repetition time (in MR scanning)

Chapter 1

Introduction and discussion of problems

Visualization is a method of extracting meaningful information from complex data sets through the use of interactive graphics and imaging. It provides processes for seeing and steering the unseen, thereby enriching existing scientific methods [29].

Visualization in the medical field is established for a rather long time. As the computing power of newly developed computers increases, and research both in the medical field as well as in the field of computer graphics progresses, new visualization approaches and methods are needed.

A variety of different visualization areas already exists, as for example: vector-field visualization, information visualization, 2D-plots (business graphics), etc.

One special subfield of visualization is volume visualization. Volume visualization is concerned with the representation, manipulation, interpretation and rendering of complex volumetric data. In medical volume visualization, this volumetric data is usually obtained by scanning the patient. For this purpose many different methods can be applied, each of them having its own characteristics applied to the measured data sets.

1.1 Why is volume visualization used in a medical context?

Since the days, when techniques like CT or MRI have been invented and constantly enhanced, doctors at hospitals and in other medical areas have used these techniques for diagnosis and therapy planning.

Visually representing the obtained data in the best way possible has been a problem from the very beginning. The measured data was visualized slice by slice as grey-scale images printed on paper or slides. The production of such an image took a rather long time in those days.

Nowadays, due to the increased calculating capacity of modern computers, these images can be calculated, visualized and manipulated interactively, which in fact is, what users (in the medical field the doctors) expect, to be able to work more efficiently and conveniently with the obtained data sets.

But users today are not that satisfied with the use of only twodimensional slices or cuts anymore. They would like to have a more realistic and more real-world-like representation, which in fact is a threedimensional visualization of the measured data, as this data is threedimensional in reality, too. This would help to more clearly see and imagine the structures and spatial relationships of object features. 3D visualization techniques (volume-visualization techniques) are capable of reconstructing the volumes given by successive slice images. They even allow a full understanding of difficult situations with complex topology [12][20]. However, volume visualization techniques are only applicable in clinical routine, if they provide the necessary ease of interpretation. Therefore, any change of visualization parameters must be achieved interactively and intuitively [60], and detail information within the slice and a global overview must be accessible in a combined image representation [19].

1.1.1 Why is multimodal volume visualization used in a medical context?

Over the past decade, the use of digital imaging modalities for clinical diagnosis and therapy planning has dramatically increased. New developments in threedimensional volume acquisitions such as spiral CT (*Computed Tomography*) scanning, faster and better pulse sequences for MRI (*Magnetic Resonance Imaging*) and improved functional imaging provided by modalities such as PET (*Positron Emission Tomography*), SPECT (*Single Photon Emission Computed Tomography*), EEG (*Electro EncephaloGraphy*), MEG (*Magneto EncephaloGraphy*), and fMRI (*functional MRI*) are creating a rapidly increasing demand for integrated multimodality 3-D visualization. It aims at efficiently presenting information from multiple modalities, in order to obtain a comprehensive picture of clinical cases. Therefore, depending on the clinical requirements, it might often be not sufficient to consider vascular, anatomical and functional information separately but to superimpose images of different modalities.

1.2 Multimodal data

Multimodal data, especially when regarding multimodal volume data, can be of different kinds. Multimodal volumetric data in the medical field can be of one of the following types:

- data of the same object obtained by different measurement methods
- data of the same object obtained by different measurements with the same method
- data of different objects obtained by either different or similar measurement methods

In the first case, we can distinguish two major types of multimodality visualization: the combination of (i) anatomical data sets from different modalities, and (ii) the combination of functional with anatomical data [59]. While integration of multimodal anatomical information is often used (e.g. for radiation treatment planning), only the problem of volume visualization of functional information within an anatomical context is addressed in this work. The functional information (data) in the special case of visualizing multimodal data of the human brain measures the activation potential of the brain at a certain point in time. Therefore the characteristic of the measured data is timevariant.

A simple example of this would be the following: A patient has to tip thumb and index fingers at each other. The activation potential of the brain is measured. This gives the functional context of the reaction to the action of his fingers. This data is measured repeatedly for example every second, during the whole action that the patient performs.

The functional data has to be registered to the anatomical reference data of the patients brain. Visualization is used to identify the regions of the brain, where the activation potential changes.

The second and third cases in the list above will not be addressed in this thesis.

1.2.1 Different modalities and their characteristics

A variety of imaging modalities give insight to different feature spaces and kinds of information in current medical practice [41]. Conventional radiography is one of the most wide-spread and established modalities. However, it only provides twodimensional (2D) projection images with limited depth information.

More recently, other techniques gained more attention since they provide threedimensional (3D) information with a great diversity of structural, anatomical and functional details allowing more comprehensive examinations. As a common feature all these modalities produce stacks of equidistant and spatially assignable slice images forming a volume data set. The sample points are usually arranged on a uniform grid.

Using CT (*X-ray Computed Tomography*) scanning the 3D distribution of X-ray densities is calculated by reconstructing the X-ray transmission measurements from a variety of different viewing angles.

Completely different images with a wide variety of characteristics are obtained with MRI (*Magnetic Resonance Imaging*) scanning. Soft tissue is represented very well with highly detailed structural information. Using a locally variable static magnetic field and long-wave radio frequencies resonance absorption of magnetic nuclei in biological tissue is measured in order to calculate the spatial distribution of the nuclear magnetization. Developments reducing measurement time and investigating the potential of flow (MRA: *Magnetic Resonance Angiography*) and functional imaging (fMRI: *functional Magnetic Resonance Imaging*) will further increase the importance of MRI in the future.

Finally, with modalities of nuclear medicine such as PET (*Positron Emission Tomography*) and SPECT (*Single Photon Emission Computed Tomography*) the distribution of radio-isotopes in specific organs or structures is measured giving information on metabolism and functionality.

As these data sets gained from different imaging modalities are very often (at least to some extend) complimentary, one should think about combining them. For example, MR images are the best for delineating soft-tissue anatomy. CT images are more suitable for depicting bones and calcifications. On the other hand, functional imaging modalities, such as for example fMRI of PET, provide the user with functional information of the scanned parts of the human brain. Therefore, it is desirable to combine these different data sets into one single threedimensional display, so that these different types of information can be visualized and processed simultaneously [23].

The above mentioned imaging modalities are just a small part of modern imaging modalities, that are in use or available. They are the most commonly used for the special application this thesis is dealing with, i.e., multimodal data of the human brain. For a more complete description and explanation of other medical imaging modalities, see for example [36].

1.3 Problems in multimodal volume visualization

In multimodal volume visualization all the problems, that have to be taken into consideration for general volume visualization, are also of concern. Furthermore some even more difficult problems arise.

In the case of multimodal volume visualization some of the major additional problems arising are the following:

- the data sets might have different resolutions
- the spatial extend of the data sets can be different
- the orientation of the data sets can be different
- the data sets might have different distortions due to the use of different measurement methods and different accuracies of the measurements (different data ranges)
- volumetric data sets are usually very large, processing two volumetric data sets simultaneously is even more memory demanding

In the special case of anatomical-functional multimodal volume visualization of the human brain, some of these problems are even more complex than they are for anatomical-anatomical multimodal volume visualization, for example when using CT and MRI.

For example the resolution of the functional data sets usually is four times lower than that of the corresponding anatomical data set, as the data has to be measured interactively (timeseries are measured). Faster measurement is currently not possible. Also the functional data set usually covers only a small part of the spatial region of the anatomical data set. For example for the human brain only a small layer of 5-7 slices covering 2-3 cms in height is taken.

Some other problems arising in this special field of application will be discussed later.

1.4 Aim of this work

In this work volume visualization methods for multimodal data of the human brain are developed, consisting of functional data of the human brain (measured mainly by fMRI) and anatomical data of the brain (measured by different methods and modalities, such as for example different methods of MRI).

In the next chapter the theoretical background in MRI as well as in functional MRI will be dealt with. Topics like image characteristics and image artifacts in MR imaging, as well as the problem of data acquisition will be discussed.

In Chapter 3 the state of the art in general volume visualization, as well as in medical volume visualization, and especially in multimodal volume visualization will be discussed. Special issues like segmentation, registration, and the corresponding problems as well as currently available approaches and solutions to these problems are addressed.

After this state of the art review in volume visualization and multimodality issues, in Chapter 4 an overview and a short description of some of the most commonly used volume-visualization software-systems is given, with special emphasis on the usability in the case of multimodal volume visualization.

In Chapter 5 a description of four different visualization techniques that were developed during the work on this diploma thesis is given. In Chapter 6 the implementation details are provided.

After results are illustrated in Chapter 7, conclusions are drawn in Chapter 8. A short version of this whole work can be found in Chapter 9, followed by some acknowledgments in Chapter 10.

Chapter 2

Theoretical background in MRI

In this chapter, first of all an introduction to MRI and fMRI will be given, explaining the most important characteristics of these measurement methods. After this introduction, image characteristics and image artifacts resulting from different parameter settings of the measurements will be discussed briefly. Afterwards data acquisition will be dealt with.

2.1 Basic principles of MRI

Magnetic Resonance Imaging (MRI) is an imaging technique used primarily in medical settings to produce high quality images of the inner regions of the human body. MRI is based on the principles of nuclear magnetic resonance (NMR), a spectroscopic technique used by scientists to obtain microscopic, chemical and physical information about molecules.

The human body primarily consists of fat and water. Fat and water are based on hydrogen atoms where the human body consists of approximately 63% hydrogen atoms. Hydrogen nuclei have an NMR signal. For these reasons magnetic resonance imaging primarily measures the NMR signal from the hydrogen nuclei.

The basis of MRI is the directed magnetic field or moment, associated with charged particles in motion. Nuclei containing an odd number of protons and/or neutrons have a characteristic motion called precession. Because nuclei are charged particles, this precession produces a small, but measurable magnetic moment.

To obtain an MR image of an object, the object is placed in a uniform magnetic field. As a result, hydrogen nuclei of the object align with the magnetic field and create a net magnetic moment parallel to the magnetic field (see Figure 2.1).

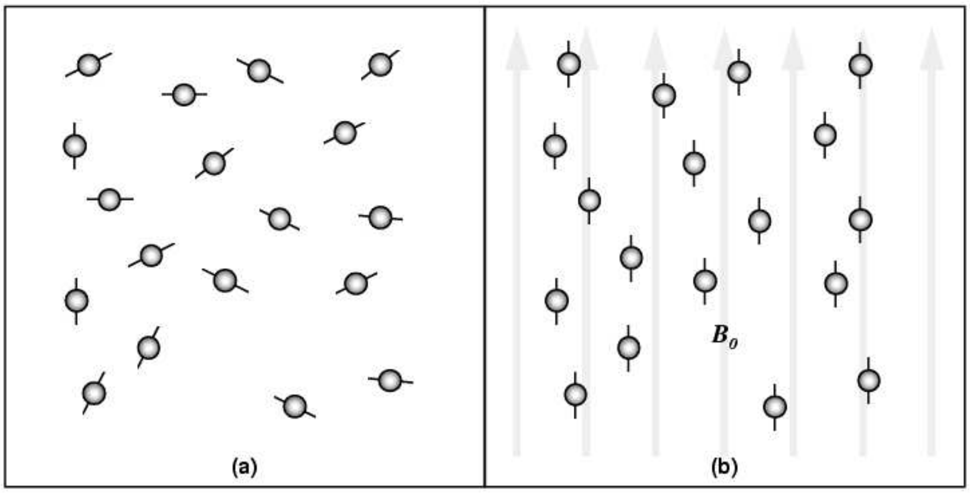


Figure 2.1: In the absence of a strong magnetic field the hydrogen nuclei are randomly aligned (a). When a strong magnetic field (B_0) is applied, the hydrogen nuclei align in the direction of the field (b) (see [46]).

Then a radio-frequency (RF) pulse is applied perpendicular to the magnetic field. This pulse causes the magnetic moment to tilt away from the magnetic field (see Figure 2.2 (a)). The angle of rotation of the net magnetization is called the "flip" or "tip" angle. Pulses with a flip angle less than 90 degrees are especially used in very fast scanning techniques (e.g. FLASH, FISP). As soon as the RF signal is removed, the nuclei realign themselves such that their magnetic moment is again parallel to the magnetic field. This return to the equilibrium is referred to as relaxation. During relaxation the nuclei lose energy by emitting their own RF signal (see Figure 2.2 (b)). This

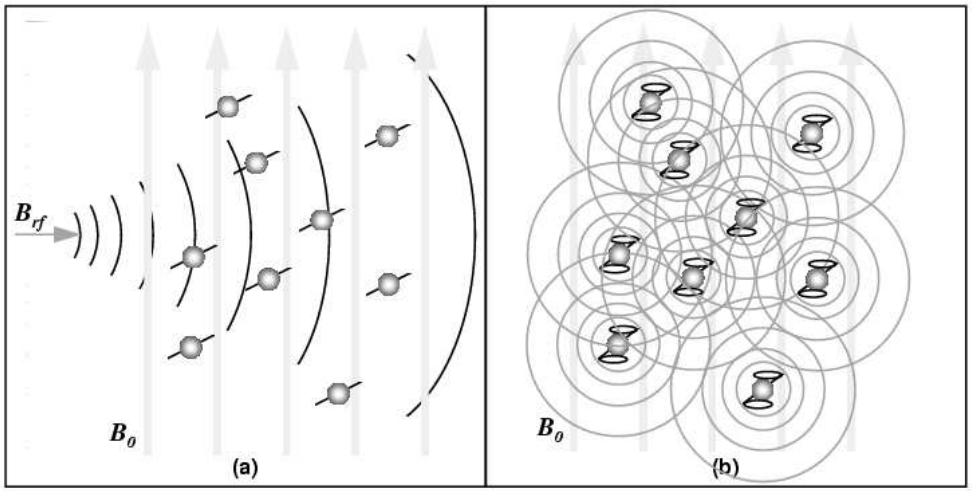


Figure 2.2: The RF pulse (B_{rf}) causes the net magnetic moment of the nuclei to tilt away from B_0 (a). When the RF pulse stops, the nuclei return to equilibrium such that the net magnetic moment is again parallel to B_0. During this realignment the nuclei loose energy and emit a measurable RF signal (b) (see [46]).

response signal (also referred to as the free-induction decay or FID response signal) is measured by a conductive field coil placed around the object being imaged. This measurement is processed and reconstructed to obtain grey-scale MR images [55]. The strength of the response signal depends on the tissue type.

Magnetic resonance is characterized by different forms of relaxation:

1. T1 relaxation, also referred to as *longitudinal* or *spin-lattice* relaxation

2. T2 relaxation, also referred to as *transverse* or *spin-spin* relaxation

3. T2*relaxation

T1 measures the time required for the magnetic moment of the displaced nuclei to return to equilibrium (i.e., realign themselves with the magnetic field B_0). T2 indicates the time required for the FID response signal from a given tissue type to decay. Two factors contribute to the decay of transverse magnetization:

- molecular interactions
- variations in B_0 caused by inhomogeneities

The combination of these two factors results in the decay of transverse magnetization. The combined time constant is called $T2^*$relaxation. $T2^*$is characterized by a decay of transverse magnetization at a rate greater than T2 caused by a loss of signal seen with dephasing of individual magnetizations.

The signal intensity which characterizes a given tissue type (e.g. white brain matter, cerebrospinal fluid (CSF), ...) depends on the proton density (PD) of the tissue. The higher the proton density is, the stronger is the FID response signal and the larger are the T1 and T2 relaxation times. In pure water the T1 and T2 times are approximately the same (2-3 seconds). In biological materials the T2 relaxation time is considerably shorter than the T1 time. For example, for white brain matter, T1=0.5 seconds and T2=0.07 seconds.

During the scanning process the RF pulse, B_{rf}, is repeated at a predefined rate. The period of the RF pulse sequence is the *repetition time*, TR. The FID response signals can be measured at various times within the TR interval. The time between the moment the RF pulse is applied and the response signal is measured is the *echo delay time* TE. A short repetition time TR and a short echo time TE results in a signal, which is mainly determined by T1 and therefore is called a *T1-weighted* signal. A longer repetition time TR and a longer echo time TE result in a *T2-weighted* signal (see Table 2.1).

There are a number of different pulse sequences available for MRI. One of the most commonly used is the *spin echo pulse sequence*. A detailed description of this sequence can be found in the literature [55]. The pulse sequence timing can be adjusted to give T1-weighted, proton or spin density, or T2- and $T2^*$-weighted images (see Figure 2.3 for images). Dual-echo and

	TR short (500-700ms)	**TR long** (2-2.5 sec)
TE long (70-120ms)	—	T2-weighted
TE short (15-15ms)	T1-weighted	PD-weighted

Table 2.1: Relation between the sequence parameters (TE, TR) and the signal weighting

multi-echo sequences can be used to obtain both, proton density and T2-weighted images, simultaneously.

A detailed review of the physics of MRI is outside of the scope of this manuscript. For a more complete description of MRI specialized literature ([6], [52]) should be consulted.

As another important point MRI technology uses low frequencies with low energy. MRI does not have dangerous side effects like the X-ray radiation of Computed Tomography. This is also true for fMRI, which is also based on nuclear magnetic resonance (NMR).

2.1.1 Functional MRI

The main advantage of functional magnetic resonance imaging (fMRI) is the possibility to show in a non-invasive way which region of the human brain is activated by a certain stimulus without the usage of endogenous contrast

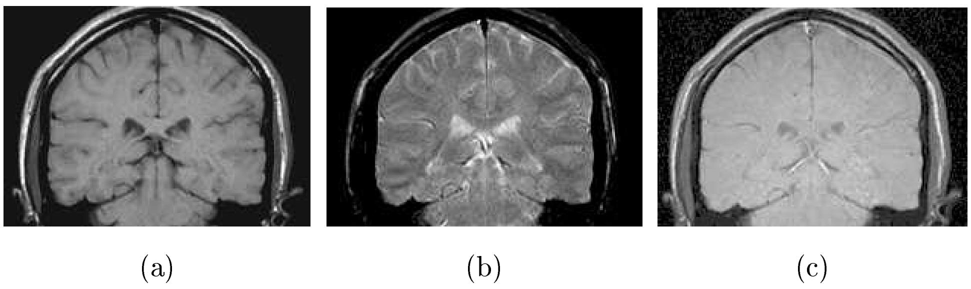


Figure 2.3: Example MR images with three different weightings: (a): T1 weighting, (b): T2 weighting, and (c): PD weighting.

agents.

Until now, no evidence has been found for irreversible changes in human tissue due to MRI, if certain security standards are met. Static magnetic fields of the magnitude of about 1 to 3 Tesla, magnetic gradients between 0.01 and 1 Tesla/m and (electro)magnetic waves with frequencies of radio-waves (42.5 MHz/Tesla for protons) are not dangerous to the human body.

The physiological processes responsible for the signal change during an fMRI experiment are known in principle. A neuronally active cortical region needs more oxygen and glucose than the surrounding tissue. After a particular stimulus has been applied during a certain experiment (for example tipping of thumb and index fingers against each other), more blood has to be transported to the corresponding cortical region as the lack of biochemical substances during neuronal activation has to be compensated. These biochemical processes have the following effects on cerebral hemodynamics:

- blood flow increases (blood volume and blood velocity rises)
- blood oxygenation level changes (also referred to as "*BOLD*" effect)

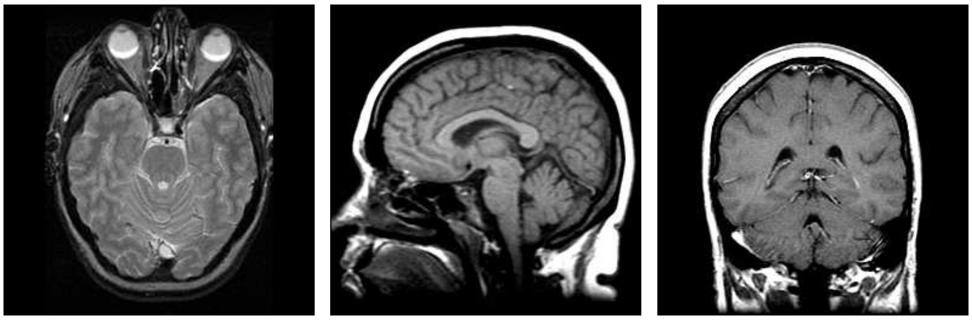
These changes can be detected with fMRI through applications of measurement sequences, which are sensitive to moving spins and/or changes in local magnetic field inhomogeneities.

After a particular stimulus has been applied during a certain experiment, about 20-30% more blood has to be transported to the corresponding cortical region [14]. These massive changes in local cerebral hemodynamics do not only have the effect that blood flow and velocity increases, but result also in changes of the blood oxygenation level as oxygen consumption increases by about 5% [14]. Therefore, more diamagnetic oxyhemoglobin is present in the venous vascular system during the stimulation than in the resting condition. The magnetic inhomogeneity in and around the vessels is reduced and thus causes less dephasing magnetization. As a result, $T2^*$rises and the intensity of the signal sampled at the echo time TE is higher. This is the reason, why the signal intensity increases during stimulation in fMRI experiments.

The images gained from fMRI are scanned during stimulation as well as under resting conditions. By calculating the differences of these images and doing some comparisons the activated regions in the images can be obtained. Functional MRI scans last for approximately 3-5 seconds and are repeated 20-30 times.

2.2 Image Characteristics

Unlike CT where the image appearance is primarily related to the electron density of the material, MR images depend on many different variables, including for example proton density, relaxation effects, flow effects, and others. Image quality is described in terms of signal-to-noise-ratio (SNR, see further down), spatial resolution, and contrast. Unfortunately there is an interdependence between SNR, resolution, and the time of acquisition, meaning that if you want to obtain higher quality images (less SNR, high resolution), the data acquisition times will be longer.



(a) (b) (c)

Figure 2.4: Example MR scans of the three orthogonal acquisition planes: (a): axial, (b): sagittal, and (c): coronal.

MR images are acquired in slices in one of the three orthogonal imaging planes: *axial*, *sagittal* or *coronal* (see 2.4). In a particular slice image, each voxel (see next chapter) has a brightness value that corresponds to a measurement of tissue averaged over a small 3D region.

In addition to the type of tissue at a given location the exact intensities in the resulting image are also influenced by many other features: the neighboring tissues, the radio frequency pulse sequence used to acquire the scan, the quality of the calibration of the scanner, the motion of the tissue or fluid, and many more possible artifacts [55].

2.3 Image Artifacts

An image artifact is any feature appearing in an image which is not present in the original imaged object. An image artifact can either result from an improper operation of the imager, or as a consequence of natural processes or properties of the imaged object, in this case the human body. Some of the most important image artifacts will be described very shortly in the following subsections [55].

2.3.1 Noise

The SNR (signal-to-noise-ratio) determines the graininess of the image appearance. With decreasing value of SNR, the graininess in the image increases. A detailed discussion of the calculation of SNR is given in other manuscripts [55].

2.3.2 RF inhomogeneity

MR images contain unwanted distortions, which can be either spatial distortions or intensity distortions, or both. These variations may result from many different sources, e.g. inhomogeneities in the magnetic field, inhomogeneities in the applied RF pulse sequence, non-uniformity in RF field coil(s) used to measure the FID response signal, etc.

The overall intensity variation is referred to as the RF inhomogeneity. This inhomogeneities can cause different slices to be considerably lighter or darker, but also the intensity gradients within a single slice can vary dramatically. An example can be seen in Figure 2.5.

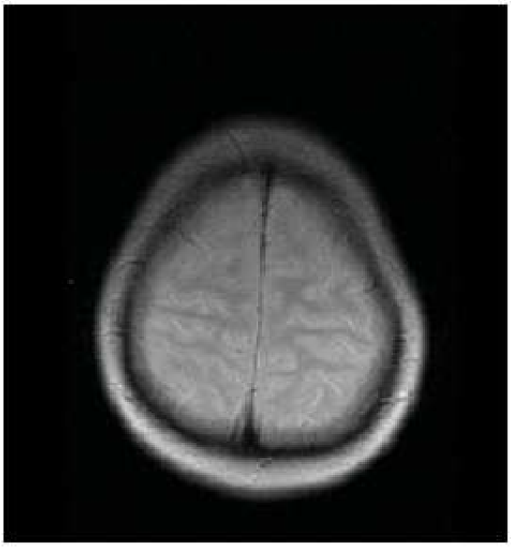
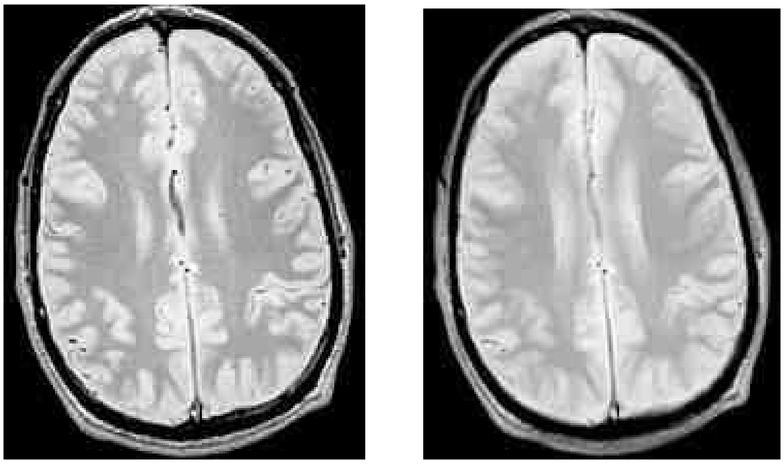


Figure 2.5: An example image showing a considerable intensity gradient. A large intensity variation from top to bottom is present.

2.3.3 Partial volume effect

A partial volume effect is caused by the size of the image voxels. For example, as a small voxel contains either a fat or water signal, a larger voxel may contain a combination of these two. The large voxel's signal intensity is equal to the weighted average of the quantity of fat and water being present in this voxel. This causes a degradation in resolution, as there are multiple features present in a single image voxel. See Figure 2.6 for a comparison of two different images coming from data sets with different slice thickness.



(a) (b)

Figure 2.6: The partial volume effect, illustrated by two images with different slice thickness: 3mm (a) and 10mm (b).

2.3.4 Motion

As the name implies, motion artifacts are caused by motion of the whole or just a part of the scanned object. The motion of the entire object during the imaging sequence generally results in a blurring of the entire image, whereas the movement of a small part of the object results in blurring of the specific part of the image representing the moving feature. The solution to motion artifacts is to immobilize the scanned object (patient in the scope of this thesis). Quite difficult in this context are artifacts caused by heart beating and breathing actions.

2.3.5 Flow

Flow artifacts are caused by flowing blood or other fluids in the body. A liquid flowing through a slice can receive a RF pulse and then leave the region of the slice by the time the signal is recorded.

2.4 Data Acquisition

The data sets in this work were acquired using a 3 Tesla MRI scanner. An image of an MRI scanner is given in Figure 2.7. Principally many different types of MRI scanners exist. They can be divided not only by the strength of the magnetic field used to scan the patient (approximately 0.3-3 Tesla in modern systems), but also according to the comfort they provide to the patients (open vs. closed systems).

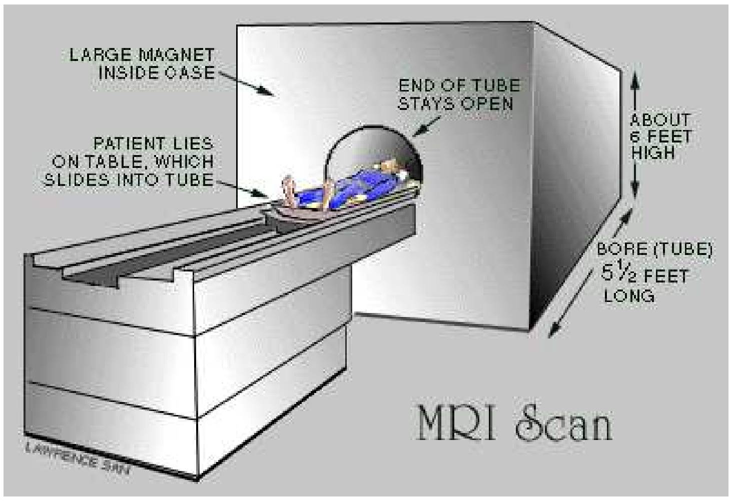
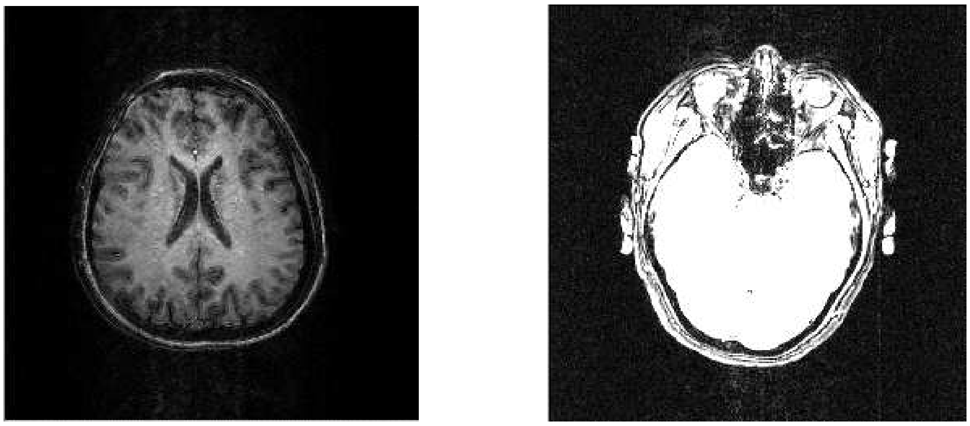


Figure 2.7: An example image of an MRI scanner

The anatomical data sets used for this work were acquired using a pulse sequence called *MDEFT* (Modified Driven Equilibrium Fourier Transform). These images are T1-weighted. There would have also been the possibility to use anatomical data sets generated with other techniques, like for example *FLASH* (Fast Low Angle SHot), a $T2^*$-weighted technique. The MDEFT-data sets seemed to be the most useful for the multimodal volume visualization techniques developed during this work, as they provide more detailed information about the anatomical inner structures of the human brain. For a comparison of sample images obtained with MDEFT and FLASH see Figure 2.8.



(a) (b)

Figure 2.8: Comparison of an image resulting from the MDEFT technique (a), and an image resulting from FLASH (b).

The functional data sets used in this work are T2*-weighted, and have already been pre-processed, so that the regions of activations are set in the volume, and all other regions are neglected. This results in a nearly binary volume for the functional activations, giving hardly any differences in activation potentials, but highlighting the spatial relations of the anatomy, when visualized simultaneously with the anatomical data.

The data sets used for the results shown in Chapter 7 had the following resolutions:

	anatomical data set	**functional data set**
resolution	256 x 256 x 64 slices	64 x 64 x 5 slices
voxel size	1mm x 1mm x 2mm	4mm x 4mm x 4mm

Table 2.2: Resolutions of the data sets used to get the results shown in this thesis

Chapter 3

State of the art

In this chapter the topic of volume visualization of multimodal (anatomical and functional) data of the human brain as a special kind of visualization will be discussed in further detail. All the necessary terminologies as well as different possible concepts, which are applied in this special field of volume visualization, together with the more general cases of conventional volume visualization, will be described.

3.1 General volume visualization

Volume visualization is a field of computer graphics, which is concerned with the representation, manipulation, and rendering of *volumetric data*. After the explanation of this type of data, the subdivision of visualization methods into either *surface rendering methods*, or on the other hand *direct volume rendering methods*, is discussed. Another very important issue in volume visualization, namely *data segmentation*, is discussed.

3.1.1 Volumetric data

Consider an array V of dimension 3 with a resolution of u,v,w in the three dimensions. As this is an extension of the 2D raster graphics data representation into 3D space the elements $v_{i,j,k}$ $(i \in \{0..u-1\}, j \in \{0..v-1\}, k \in$

$\{0..w-1\}$) are called *voxels* analogous to the term *pixel* in 2D space.

Typically voxels are arranged in a regular rectangular grid. Each voxel $v_{i,j,k}$ may contain a data value of arbitrary dimension, e.g. scalar values, vectors or tensors. The most common volume data sets used in medical imaging are created by sampling density values at discrete locations in space by the use of CT or NMR modalities. Therefore in this case scalar data values at grid positions represent density values of the measured data.

The storage requirement for a volume data set with dimension u, v, w is $u * v * w*$sizeof(data value at each grid position). Typical volume data sets used in medical imaging are about of dimensions between $256*256*64$ at the lowest resolutions and $2000*2000*1000$ for the largest data sets, i.e. very large. If scalar data values with a size of one byte are used, the volume data set uses for example 1 Gigabyte of storage for an 1000^3 data set. Therefore compression is an important aspect when dealing with volume data.

In recent years, quite a lot of visualization techniques have been developed for volume data. This techniques can be divided into two classes: *surface rendering methods* and *direct volume rendering methods*.

3.1.2 Surface rendering methods

Surface rendering is an approach of extracting meaningful and intuitive information from volume data sets. This is achieved by extracting regions of the data set with a constant data value. This constant data value is called *isovalue* or *threshold* value. The extracted regions are represented through polygonal surfaces that are called *isosurfaces*. Once these polygonal surfaces have been extracted by any of the methods described in the following, conventional computer graphics techniques are used, to render these surfaces.

Cuberille method

One of the earliest methods that generate surfaces as a representation of the volumetric data. The cuberille method ([7], [21]) is based on an isovalue. In this method the rectilinear faces of all non-transparent voxels (these are voxels with data values greater than the predefined threshold) are used to form a polygonal mesh to represent the isosurface. This polygonal mesh can be displayed using standard surface rendering techniques (for example z-buffering [13]).

Marching cubes

Marching cubes [35] is one of the best known and also the most commonly used surface rendering methods. It creates high quality images by generating a set of triangles that closely approximates the surface of interest (isosurface).

The extraction of polygons from the volumetric data sets for the generation of an isosurface is done in six steps as described below:

1. A cube consisting of eight spatially neighboring voxels (four out of two successive slices of the volume data set) is considered as a logic unit (see Figure 3.1).

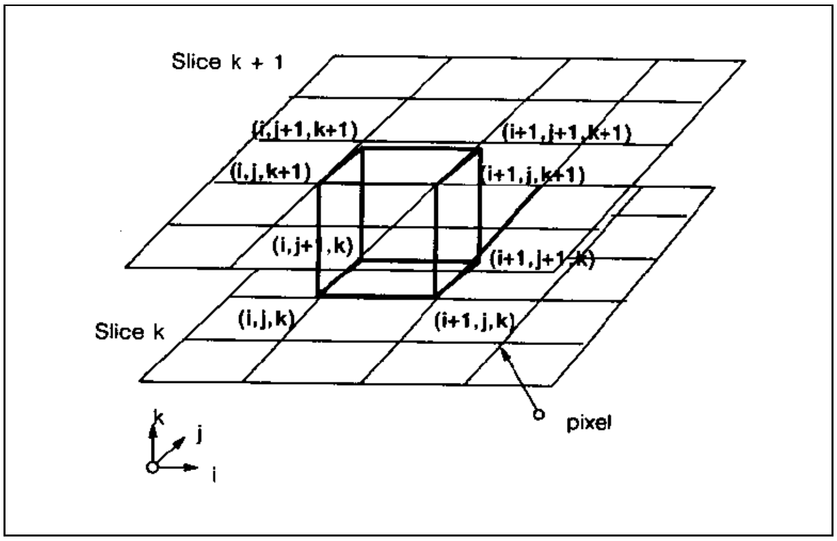


Figure 3.1: Eight neighboring voxels are referred to as a cube [35]

2. For each vertex of the cube the relationship to the isosurface is computed. The vertex is classified as *inside of the isosurface*, if the data

value at this vertex is higher than a given threshold value (isovalue), otherwise it is classified as being located *outside of the isosurface*.

3. The binary classification of all eight vertices generates an *index*. This index consists of eight bits, each bit referring to one of the vertices. If the bit is "0", the corresponding vertex is outside of the isosurface, if the bit is "1", the vertex is inside. In Figure 3.2 the relation between voxel locations and index bits is shown.

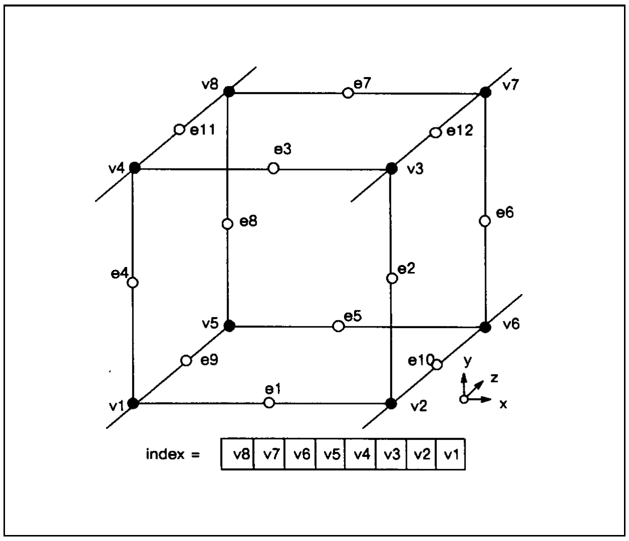


Figure 3.2: Creating the *index* from the classification of the eight vertices of the cube [35]

4. For each index a list of edges of the cube that intersect the isosurface is generated by looking it up in a precomputed table. Basically 256 different indices can be generated by using one bit for each vertex. By considering all the possible rotations and invertations for the cubes, these 256 cases can be reduced to 15 topologically different cases shown in Figure 3.3.

5. For every intersected edge of the cube an intersection point on that edge is calculated by linear interpolation using the data values at the two connected vertices and the threshold value. Always three of these

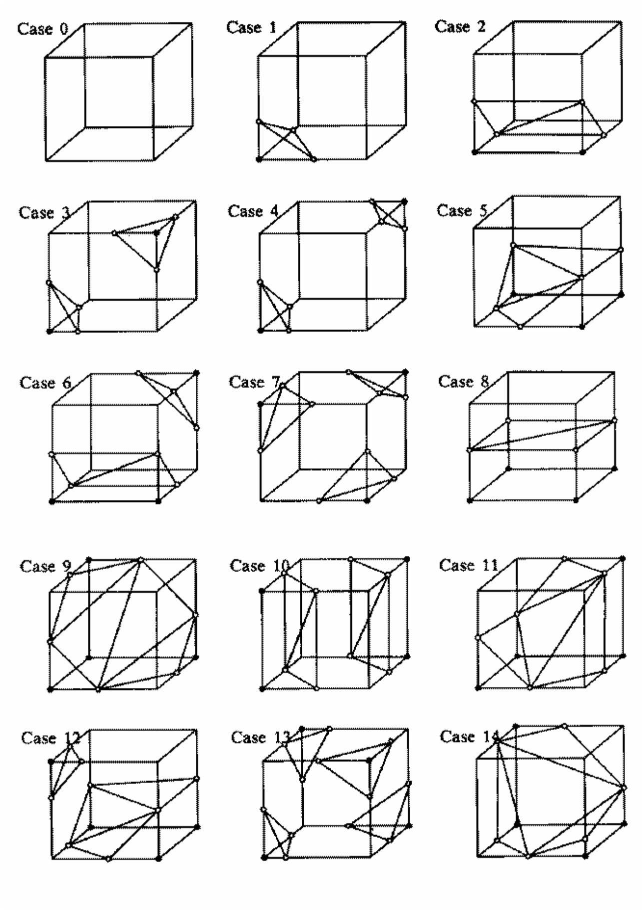


Figure 3.3: The 15 possible cases for intersections of a cube and an isosurface as presented by Cline et al. [10]

intersection points are used in the correct way to build up a triangle of the polygonal representation of the isosurface (compare Figure 3.3).

6. For each vertex of a triangle, the normal vector is calculated, again by using linear interpolation of the normal vectors at the two adjacent vertices of the cube. Usually the normal vectors at the vertices of the cube are computed using the gradients of the volume data set by applying the central differences method. New sophisticated strategies for the reconstruction of the volume gradient could be employed [38].

The triangles generated by the algorithm can be displayed using standard surface rendering techniques (for example z-buffering [13]). The normal vectors at the vertices of the triangles can be used for shading (e.g., to generate continuous shading by applying *Gouraud shading* [13], or to do *Phong shading* [13], also known as *normal-vector interpolation shading*).

Improvements

Since the first presentation of the marching cubes algorithm in 1987 many new variants of the method have been developed, the most important of them are the following:

- One of the various different optimizations to marching cubes is the *discrete marching cubes* method [40], which speeds up the original marching cubes technique by reducing the computational cost for the interpolation of the intersection point of the surface with the edges of the cube. Instead of doing interpolation for determining the exact intersection point, always the midpoint of the corresponding edge of the cube is chosen.

- Another acceleration technique regarding the marching cubes method is the *dividing cubes* method [10]. It speeds up the rendering output of the polygonal representation of the isosurface by taking advantage of very small triangles. Voxel cubes are subdivided until the extents of the extracted triangles cover only the area of a single pixel. Instead of rendering polygons, just pixels are colored.

- Some of the optimizations try to reduce the number of generated triangles (which can get rather large when processing data sets with a high resolution). One of these methods is the *adaptive marching cubes* technique [53], where a number of small connected triangles, that are nearly coplanar, are merged and replaced by one bigger triangle. A similar approach is presented in [45].

Apart from the special acceleration variants of the marching cubes algorithm, there are also some methods to speed up the conventional marching-cubes technique. The most recent of these methods (an interval-tree method) [9] exploits the knowledge that for any given isovalue not all of the given voxels in the volumetric data set have to be checked for surface contribution.

Another improvement of the optical appearance of the resulting image is, to render more than one isosurface (for different isovalues) simultaneously, by assigning the surfaces a certain degree of transparency. Nevertheless this results also in a loss of contrast in the image, and details can not be perceived as clearly any more. To overcome this disadvantage of rendering many isosurfaces simultaneously, recently techniques have been presented ([25],[26],[27]), that enhance transparent surfaces with additional information by, e.g., depicting local characteristics of the surface.

Problems of surface-rendering methods

Although research in the area of surface rendering methods for volume visualization is already a quite mature topic, there are still major problems that have not been solved satisfactorily up to now.

One of these problems is the problem of the huge amount of triangles (or polygons), that are created by surface extraction, due to the large amount of data, but also due to the accuracy, that has to be achieved in medical applications. This large number of polygons in the representation currently prohibits interactive update rates, which are required and asked for by the users nowadays.

Another big disadvantage of surface-rendering methods is that they represent only a small part of the whole data set, and never all the data at once.

Furthermore it is sometimes not easy to define a threshold value that exactly shows the features of the data set the user wants to investigate.

3.1.3 Direct volume rendering

Direct volume rendering (DVR) is another approach to the visualization of volumetric data. DVR techniques do not only extract parts of the data sets (isosurfaces) but rather display all of the data in it's entirety. The values of the voxels are mapped to displayable properties such as for example color, lightness and opacity. This mapping is one by so called *transfer functions.* There are a large variety of different methods in direct volume rendering, which can be roughly divided into two main categories, depending on the kind of space they operate on (see Figure 3.4):

- *image space methods* - these methods gather information required for the visualization depending on the geometry and the orientation and location of the image plane. An example is the ray-casting technique [33].
- *object space methods* - in contrast to the image space based approaches, these methods project information derived from voxels onto the image plane, where the result image is accumulated. An example is Splatting [58].

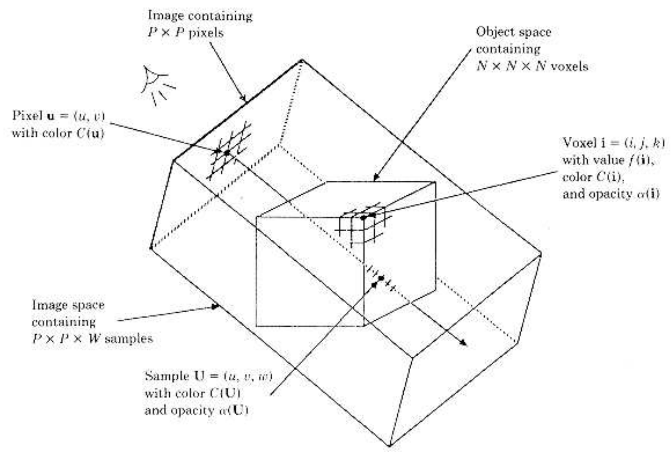


Figure 3.4: Coordinate systems used during volume rendering [34]

In the following subsections one representative method (the most used method) for each of the two classes will be discussed in more detail. After that open problems in direct volume rendering will be highlighted briefly.

Ray casting

Ray casting [33] is an image space method, where for every pixel of the image plane, a ray is traced through the volume data set. This ray is oriented in the viewing direction and originates at the eye point of the viewer (viewpoint). At equidistant locations along the ray data values are evaluated by trilinear interpolation of the surrounding eight voxels. These data values are mapped to color and opacity values with the help of transfer functions.

One example for a transfer function in medical data sets could be the following: data values representing the density of bones are mapped to white color and very high opacity (nearly opaque), that means that they are not transparent, whereas data values corresponding to the density of muscles are mapped to color red and a low opacity (very transparent).

These color and opacity values are then accumulated along the ray using *alpha channel blending* in the corresponding order and in a way, that the final appearance of the image is optically correct.

The result is used as the color value for the pixel of the image plane, which the ray was calculated for.

As traversing the ray through the volume data set is computationally expensive and very time consuming, there have been developed several speed-up techniques for the standard ray casting. Two of these speed up methods are presented in another publication [34] and are quite useful.

The first of these two techniques accelerates the calculations by applying a hierarchical data structure on the data sets. Voxels with the same density values (data values) are merged in upper hierarchy levels (see Figure 3.5). Therefore, when checking the upper hierarchy levels for the value of the voxel, sometimes big regions where the data values correspond to 0 don't have to be considered any more, and can be skipped. This speeds up the

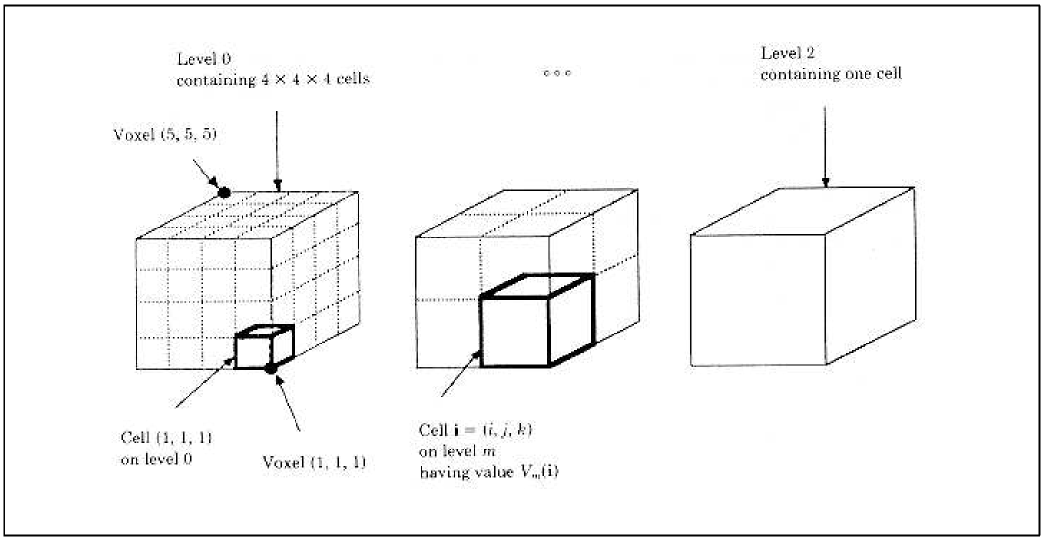


Figure 3.5: An example for a hierarchical enumeration of object space [34]

color compositing calculations.

The second acceleration technique uses an opacity threshold to adaptively terminate ray tracing whenever the accumulated opacity is greater than this threshold, as the following voxels will not contribute significantly to the result image any more.

Splatting

Splatting [58] is an object space method, where all voxels are projected onto the image plane successively. The order in which the voxels are projected results from the distance to the viewer, the voxels with largest distance are processed first.

With this method each voxel is represented as a *footprint* (cloud of points). The projections of these footprints to the image plane are precalculated in a lookup table. The size of this *generic footprint table* (see Figure 3.6) influences the quality of the result image significantly. This method is somehow similar to splatting a snowball to a wall (hence the name). After having projected all voxels to the image plane using their footprint representations, the result is a high quality image of the volumetric data set.

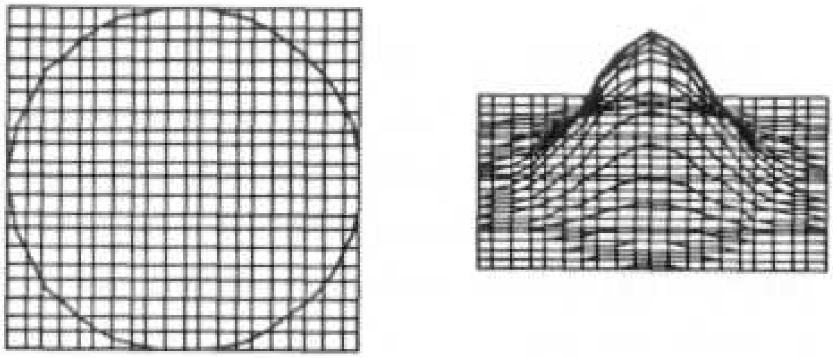


Figure 3.6: Representation of a generic footprint table [58]

The standard splatting method has also been accelerated by a hierarchical approach [32], again using an octree for data representation. For this approach also footprint tables for each level of the hierarchy with different accuracies in resolution have to be generated.

Open problems in direct volume rendering

Although direct volume rendering is an important research area for quite some time, there are still some major unsolved problems. The most important ones are listed below:

Similar to surface extraction methods it is also very difficult to determine the most suitable visualization parameters in direct volume rendering. For example the specification of useful transfer functions, that provide the user with the image, that he wants to obtain is a very hard task and not always easy to solve by a novice user.

Another problem is, that computation times and display rates for standard direct volume rendering are still not interactive. Therefore a compromise has to be taken into account, which either results in a loss of information or restricts the users freedom in steering the the application.

3.1.4 Segmentation

Segmentation is one of the main pre-processing steps in medical volume visualization. During segmentation of the data, the interesting parts of the data are emphasized, and the rest of the data is masked out. This can for example be used to obtain only the anatomical data of the human brain out of an anatomical data set of the full head of a patient.

As segmentation of medical volume data is a very important field of research, many different approaches have been developed to overcome the problems encountered during the segmentation process. These problems include for example different grey-values in the measured data sets for regions of the same data due to inhomogeneities in the measurement. Up to now there exists no reasonably stable technique, that works fully automatic. There is a lot of ongoing research in this area.

It is out of the scope of this work to give an overview over all existing approaches. The anatomical data of the human brain used in this project was segmented manually, segmentation issues were not addressed.

Some of the most well known and recently published segmentation approaches include *intelligent scissors* [42], *watershed* algorithms ([37],[54]), *region growing* techniques [39], *neuronal networks* based approaches ([8],[47],[55],[57]), and techniques using *snakes and deformable models* ([28],[48]).

3.2 Multimodal volume visualization

In multimodal volume visualization all methods and problems discussed and presented in the previous sections are also relevant, but there exist some additional problems as well. Furthermore some interesting approaches to the topic of this thesis are presented. Two topics especially relevant in *functional and anatomical multimodal volume visualization* are presented in the following subsections, namely the *registration* of the various data sets involved in the visualization process, and the special requirements and properties of functional volume data.

3.2.1 Registration

Since information gained from two different imaging modalities is usually of complementary nature, proper integration of useful data obtained from the separate volumes is desired. A first step in this integration process is to bring the modalities involved into spatial alignment, a procedure referred to as *registration*.

A short overview of the different types and approaches to registration in medical imaging is given as outlined in [36]. A detailed description and a very good literature review of most of the recent applications and publications in the field of medical image registration can be found in the publication by Maintz et al. [36]

Two major classification criteria of registration approaches exist at the moment. The first criterion determines whether the registration is between two anatomical data sets or with the registration between an anatomical and a functional data set. The second criterion classifies the registration according to the origin of the data sets: the different data sets can either originate from the same modality, but with different acquisition times. This case is referred to as *monomodal* registration. On the other hand, the data sets can also result from scanning at different modalities, this referred to as *multimodal* registration.

In the scope of this work only the case of *multimodal anatomical-functional registration and volume visualization* is addressed.

Besides these two major classification criteria, several other criteria can be used to characterize the different types of registrations. These are discussed now briefly. Note, that in the context of registration of volume data, a volume is often also referred to as *image*, as is done in the following.

Dimensionality

The data can either be given only in spatial dimensions, or it can also be represented by a time series (two or more images), along with the spatial dimension. Registering has to deal with data sets that are in 3D only, in 2D only, or one data set is in 2D and the other one is in 3D.

Nature of the registration basis

There exist three major types of registration bases:

- *extrinsic* methods, based on foreign objects introduced into the image space.
- *intrinsic* methods, i.e. based on the image information scanned from the patient.
- *non-image based* - this is possible, if the imaging coordinate systems of the two modalities involved are somehow calibrated to each other.

Extrinsic methods can be further divided into *invasive* and *non-invasive* ones. *Intrinsic* methods can be split up into *landmark based* methods (these are based on significant points, like for example, location of the nose), *segmentation based* methods (trying to register two segmented, i.e., binary images), and *voxel property based* methods (depending on the grey value of the given data).

Nature and domain of the transformation

The nature of the transformation can be divided into four major classes, namely into *rigid*, *affine*, *projective*, and *curved* transformations. An image coordinate transformation is called *rigid*, when only translations and rotations are allowed. If the transformation maps parallel lines onto parallel lines it is called *affine*. If it maps lines onto lines, it is called *projective*, and finally, if it maps lines onto curves, it is called *curved*. The domain of the transformation can be either *global*, if it applies to the entire image (volume), or *local*, if only subsections of the image (volume) are transformed by the transformation.

Interaction

When considering registration algorithms, three levels of interaction can be recognized. Registration can be *automatic*, where the user only supplies the algorithm with the image data and possibly information on the image acquisition. It can also be *interactive*, where the user does the registration himself, assisted by software supplying a visual or numerical feedback of the current transformation, and maybe an initial transformation guess. The third possible level of interaction for registration algorithms is *semi-automatic*, where the interaction required can be of two different natures: the user needs to initialize the algorithm, e.g., by roughly segmenting the data, or steering the algorithm, e.g., by rejecting or accepting suggested registration hypotheses.

Nowadays a majority of researchers strive for fully automated registration algorithms, but it can be discussed whether this is desired or even possible in all current clinical applications. The point is that many current methods have a trade-off between minimal interaction required and speed, accuracy, and robustness. Some methods would doubtlessly benefit if the user would aid the system by steering the optimization.

Modalities involved in the registration

Four classes of registration tasks can be recognized based on the modalities that are involved. In *monomodal* applications, the images to be registered belong to the same modality, as opposed to *multimodal* registration tasks, where the images to be registered are acquired by different modalities. In *modality-to-model* and *patient-to-modality* registration only one image is involved, and the other "modality" is either a (mathematical) model or the patient himself.

Subject

When all of the images involved in a registration task are acquired from a single patient, we refer to it as *intrasubject registration*. If the registration is accomplished using images of different patients (or a patient and a model), this is referred to as *intersubject registration*. If one image is acquired from a single patient, and the other image is somehow constructed from an image-information database obtained using images of various subjects, it is called *atlas registration*.

This enumeration of different criteria for subdividing the field of medical image registration is of complete. There exist several other additional criteria. To mention all of them, would be beyond the scope of this thesis.

An open problem for all the different types of registration found in the medical field is the *validation* of the registration results. This validation includes not only *accuracy*, but also several other points, like *precision*, *reliability*, *robustness/stability*, *resource requirements*, and *clinical use*.

3.2.2 Functional MRI and other functional modalities

Functional MRI (as described in Chapter 2) is only one of the many modalities from the rapidly emerging field of functional modalities in use recently. All of these functional modalities (see Chapter 1 for a discussion of some of them) have at least the following similar attributes:

- All of the functional modalities produce volume data sets with a significantly lower resolution than the anatomical data sets.
- They usually scan volumetric data sets (or other data, which are later on reconstructed to volumetric data) with smaller spatial extension than can be gained from anatomical data sets.
- The measured data values are less accurate than in comparable anatomical data sets.

All these attributes originate in the significantly shorter acquisition time necessary to measure functional activities of the human brain. Anatomical data sets are acquired using much more time, as they have to be acquired only once. Functional data has to be measured for example 30 times over a time period of 2 minutes, while the test is performed.

All these constraints listed above restrict the accuracy of every type of visualization of these functional data together with an anatomical data set.

The field of multimodal visualization of anatomical-functional data of the human brain is currently a very active research area.

Today neuroimaging experts still predominantly investigate multimodal scans from twodimensional slices of the human brain, where the functionally activated regions are highlighted in some special way (see Figure 3.7, top row). Viewing several slices at once (side by side) is referred to as the *image gallery* approach. This way of investigating the data sets provides detailed 2D anatomical context information. Nevertheless, this way of dealing with multiple data sets does not aid the user in forming a threedimensional model of the underlying spatial relationship of functional activations and the anatomical geometry. Other approaches try to increase the amount of presented information by utilizing *maximum intensity projections* (MIP) [51], mapped onto 2D reference grids with brain outlines [15] (see Figure 3.7, middle and bottom row). These representations aid the task of side by side comparisons, but provide only very rough information of the anatomical context. Therefore it is very hard for the user to mentally reconstruct the threedimensional structure of the patients brain.

Threedimensional surface renderings of the brain, where the functional activation is encoded by coloring the cerebral cortex provides a misleading anatomical context. Activated subcortical foci below the surface of the brain are obscured. Furthermore, all foci seem to lie on the surface of the cortex, as no depth cues are provided.

Direct volume rendering would be able to overcome this problem. Assigning transparencies to data ranges would allow the deeper structures to

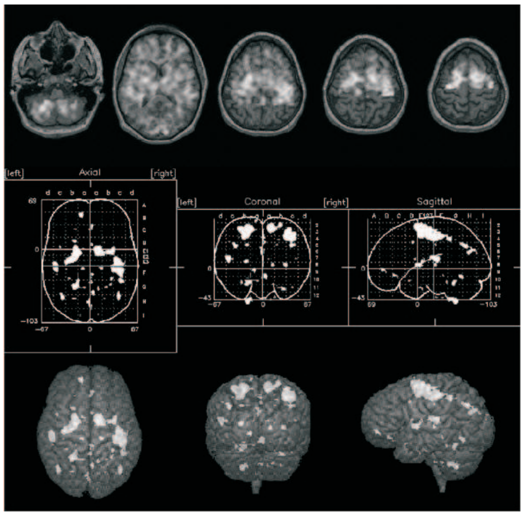


Figure 3.7: Conventional approaches to multimodal visualization [49]. Top, selected MRI slices are merged with functional data by pixel interleaving [50]. Middle, the maximum intensities of the data are projected onto axial, coronal and sagittal planes bearing brain outlines and millimetric Talairach coordinates [56]. Bottom, the maximum data intensities are projected along the line of sight onto the cortical surface.

be visualized. Therefore direct volume rendering would be the most usable approach to the solution of this problem. Usually, however, anatomical detail has to be sacrificed, when employing transparencies [22], decreasing the visual appearance of details significantly.

For these reasons K. Rehm and other authors recently proposed, that the anatomy of the brain should not be visualized threedimensionally at all [49]. They use symbolic representations known from other fields of visualization (like fluid or information visualization) to encode the location and size of foci. In order to place the foci into anatomical context, they are projected onto the walls of a cube-like environment together with different representations of the brain. A similar approach of adding information to the visualization is presented in Section 5.3 of this work.

Direct volume rendering still seems to be the most useful approach to clearly give visible hints on the location and size of activated regions. Zuiderveld et al. [59] showed interesting results and concepts regarding multi-modality visualization of medical volume data. In this thesis some extensions to overcome the problem of reduced visibility will be presented in Chapter 5.

Chapter 4

Volume visualization systems

During the first phase of this diploma thesis, many different volume visualization systems have been investigated and evaluated. These were either general volume visualization systems or special purpose volume visualization systems, for example especially useful for volume visualization of medical datasets, biomedical datasets, or chemical datasets.

Some of the investigated volume visualization systems are listed in table 4.1. The names of the systems are given in the first column. The second column reports, if the system is a general volume visualization system, or if it is rather a special purpose volume visualization system. In the third column the costs for the system are given, or at least some sort of comparison to the others is shown. In the fourth column, the capability to work on multimodal data and also visualize multimodal data is stated. Sometimes a rating, to what extend these capabilities are available, is provided. In the last of the five columns some additional short remarks are given, longer remarks and sometimes also descriptions can be found in the following subsections for the most important of the mentioned systems.

Most important for the investigation of the different systems and their capabilities were compatibility, availability, and their degree of support for multimodal data sets and the visualization of these data sets. Some of the systems have not been fully investigated with respect to all points, if at some earlier stage of the investigation it became obvious, that one of the main features is not supported at all. Some of the listed systems are described in more detail in the next subsections. The entries for the table are sorted alphabetically.

Not all investigated volume visualization systems are enumerated in this list, only the systems investigated in more detail are described. On the other hand it might be possible, that not all available volume visualization systems, that would have been of interest, were known to us. The commercial market for volume-visualization software is expanding rapidly. There exist hundreds of different systems in this field of interest.

4.1 3DVIEWNIX [18]

This program was developed by the Medical Image Processing Group, Department of Radiology, University of Pennsylvania, Philadelphia. It is a rather inexpensive software system for the visualization and analysis of multidimensional images. It is designed to be data, machine, and application independent. It is able of handling object information from multiple modalities and longitudinal acquisitions. 3DVIEWNIX is an open system, it is distributed together with the source code (C-code).

This system supports features like:

- different segmentation methods
- filtering and masking (for assisting segmentation)
- image algebra (addition, subtraction, ...)
- interpolation
  - create isotropically sampled data of lower or higher resolution than the input data
  - interpolation in n dimensions and with many interpolating functions

Systemname	General or special purpose system	Costs	Multimodality supported?	Short remarks
3DVIEWNIX	medical	1.000$	yes, multimodality is supported	unfortunatly I was not able to test it, very promising, and claims to handle real multimodality
Alice	medical	1495$ (academic version), 2995$ (non academic)	no	mainly image processing, mostly 2D, some first steps to 3D, but only basic
Analyze	biomedical	16.000$ (academic), 26.000$ (non academic)	yes, multimodality is supported very strongly	strong in 3D, also real volume rendering is supported, but non-intuitive user-interface, very hard to handle
AVS	general	expensive	no	works on data flow concept, strong in 3D, no special medical routines, very memory-demanding
Biological Detection Systems	biological	at least 10.000$ (basic package)	not checked	only for MacIntosh Computers, no compatibility, software comes together with hardware, not interesting for this investigation
BioQuant	biological	no information	no	only Windows, not for medical purpose, not interesting for this investigation
Cheshire	medical	approx. 4995$	no	provides state of the art multiple slices, animations of slices, measuring tools, no 3D
C_Images 3D	general	no information	yes, but only used for anatomical-anatomical fusion	mainly for image processing, not interesting for this investigation
dicom2	image conversion (and processing) tool	freely available	no	for image fileformat conversions and some rudimentary image processing tasks only, no 3D
Dr Razz	image display and analysis	freely available	no	only for Macintosh computers, image display and analysis program
Voxel-Man	medical	approx. 1000$	yes	tool for teaching and studying anatomy and its radiological appearance, unfortunately it can not process other data than included
VolVis	general	freely available	not fully	very strong "real" volume rendering system, unfortunately multimodality support is questionable

Table 4.1: Different Volume Visualization Systems in comparison

- measurements
- registration (for merging information from multiple modalities)
- slicing
  - multiple input volumes of any dimensionality can be handled simultaneously
  - multiple color maps
- specification of volumes-of-interest
- surface rendering (also multiple objects with translucency and color)
- thresholding
  - multiple intervals can be specified
  - isosurface generation at any resolution
- volume rendering (new method: shell rendering, fuzzy surfaces)

4.2 Alice [16]

Alice is a 3D-oriented analysis software system for scientific and especially medical imaging. It uses mainly 2D image representations. Multiple slices are shown next to each other. Showing the slices can also be animated. Another feature are multi-planar views, where three reconstructed slices, all orthogonal to each other, are shown, and a cross hair can be used, to browse through the data set.

Also one-dimensional graph views of the data along a line can be shown, giving a profile of the data along such a ray.

In the newest version available, first steps towards 3D visualizations are made by using simple surface rendering techniques. A solid 3D object is generated, which can be rotated and manipulated in some (limited) ways.

This system implements rather basic approaches. It has its strongest capabilities in data measurements and statistical processing of the data. The most outstanding capabilities of Alice seem to be the image processing features. 3D volume visualization does not seem to be the main aim of this system. Surface rendering is just supported in a very basic way. Multimodal visualization is not supported at all.

4.3 Analyze [11]

Analyze provides the user with a comprehensive set of tools for visualization and analysis of image data. It supports multi-dimensional data from many different modalities and image sources.
It provides a multi-window environment for interactive visualization, manipulation and measurement in both 2D and 3D multimodal image data. It supports a broad range of different computer systems, as well as a variety of different data types and structure formats.

The following features are supported by Analyze in the latest version available at the moment:

- *interactive 2D image generation and display*, also multiple image views and animations
- *3D image segmentation*
  - simple region growing
  - semi-automatic segmentation using advanced 2D and 3D morphological operations with protocols saved for future use
  - manual editing and connecting/deleting of multiple objects using region growing, manual tracing and spline curve manipulation
  - multimodal image classification and object definition via multivariate analysis of cross-modal data using supervised and unsupervised classification routines.

- *multimodal image analysis*
  - 2D and 3D geometric image registration across multiple modalities using voxel matching, object surfaces or point files.
  - multimodal registration by fiducial markers
  - fusion-image generation and display with linked cursor for analysis
  - neuronal network classifications
- *volume rendering using ray casting*: many different placement algorithms and controls for the rendering algorithms are available
- *surface extraction*
- a large number of different *image and data manipulation techniques*, most of them originating from the field of image processing
- several types of different *image measurement methods*

Because of the features and capabilities mentioned above, especially in the field of volume visualization of multimodal data, this system would be the best solution to the stated problem of this thesis.

On the other hand this system is rather expensive. At the moment it is not affordable for the low cost selling we are aiming at.

4.4 AVS [24]

AVS is the *Advanced Visualization System* of Advanced Visual System Inc. It is a very powerful, general purpose visualization system, that supports volume visualization very well. AVS works on the data flow concept, different modules (input modules, filter modules, output modules) can be assembled into a data-flow net to generate a user-tailored visualization method. Modules can be added by programming them from scratch, if the functionality is not yet available at one of the module repositories in the Internet (e.g.: [5]).

A description, how to programm modules for AVS can be found in manuals accompanying the software [1]. Unfortunately, AVS is very expensive, and also rather memory-demanding, as well as rather old-fashioned concerning software architectural designs. This causes quite often problems for the users.

The newest product from Advanced Visualization Systems is *AVS express*, the PC-platform version of the AVS system.

4.5 Cheshire [17]

Cheshire provides state of the art medical image analysis features and visualization tools, such as for example:

- single image and multiple image views (the classical image gallery approach: multiple slices next to each other)
- multiplanar views: concurrent displaing of the three orthogonal views while browsing through the data set
- one dimensional graph viewing of profiles for measuring, etc.
- animated slice sequences ("cine view")

A typical screen shot taken from a session working with Cheshire is shown in Figure 4.1. It shows the main features of the system, which does not provide any 3D capabilities.

4.6 C_Images 3D [2]

C_Images 3D is an image processing, image analysis and also image fusion system developed by Foster Findlay Associates. It is capable of multimodal matching of CT and MRI data (only anatomical-anatomical fusion). It is a modular system, providing also functions implemented as modules for AVS (see Section 4.4).

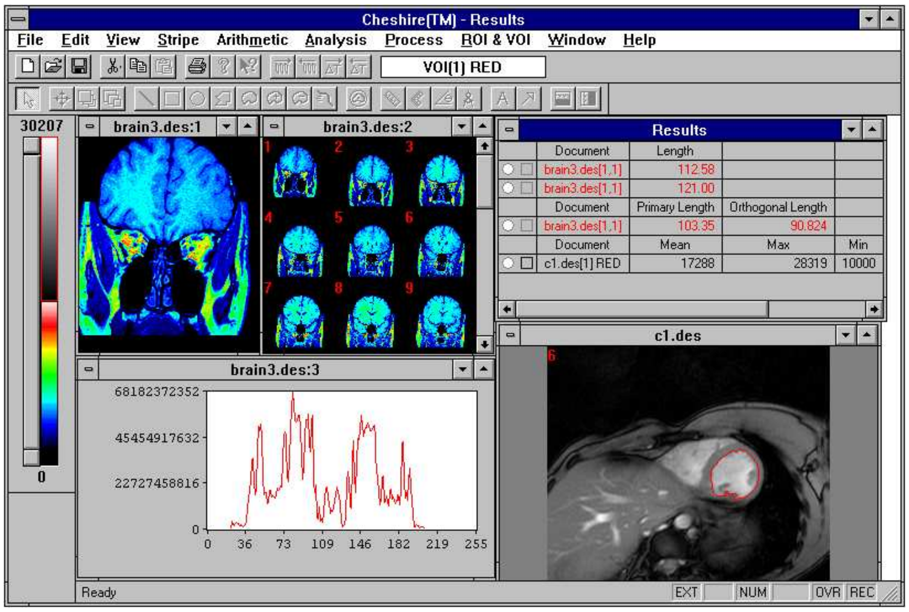


Figure 4.1: Screen shot taken during a typical session with Cheshire [17]

It supports real three dimensional display and measuring functionality, sub volumes can be defined, regions of interest and point sets can be used. More information on the capablities of this system can be looked up at [2].

4.7 Voxel-Man [44]

"Voxel-Man Brain & Skull" atlas (and "Voxel-Man Junior") are the result of a research project investigating new computer-based techniques in medicine. The project was carried out by Höhne et al. at the University of Hamburg. It is a tool for teaching and studying anatomy and its radiological appearance. It does also support multimodal data, anatomical-anatomical data as well as functional-anatomical data. Unfortunatly it can not process any other data, than the provided data that comes with the software. This software can be seen as an interactive atlas. It has the following capabilities:

- unlimited choice of views, including endoscopic and stereo viewing

- free cutting operations, allowing simulation of dissection, surgery or puncture
- simulation of radiological imaging without restrictions to viewing direction or slice orientation
- simultaneous display of names, descriptions and histology added to the anatomical three dimensional display
- perform distance measurements, save images and animations

One of the images generated with Voxel-Man can be viewed in Figure 4.2. More information about this project can be found at [44].

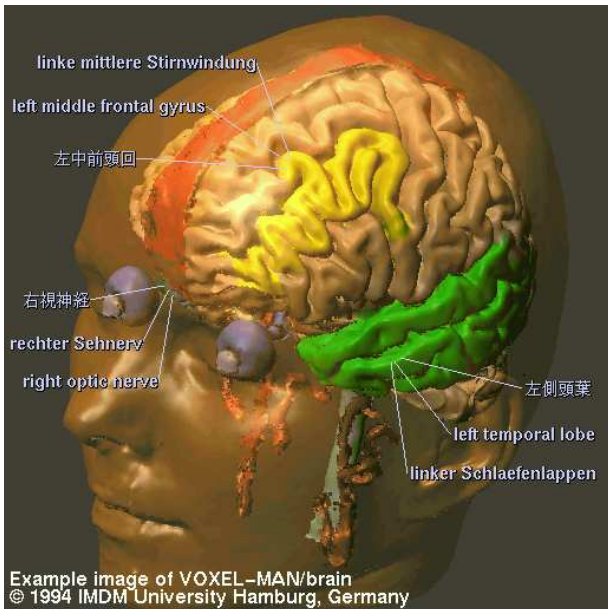


Figure 4.2: Sample image produced with the Voxel-Man system

4.8 VolVis [43]

VolVis is a general purpose volume visualization system, that was developed at the Department of Computer Science, State University of New York at

Stony Brook. It is an open volume visualization system, that can be extended by programming user defined modules. It is freely available, is written in C and works on most Unix platforms, e.g., SGIs. Its capabilities to support real volume rendering are quite high. The user interface, however, does not allow a great freedom in steering the visualization process. The programm currently supports basic functionality for the handling of multimodal data sets.

More information on this general purpose volume visualization system can be gained from literature [3] or at [43].

Chapter 5

Volume visualization techniques for multimodal, anatomical and functional data of the human brain

In the following sections four different techniques for volume visualization of multimodal data of the human brain, consisting of anatomical and functional data, are presented and described in detail. In the next chapter the implementation issues and the concrete implementation of these four visualization techniques will be presented.

5.1 Multimodal ray casting

The first technique investigated in the scope of the presented work is called multimodal ray casting. In this approach the two volumes containing the anatomical and the functional data were traced simultaneously, and the contributions of two viewing rays were accumulated into the corresponding pixel in the final image.

This technique works as follows:

1. For each pixel two rays (equal in origin and direction) are cast, one for each of the two volume data sets; samples are calculated at uniform distances along these rays and stored for later use. This is a precomputational step, which is very memory demanding, as many samples for each of the rays have to be stored for later use. This helps for faster image generation afterwards, if only the transfer functions are changed, but the viewing direction is fixed.

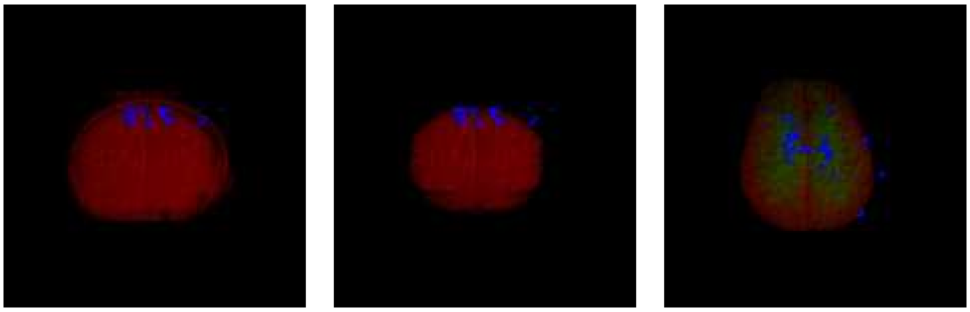
2. For each pixel the sample points along the two corresponding rays are processed in front to back order to determine the color of the pixel in the final image. Therefore always the next sample point at each ray is considered. The sample point with the smaller distance to the origin of the ray (the pixel) is the one processed in the incremental compositing step.

3. Once the closer of the two sample points on the two rays is determined, it is used to contribute to the color of the pixel. This is done as in each standard ray casting process, by using a transfer function to map the data value at that point to a color and opacity value. These color and opacity values are added to the so far composited color and opacity value of the pixel by using a standard front-to-back compositing approach. The transfer function, that has to be used for the mapping is either the anatomical or the functional transfer function, depending on which sample point of the two available ones has been processed.

4. After this compositing step has been performed, the next sample point of the currently processed ray is computed and the process starts again by choosing the closer sample point and continuing at step 3. This algorithm is performed until all of the sample points at one of the two rays are processed, than the other ray is processed as in standard ray casting until it leaves the volume-data set. Early ray termination

is used, when an opacity threshold is reached to avoid unnecessary computational effort.

In Figure 5.1 three images produced with this multimodal ray-casting technique are presented. Figure 5.1(a) was produced using an unsegmented anatomical data set of a human head, Figures 5.1(b) and 5.1(c) were rendered using the segmented data of the human head from the same anatomical data set as before. More results with higher resolution produced by using this visualization technique are presented in section 7.1 of this thesis.



(a) (b) (c)

Figure 5.1: Three images produced by the multimodal ray-casting technique: (a): not segmented, (b) and (c): segmented data sets were used.

The technique has several disadvantages:

- The precomputations of all the samples along each ray are rather time-consuming, as the ray casting step has to be done for both volumes. Additionally no early ray termination during this precomputational composition of the ray samples can be performed, as early ray termination is depending only on the transfer functions that are used in the compositing step, and these are not fixed.

- The storage of all the samples for each ray is quite memory consuming, with quadratic growth corresponding to the image size (resulting from the quadratic image shape used in this approach). As for each pixel of the final image, always two rays with several hundred sample points have to be calculated, this approach is clearly not scalable.

- The display of functional data at a specific point in the image allows not for a concurrent visualization of the anatomy at that location of the anatomical volumetric data set. This results in a considerable loss of information during the visualization step.

- As the spatial resolution (and therefore also the accuracy) of the functional data sets is considerably lower than that of the anatomical data sets, functional activities are displayed in regions, where no anatomy of the human brain is located. This is due to interpolation inaccuracy resulting from the low resolution of the functional data set, and additionally due to the fact, that measured activation potentials in the special experiment of tipping thumb and index fingers at each other usually lie on the cortex (outer border) of the brain.

To overcome these disadvantages the data structure presented in the next section was developed.

5.2 Transfer-function volumes

Transfer-function volumes have been developed to overcome the main disadvantages described in detail in the previous section. *Transfer-function volumes* are a volumetric data structure, where each voxel corresponds to one voxel in the anatomical data set, that is to be visualized. In the transfer-function volume at each voxel an index to a transfer function is stored. The voxels of the transfer-function volume are considered to be of the same size as the voxels of the anatomical data set. They are also considered to be in the same spatial positions as well as in the same order than the corresponding voxels in the anatomical data set. For this reason the transfer-function volumes can be considered to be a data structure providing additional information to each point in the anatomical data set. This helps to get a very useful and rather simple integration of the anatomical and functional data sets.

Every transfer function has to be entered in the transfer-function volume before it can be used. At this "check-in" an index is assigned to this transfer function, so that it can be addressed later on by the transfer-function volume.

The registering of the anatomical and functional data sets and the combined visualization of these two data sets in a concurrent way works as follows:

1. All voxels of the transfer-function volume are assigned the index value of the transfer function of the anatomical data set.

2. If only parts of the volume data set are of interest, the other parts of the data set could be hidden by assigning their voxels an index to the "NULL-transfer function", which has zero opacity. These parts do not produce any visible contribution in the visualization step. Note, that this step (Step 2) can also be made after the registration procedure has been performed (after Step 4). Then the functional activations in these regions of the anatomical data set are not visualized either.

3. The registering works non-image based. The two coordinate systems of the two data sets are known (provided by the user, see Section 6.1.1). For every voxel in the functional data set that has a data value different from zero, the eight spatially neighboring (surrounding) voxels in the corresponding anatomical data set are identified using the knowledge of the two coordinate systems of the data sets.

4. The index of the transfer function in each of the eight voxels in the transfer-function volume is set to the index of the transfer function assigned to the functional data set.

5. At the visualization step for each voxel in the anatomical data set the corresponding transfer function stored in the transfer-function volume is used to determine the color and opacity contribution to the ray. *Only*

the anatomical data is visualized during this step. In areas which correspond to non-zero functional activation potentials, different transfer functions are used.

The transfer-function volume provides the following functionalities:

- Evaluate the correct color and opacity for a point within the volume by using the transfer function assigned to that point.
- Set a new index value for each voxel being represented by another transfer function.
- Set a new index value for a complete sub block of the volume.
- Use a "NULL-transfer function" (evaluates opacity to zero for all data values) to sculpt the graphical representation of the data set. Using for instance a rectilinear setting scheme, blocks can be cut out. By the specification of cutting planes, entire half spaces can be removed.
- Get the currently active transfer function (an index to it) for each voxel of the anatomical data set.

In Figure 5.2 three images produced with transfer-function volumes are shown. The three images show three different views of the same volume data sets. More images with higher resolution and accuracies produced with this technique are presented in the results section (section 7.2) of this thesis.

This volume visualization technique for visualizing multimodal (anatomical - functional) data using transfer-function volumes has the following advantages compared to the previously discussed technique of multimodal ray casting:

- One of the big advantages of this technique is that both, the anatomical and the functional data can be displayed concurrently. The anatomical data is considered in the ray casting step, but where there is also

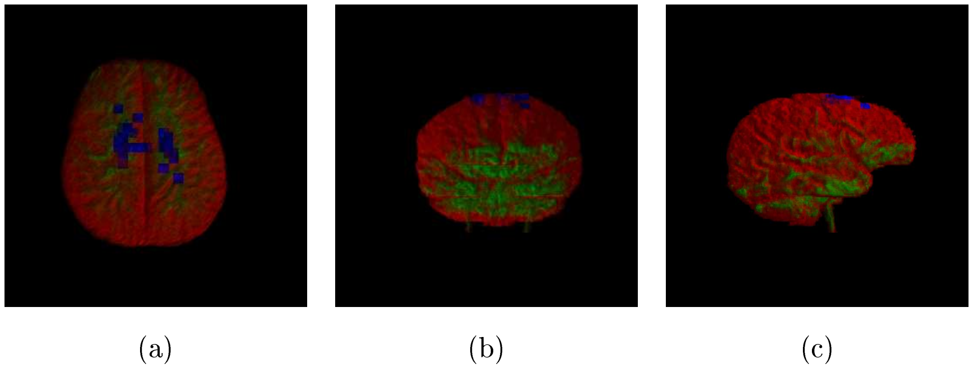


Figure 5.2: Three images produced by using transfer-function volumes. The same anatomical and functional data sets were used: axial view (a), coronal view (b), and sagittal view (c).

functional activity measured in the functional data, another transfer function is used.

- Another advantage is, that functional activities are only displayed in regions where anatomical data is encountered. Therefore the amount of incorrectly visualized functional activities due to interpolation and inaccuracy errors is reduced. In addition to removing inaccuracy errors, also the problem of emphasizing outliers in the measured functional data is solved rather satisfactorily in this approach, as they are not shown either. This is due to the fact, that no anatomical data is available in these regions. (compare figures 5.2 (a) and (b) with figures 5.1 (c) and (b)).

- As computational efforts are concerned, this method needs nearly no precomputation time (a few seconds for standard anatomical and functional data sets) to register the data sets. After the registration has been performed, ray casting behaves nearly the same as in standard cases, only instead of using the transfer function directly, a call to the transfer-function volume has to be included. No other computational overhead as for example encountered in the previously described method (where two rays have to be cast for each pixel) occurs.

A disadvantage of this volume visualization technique, is the following: As the implementation of the transfer-function volumes and the corresponding registration are done in a discrete way, it is not possible for any voxel within the anatomical data set, to have more than one active transfer function associated with it. This leads to rather large cube-shaped areas within the transfer-function volumes, where the functional transfer function is active. This results from the fact, that the functional data sets usually have a much lower resolution than the anatomical data sets. Another problem originates from the algorithm itself. For each voxel in the functional data set all of the surrounding neighbors in the corresponding anatomical data set are either set to the functional transfer function, or not. There is no way of smoothly interpolating any transfer functions. This will be solved by so called *interpolated transfer-function volumes*.

5.3 Multiple Views

The images produced with transfer-function volumes are already of useful appearance and show the interesting details quite satisfactorily. Nevertheless, the three-dimensional context of the data is missing. From a single still image it is very hard to get the correct visual impression of depth.

In this technique the image produced with the transfer-function volume is enhanced by adding three different views of the object to the final image. This is achieved by placing three image planes behind and below the object. Images displayed on this image planes are generated by using the ray casting technique as described in Section 5.2.

On these planes multiple views of the same object are given simultaneously, hence the name *multiple views*. The image planes used for these multiple views are quite different in their properties from, e.g., mirrors or shadow walls. Placing a real mirror at the same location (an approach used by A. Kaufman [3]) would result in completely different images.

The image planes used in this approach are parallel-projecting image planes, the images displayed on them are produced by a ray casting step for each of the additional image planes. The viewing direction for this ray casting step is from behind the image plane to the object. These images are stored as textures for the compositing of the final picture ray casting, where the image planes and these textures are accumulated correctly to give the resulting image.

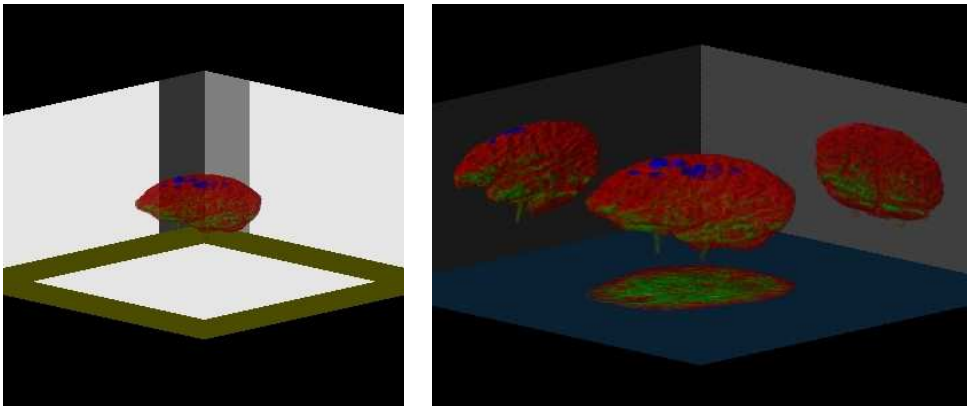
The advantages of using projection planes for the multiple views over the usage of real mirrors, as presented for example in [3] include the following:

- By using parallel-projection planes the images shown on the planes behind and below the object are less occluded, than by applying physical correct mirroring. Therefore, more information is visible. By adjusting the distances and sizes of the image planes used in the multiple views technique, the visibility of the complete images on the planes can be achieved.

- The users (doctors) are used to orthogonal projections of the data given as slices. They are therefore in favor of keeping the orthogonal projections instead of using real mirrors, where some additional distortions are present.

- The generated textures for the additional image planes can be viewed separately, when orthogonal projections are used, and give views from the respective directions on the volume data sets. This also pleases the users, as they are used to twodimensional representations, especially in the three orthogonal slice directions. Theses views could be viewed simultaneously to the multiple view visualization, adding help for the users, to interpret the threedimensional representation more easily.

In Figure 5.3 (a) the additional image planes are illustrated without the textures. The white regions on these image planes show the location where the texture is applied in the final ray casting step. In Figure 5.3 (b) the resulting image rendered with all the textures is presented.



(a) (b)

Figure 5.3: The principle of placing multiple views into the scene for enhancing volume visualization: without (a) and with (b) textures added to the additional image planes.

The developed multiple views are capable of some features, which can be chosen by the user for adjusting the results according to different visualization tasks. In Figure 5.4 (a) and (b) for example the background colors of the three additional image planes has been chosen differently. One of the two solutions might be better for on-screen viewing, while the other one might be better suited for demonstrations using overhead slides or a video beam. Furthermore, the resolutions, spatial sizes and distances of the image planes can be changed, etc.

In Figure 5.4 (c) and (d) the capability of the transfer-function volumes to cut away blocks of data (see Section 5.2) is illustrated. In these images also the advantage of multiple views becomes obvious, as the three-dimensional extend of the cuts is clearly visible contrary to if only a single image would have been displayed instead.

For more images in higher resolutions and accuracies refer to the results

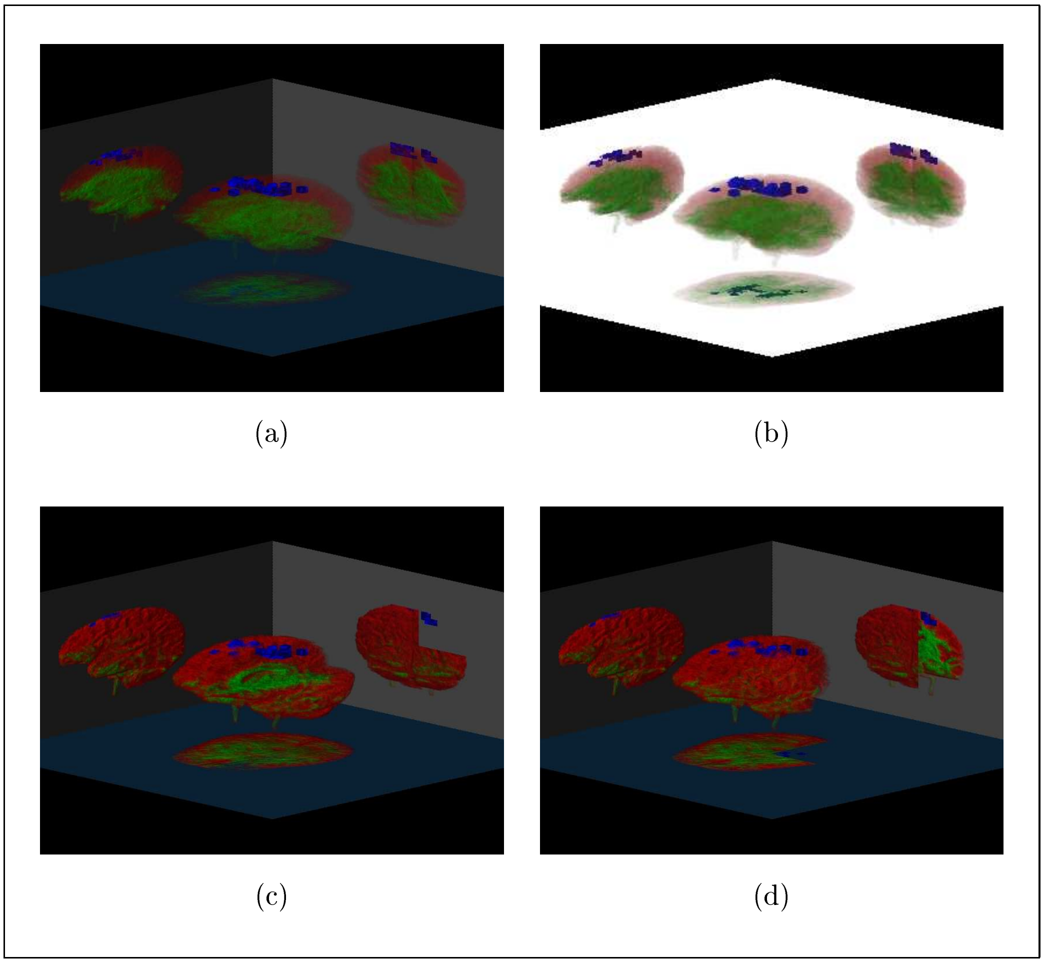


Figure 5.4: Some examples for different capabilities of this technique: (a) and (b) show different coloring of the image planes, (c) and (d) show cutting away of blocks, to improve insight into the inner structures.

section of this work (see Section 7.3). References to URLs providing electronic versions of the images and some pre-rendered animations are also given there.

The advantages of multiple-view images are the following:

- For still images the multiple views on the object (data sets) help to perceive a better understanding of the three-dimensionality and the spatial relations of the imaged data.

- When the represented data is in motion like in an animation or in interactive usage (if possible), multiple views improve the ability to better recognize the properties further.

- By using multiple views features in the data can be visualized, that could not be shown in conventional single images, as not all details are visible from one viewing direction. For example two different features present in the data on two opposite sides of the data sets, might not be visualized simultaneously when looking onto the data only from one single view point, as one of the features might always be occluded by the rest of the data.

On the other hand, the biggest disadvantage of this technique is clearly the significant slowdown in speed. One would presume the multiple views technique to be four times slower than rendering a single image, as four images have to be rendered. The slowdown is not really as bad as that, because the first three images rendered for the textures of the additional image planes do not have to be as accurate and don't have to show the same resolution as the final image. The generation of the composite image is computationally a little bit more expensive, due to the additional accumulation of the values from the additional image planes.

Despite the higher computational requirements the results achieved with multiple views added to the scene are well worth the effort.

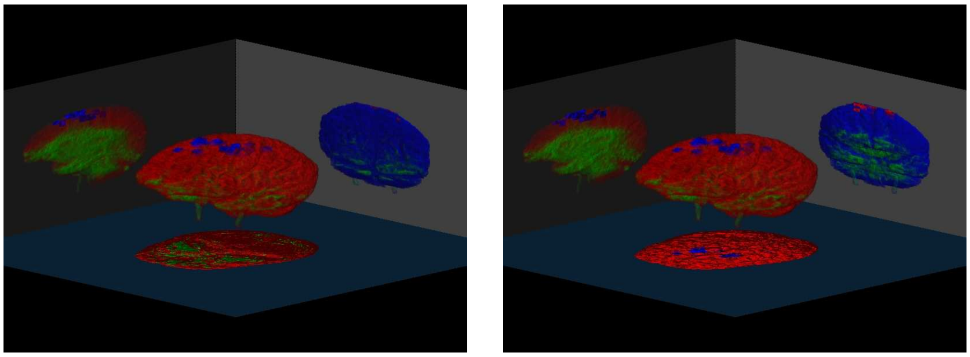
5.4 Magic Mirrors - simultaneous visualization of different information

Magic mirrors is a further technique developed in the scope of this work, which improves the idea of multiple-view images. This technique enhances the visualization by depicting different kinds of information in a single visualization. As the multiple views combine different images (of one object) into one visualization, this ability can also be used to provide different information with each of these images.

During this project, three different kinds of enhancements to the multiple views have been implemented:

- The use of different transfer functions for each additional image plane
- *Front projecting* vs. *back projecting*
- *Contour tracing*

In the first step we wanted to take advantage of the fact that three different additional image planes are available in the composite image, adding three mappings of the objects to these image planes. To increase the amount of information provided by these three additional images, we added the possibility to assign to each image plane separate transfer functions (both for the anatomical and the functional data) to get different results at each of the planes and combine them afterwards with the resulting representation of the data sets. An example of such a visualization with three different kinds of transfer functions used for producing the images displayed on the additional image planes is shown in Figure 5.5 (a). The image on the lower plane was produced using transfer functions with much higher opacity, therefore the resulting image is very similar to surface rendering. On the left upper image plane the opacity for the anatomical transfer function has been chosen to be very low, this gives good insight by producing a transparent anatomical hull for the functional data. In the right upper image plane not the opacities, but the colors for the transfer functions have been changed, resulting in a completely different visualization of the same data (this was just done to, to show the wide range of different capabilities of this visualization technique).



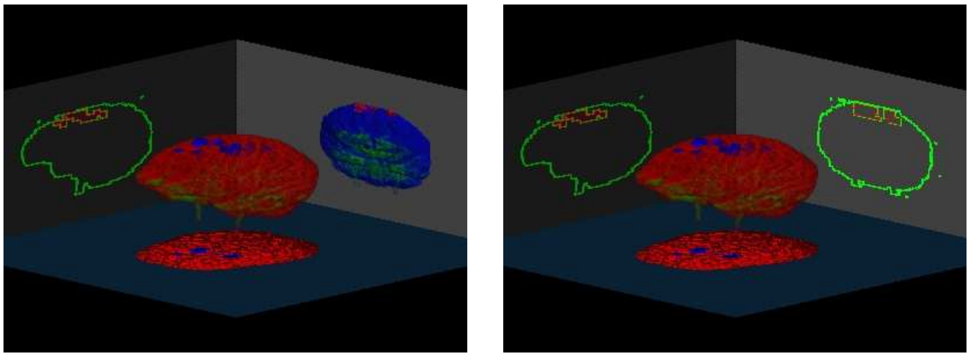
(a) (b)

Figure 5.5: *Magic mirrors*: using different transfer functions for each additional image plane (a), *front projecting* (b).

A second approach to enhance the additional image planes with special features is presented in Figure 5.5 (b). In this case, the images displayed on the additional image planes are not produced by ray casting using the viewing direction from behind the image planes, as in the multiple views technique described in Section 5.3, referred to as *back projecting* here. In this case, the viewing direction was from the opposite direction, from in front of the image planes. This is referred to as *front projecting* here.

The third extension of the additional image planes is a technique called *contour tracing*. Using this technique, any of the three additional image planes can display a contour image of the given data sets. The contour image is calculated as follows:

1. For each pixel of the image on the image plane a ray is cast through the transfer-function volume. This casting can return three possible results:



(a) (b)

Figure 5.6: *Magic mirrors*: examples using the concept of *contour tracing*.

(a) no data encountered along the whole ray, or only "NULL-transfer functions" found

(b) data and anatomical transfer functions encountered, but no functional transfer function together with data values

(c) at least once a functional transfer function with an active data value at the same position encountered.

These results are generated very similar to a MIP (Maximum Intensity Projecting) approach, as in the first case only indices 0 are encountered, anatomical data is marked with index 1, and functional data is assigned index 2. This is why only the maximum along the viewing ray has to be computed.

2. These casting results are stored in an *active transfer functions image.*

3. In a post-processing step, the borders between the different regions of the three possible results are calculated using an edge detection filter and marked in the resulting texture, that is displayed on the according image plane.

This visualization technique seems to be very promising, as it produces images that provide enough information for a sufficiently good understanding of the spatial threedimensional relations, even when only viewed as a still image. By animating this technique better results can be achieved, proving the usefulness of such a rather twodimensional visualization added to the scene on the magic mirrors. Two example visualizations using *contour tracing* are presented in Figure 5.6 (a) and (b).

One can also think of many other possible enhancements. For example a slice parallel to one of the additional image planes could be displayed in the 3D volume, and the corresponding gray value image of this slice could be displayed on the corresponding image plane. This technique might be useful for interactive browsing through the volume data, to more accurately locate the activation centers in the middle of the data. The image of the slice displayed on the additional image plane could also be a image rendered using transfer functions, or an image showing only segmented parts of the data.

The results section of this thesis (see Section 7.4) provides more visualizations in higher quality produced with magic mirrors. Additionally references to animations produced with this technique can be found there, and some statistics are given concerning computational costs. The calculation cost of magic mirrors is about the same as for multiple views (presented in Section 5.3).

Chapter 6

Implementation

The implementation of the methods described in Chapter 5 is was done on an IBM PC compatible platform. A 300MHz Intel Pentium 2 hardware environment featuring 256 MByte of memory is used. The developed software system was also tested on different other compatible hardware combinations, for example on a dual 400MHz Intel Pentium 2 environment equipped with 512 MByte of memory.

The operating system installed on the machines was Windows NT Workstation version 4.0. Microsoft Visual C++ 5.0 was used as the development environment.

6.1 General description

The software prototype, which has been developed during the implementation phase of this project works task-file driven. A special parser interprets a textual task file, which defines all the parameters of the program according to user specification.

This approach was chosen, because as a starting point for this software prototype a structure from another medical volume visualization system was used, namely the object oriented structure of the $\mathrm{V^{is}M^{ed}}$ [30] visualization system. It was adapted and further parts (additional and specialized classes) have been added according to the needs of the project.

The changed or added classes of the object hierarchy are described briefly in the following section. Note that for the sake of simplicity a subdivision into private and protected members in this descriptions has been omitted. Furthermore only the most important members and methods of each class are listed and will be described in more detail.

Before describing the different new or changed classes, a short overview of the capabilities concerning the user interaction by editing the task file of the program is given.

6.1.1 User interaction via the task file

The user can specify via the task file (which is defined by an command line argument to the program) most parameters of the visualization process. The following enumeration shows the capabilities, but does not explain any syntax topics:

- the locations and orientations of both, the anatomical and the functional data sets and their corresponding coordinate systems can be defined in a general reference coordinate system (this simplifies the task of registering the data sets)
- the location and orientation of the image plane according to the same reference coordinate system
- the resolution of the resulting image (size of the image plane) in pixels
- transfer functions for both data sets can be either read from files, or specified analytically by the use of different basis functions [4]
- transfer functions can be specified and saved to files for later usage
- the two data sets are loaded from specified files
- a filename for the output image is specified

- a light vector for the lightning of the scene is specified
- if *transfer-function volumes* are used, blocks can be cut out of the data, or planes can be specified to cut the data off at some point. This can either be done for the anatomical data set, or also for the functional data set
- the three image planes for the multiple views can be specified, and if so, a number of additional specifications concerning these image planes can be given:
  - filenames for saving the textures for the image planes are given
  - orientations, locations, and sizes of each image plane can be chosen
  - the background color for each of the multiple views can be specified
  - for each of the multiple views different transfer functions can be specified
  - contouring colors for contouring mode can be chosen
  - the projection mode (*front* or *back projecting*) can be chosen
- contouring for the different image planes can be switched on and off
- animations (not interactively) for rotations of the objects or changes in the transfer functions can be calculated

6.2 The object-oriented classes of the software prototype

In the next subsections, the main changes to already existing classes, as well as all the new classes for three of the four in the previous chapter described multimodal volume visualization methods will be presented briefly. Only the first described method, the multimodal ray casting and its corresponding class `CMMRayCaster` will be omitted here. The interface of the classes will be shown, and the functionality of the methods will be explained.

6.2.1 CRegister

The class `CRegister` implements registering of the two data sets (anatomical and functional) into one *transfer-function volume* (see section on `CTVolume`).

```
class CRegister
{
    VEC3 m_Location;
    MAT3 m_CoordSys;
    unsigned short m_ResolutionX, m_ResolutionY, m_ResolutionZ;
    VEC3 m_funcLocation;
    MAT3 m_funcCoordSys;
    unsigned short m_funcResolutionX, m_funcResolutionY,
                   m_funcResolutionZ;
    CTransferFunction* m_TF;
    CTransferFunction* m_funcTF;
    CData* m_data;
    CData* m_funcdata;
    CTFVolume* m_tfv;

    void RegisterData();
    void SetTFVolume(CTFVolume *tfv);
    void SetTF(CTransferFunction* what);
    void SetLocation(float x, float y, float z);
    void SetData(CData *data);
    void SetCoordSys(VEC3 x, VEC3 y, VEC3 z);
    void SetfuncTF(CTransferFunction* what);
    void SetfuncLocation(float x, float y, float z);
    void SetfuncData(CData *data);
    void SetfuncCoordSys(VEC3 x, VEC3 y, VEC3 z);
};
```

The data members

- `VEC3 m_Location` and `VEC3 m_funcLocation` store the locations of the two data sets that shall be registered in terms of world coordinates
- `MAT3 m_CoordSys` and `MAT3 m_funcCoordSys` respectively store the coordinate systems of the two data sets
- the six `*Resolution*`-variables give the X-, Y-, and Z-resolutions of the two data sets
- `m_TF` and `m_funcTF` are used for the two transfer functions used for the volume visualization of the two registered data sets
- `m_data` and `m_funcdata` hold the two data sets
- `m_tfv` is the transfer-function volume, in which the registered transfer functions are written

The methods

- `RegisterData()` is responsible for the registering, after all the required parameters have been set by calling all the corresponding methods provided by this class
- all the other methods provided above are responsible for the setting of the corresponding data members mentioned

There exist also other methods (and data members) in the class `CRegister` that are not shown here. For example a set of methods to perform coordinate systems transformations (needed for the registering), or transfer function handling methods.

6.2.2 CMirror

The class `CMirror` calculates the image, which is displayed on one of the image planes. This can be done by standard ray casting of the `CTFVolume` object with the viewing direction either from behind the image plane or from the opposite direction. The image displayed on the image plane can also be calculated as a contour image of the two volumes (again by using the `CTFVolume` object).

```
class CMirror
{
    VEC3 m_IPOrientationX, m_IPOrientationY, m_IPOrientationZ;
    int m_IPXres, m_IPYres;
    char m_FileName[256];
    float m_IPLocation[3];
    CRaycaster * m_RC;
    RGB_struct * m_Image;
    char * m_indexbuffer;
    float m_Red, m_Green, m_Blue;

    void SetRC(CRaycaster *rc);
    void SetLocation(float x, float y, float z);
    void SetOrientation(VEC3 xvec, VEC3 yvec, VEC3 zvec);
    void SetResolution(int xres, int yres);
    void SetContourColorNumber(int maxIndex);
    int SetContourColor(char index, RGB_struct color);
    void Save(char *filename);
    void RenderImage(float r, float g, float b);
    void RenderContourImage(float r, float g, float b);
    int AddColorAndOpacity(CRay *ray, float* r, float* g,
                           float* b, float *opacity);
    int IntersectPlaneRay(CRay ray, float* r, float* g,
                          float* b);
};
```

The data members

- `m_IPLocation` and `m_IPOrientation[XYZ]` store the location and coordinate system of the corresponding image plane
- `m_IPXres` and `m_IPYres` store the resolutions of the image on the plane in pixel
- `m_FileName` holds the filename and complete directory path to the file, the texture on the image plane is written to
- `m_RC` is the `CRayCaster` object, used by `CMirror`
- `m_Image` holds the rendered image, which shall be displayed on the plane
- `m_indexbuffer` holds for each image pixel on the plane the index of the active transfer function along the ray. This index is calculated during the contour tracing step.
- `m_Red`, `m_Green`, and `m_Blue` store the background color of the plane for the parts of the image, where no data is rendered, in RGB-values

The methods

- `Save(char *filename)` saves the image to the file with the provided `filename`
- `RenderImage(float r, float g, float b)` renders (utilizing `m_RC`) the image, which is to be displayed on the corresponding plane
- `RenderContourImage(float r, float g, float b)` renders a contour image to be displayed on the image plane
- `AddColorAndOpacity(...)` and `IntersectPlaneRay(...)` are used by the last rendering pass, to evaluate the color value for the images on the image plane for each ray intersecting the plane

- all the other methods provided above are responsible for setting the corresponding data members mentioned

There are also other methods (and data members) in the class `CMirror` that are not listed here. For example several other data members, determining the size of the texture on the image plane and the spatial size of the plane in the scene, are implemented.

6.2.3 CTFVolume

The class `CTFVolume` implements the concept of *transfer-function volumes* (see Section 5.2 for details). In the following the interface is presented and the different data members and methods available are explained.

```
class CTFVolume
{
    CTransferFunction *m_TF[TFV_MAXANZTFS];
    char* m_volume;
    int m_x, m_y, m_z;

    CTFVolume(int x, int y, int z);
    void SetVolumeCuttingPlane(char index, float d, float rx,
                               float ry, float rz);
    int SetVolumeBlockEntry(char tf, int xi, int yi, int zi,
                            int xo, int yo, int zo);
    float Evaluate(float d, int x, int y, int z);
    float EvaluateHue(float d, int x, int y, int z);
    void SetVolumeEntry(char tf, int x, int y, int z);
    void SetTF(int index, CTransferFunction *tf);
    char EvaluateIndex(float d, int x, int y, int z);
};
```

The data members

- `CTransferFunction *m_TF[TFV_MAXANZTFS]` stores all the active transfer functions, that have been registered for usage. By using the index of each transfer function it can be accessed. For our case only three different indices were used, but more would be possible.

- `char* m_volume` stores for each voxel of the anatomical data set the index of the transfer function that is to be used

- `m_x`, `m_y`, and `m_z` are set to the x-, y-, and z-resolution of the anatomical data set respectively

The methods

- `CTFVolume(int x, int y, int z)` initializes the transfer-function volume with the correct resolutions to be used

- `SetVolumeCuttingPlane(...)` and `SetVolumeBlockEntry(...)` set a complete half space or a complete block of the transfer-function volume to the index provided in the first parameter

- `Evaluate(...)` and `EvaluateHue(...)` get the opacity and the hue of the color at the position provided (by calling the correct transfer function stored in the transfer-function volume)

- `SetVolumeEntry(...)` sets one single voxel to the desired transfer function index

- `SetTF(int index, CTransferFunction *tf)` registers a transfer function for the later use by providing an index

- `EvaluateIndex(...)` can be used to get for each position within the volume the correct index of the transfer function, that should be used there (this method is used for the *contour tracing* approach described in Section 5.4)

6.2.4 Adaption of CRaycaster

In the class `CRaycaster` (as provided by the V^{is}M^{ed} [30] software system) many changes have been done during the implementation phase of this project. The major changes are depicted and discussed below, all the other changes and the original implementation, which existed already before this project started, are not described here.

The class `CRaycaster` implements the standard ray-casting algorithm described in detail in Section 3.1.3 of this thesis. The major changes, realized during the development of the multimodal volume visualization techniques, are the following:

- The handling of transfer-function volumes had to be included, so many methods to initialize, change, and get the state of the transfer functions had to be implemented. A data member `CTFVolume *m_TFV` was included to store the associated transfer-function volume to the data to be ray casted.

- In addition to the already existing method `RGB_struct *RenderNewImage(int *ready)`, which is called to render a new image of the scene after all the settings for the ray-casting process have been defined, three new methods were included:

```
int RenderNewMirrorImage(int *ready, float red, float green,
                         float blue, RGB_struct *imagebuffer);
int RenderNewContourImage(int *ready, int maxIndex,
                          char *indexbuffer);
RGB_struct *RenderNewMVImage(int *ready, CMirror *mirror1,
                         CMirror *mirror2, CMirror *mirror3);
```

  - The first one, `RenderNewMirrorImage` is used to calculate the texture for one of the additional image planes. The variables `red`, `green`, and `blue` define the color of the background of the image plane set by the user for correct accumulation of the rendered values with the background. The variable `imagebuffer` stores the calculated texture (image) for the image plane.

– The second method, `RenderNewContourImage` calculates the image providing the index of the active transfer function (maximum) for each pixel (as described in Section 5.4). This information is later on used by the corresponding `CMirror` object to calculate the contour image as a texture for the image plane. `maxIndex` is used for early ray termination. When the maximum index is reached, the index image is returned in the variable `indexbuffer`.

– The last method presented, `RenderNewMVImage` renders the composite image using the three additional image planes, which are provided in the given `CMirror` objects. They are combined with the final pass of ray casting the 3D representation of the brain using the user defined viewing direction.

6.2.5 Adaption of CRayVolume

The class `CRayVolume` implements the handling of the volumes, when considered together with the rays of the ray-casting process. Transformations, interpolations, spatial relationships of ray and data volume, transfer function handling for each step of the ray-casting algorithm, are all realized in this class.

In the class `CRayVolume` also many adaptions and changes have been done during the development of the new techniques described in detail in Chapter 5. The most important ones are enumerated below.

- Similar to the situation of class `CRaycaster` also in the class `CRayVolume` the handling of transfer-function volumes had to be included. Basically this resulted in similar changes as described in Section 6.2.4 for `CRaycaster`.

- Other major changes concerning `CRayVolume` have also some similarities to the adaptions done in the class `CRaycaster`. They concern the compositing for the textures to be displayed on the additional image planes and the compositing for the composite image rendered together with these image planes. Some modified methods are:

```
void GetColor(CRay *ray, float* r, float* g, float* b);
void Compose(VEC3 p, VEC3 stepvec, int steps, float* r,
             float* g, float* b);
void GetColorAndOpacity(CRay *ray, float* r, float* g,
                        float* b, float* opacity);
void AdvancedCompose(VEC3 p, VEC3 stepvec, int steps,
            float* r, float* g, float* b, float* opacity);
char GetMaxIndex(CRay *ray, int maxIndex);
char MaxIndexCompose(VEC3 p, VEC3 stepvec, int steps,
                     int maxIndex);
```

  - `GetColor` and `Compose` were the already existing methods. `Compose` (called by the `CRaycaster` object), does the compositing by calling `GetColor` for each step along the ray, and returns the color for the corresponding pixel of the image in `r`, `g`, and `b`.
  - `GetColorAndOpacity` and `AdvancedCompose` were developed for the rendering of the image including all the three additional image planes. Therefore, the compositing step needed also to return the overall opacity encountered along the ray, to determine if a part of the image plane behind the data is still visible to some extend (alpha channel blending).
  - `GetMaxIndex` and `MaxIndexCompose` are the corresponding methods to do the compositing for the calculation of the transfer function images as described in Section 5.4 for the contour tracing approach (this is no compositing but rather a index determination along the ray as similar to a MIP).

6.2.6 Other classes used, which have not been adapted

Other classes used from the V$^{\text{is}}$M$^{\text{ed}}$ project [30], that have not been changed or adapted, were the following:

CBasisfunction – the basis functions are used to implement the transfer functions. Each transfer function consists of one or more basis functions, which together define the characteristics of a transfer function.

CData – implements the data handling for one data set. It stores the volume data and implements basic access to the data values.

CImageplane – this class handles the image planes, which hold the images to be calculated. It also defines the rays to be traced for the ray-casting algorithm by communicating with `CRay`.

CRay – implements a single ray used for the ray-casting algorithm.

CTransferfunction – this class implements the data structures and the methods needed to realize the mapping of data values to opacity and color hue via the specification of a transfer function.

Chapter 7

Results

In the following sections some results for each of the four in Chapter 5 discussed methods are presented. URLs are provided to give references to electronic versions of the images. At these internet pages animations can also be viewed. This is a great advantage over still images, as the capabilities of the different visualization techniques increase when adding the dimension of time. In the last section of this chapter some statistics of the obtained results are given. An outlook on how performance could be improved in order to make the methods interactive is discussed.

7.1 Multimodal ray casting

Some result images of the multimodal ray casting (see Section 5.1) are provided in the following figures. Unfortunately, due to the high computation and memory demands of this technique, all the images are rendered with rather small image size resolution.

In Figure 7.1 two results generated with this method are shown. The data set was not segmented, therefore not only the brain, but the complete human head, is rendered. Functional activities are shown in blue, the anatomical data was rendered using a red transfer function with relatively high opacity.

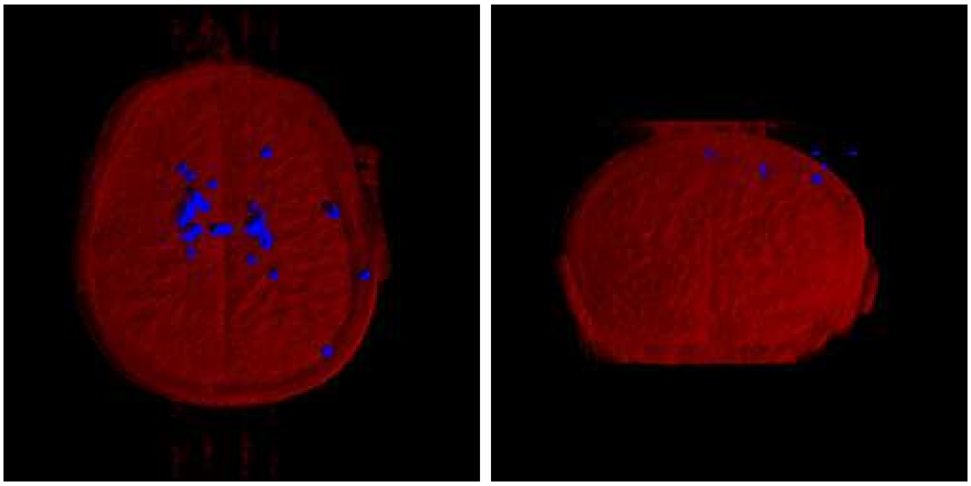


Figure 7.1: Two results generated with the multimodal ray-casting technique. Note, that the anatomical data was not segmented before the visualization.

In Figure 7.2 two examples, generated with the usage of different transfer functions are presented. Some more images, obtained by applying the multimodal ray-casting technique can be found at `http://www.cg.tuwien.ac.at/research/vis/vismed/MM/Results/RC/`.

7.2 Transfer-function volumes

As the multimodal ray casting turned out to be very costly, *transfer-function volumes* were developed (see Section 5.2).

Some of the obtained results of this method are presented in the following figures. The electronic version of all these images and further results produced by using *transfer-function volumes* are available at `http://www.cg.tuwien.ac.at/research/vis/vismed/MM/Results/TFV/`.

In Figure 7.3 two views of the same data sets are shown, generated by the use of different transfer functions for the anatomical data. In Figure 7.4 again two different transfer functions for the anatomical data have been used

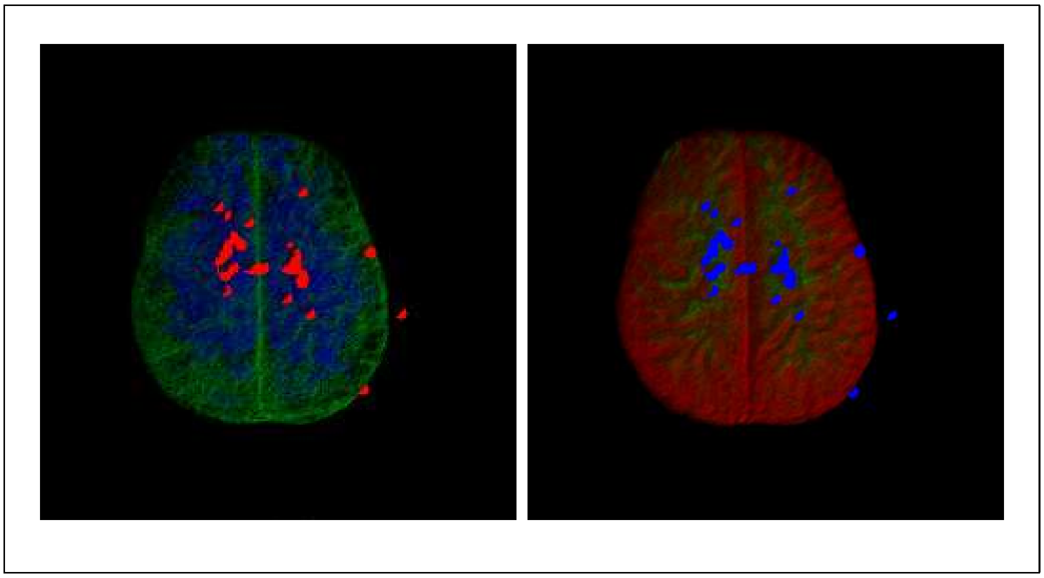


Figure 7.2: Two images generated with multimodal ray casting using two different sets of transfer functions.

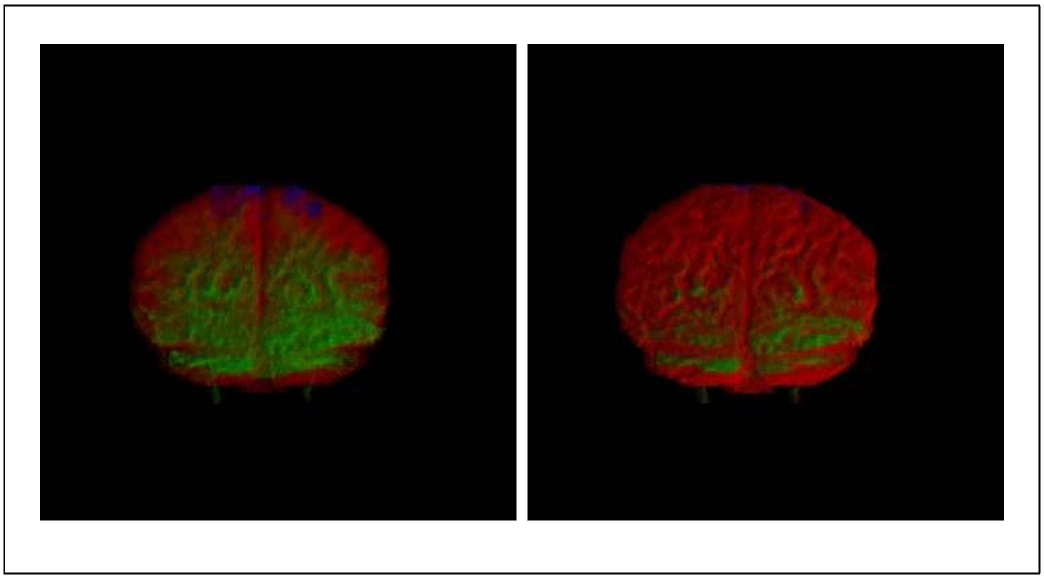
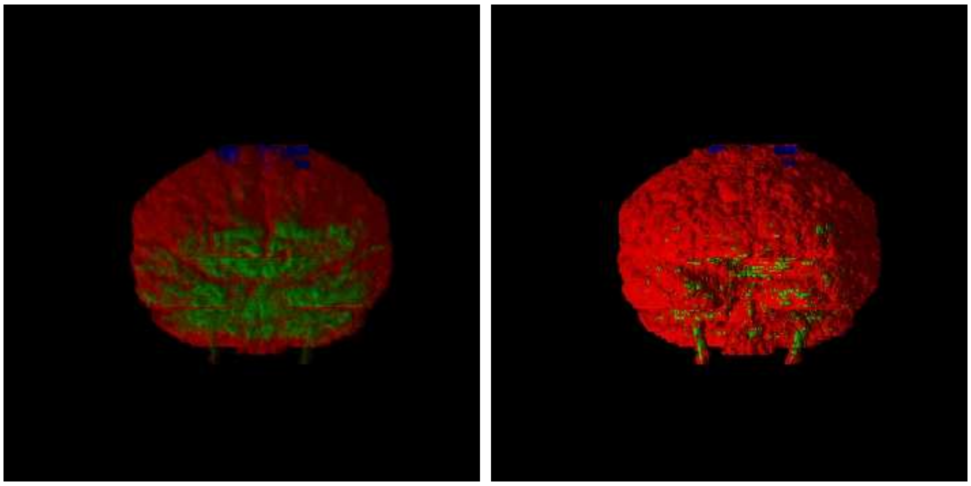


Figure 7.3: Two examples viewed from the back of the brain, using different transfer functions for the anatomical data.

on the same view, this time viewed from the front. In 7.4 (b) the opacity of the anatomical transfer function is set to the maximum. By comparing Figure 7.4 (a) with Figure 7.4 (b) it can be clearly seen, that parts of the functional activities are not on the surface of the brain, but rather a little bit inside.



(a) (b)

Figure 7.4: Two examples viewed from the front of the brain, using different transfer functions for the anatomical data: in (b) the opacity was set to the maximum. By looking at both of the two results next to each other, more spatial information on the location of the functional activities can be gained.

In Figure 7.5 a sagittal view on the same data as before is presented. Due to the higher resolution of this image the advantages of using transfer-function volumes can be clearly seen. The functional data is only represented by taking a different transfer function for these regions. The underlying anatomical data determines where data is displayed. Therefore the functional data is only visualized in regions, where there is also anatomical data, and no erroneous outliers are visible.

In Figure 7.6 (a) the segmented brain with the functional activations is visualized using transfer-function volumes and viewed axial from above the

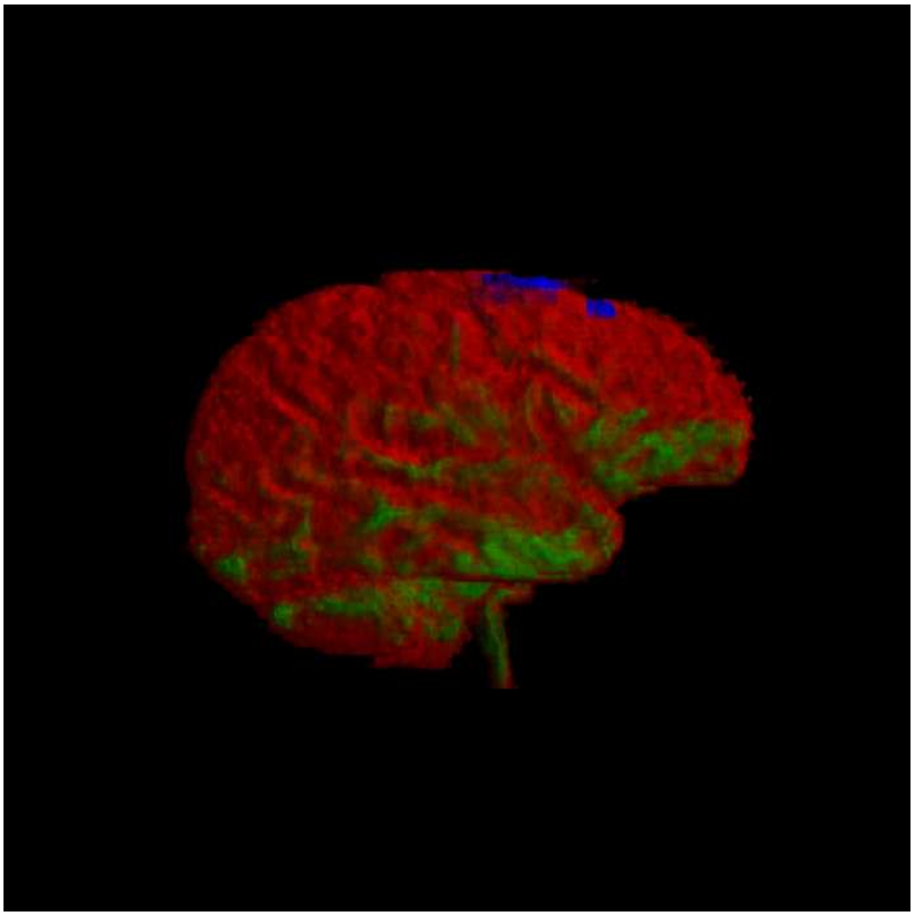


Figure 7.5: Sagittal view of the segmented human brain generated with the use of transfer-function volumes.

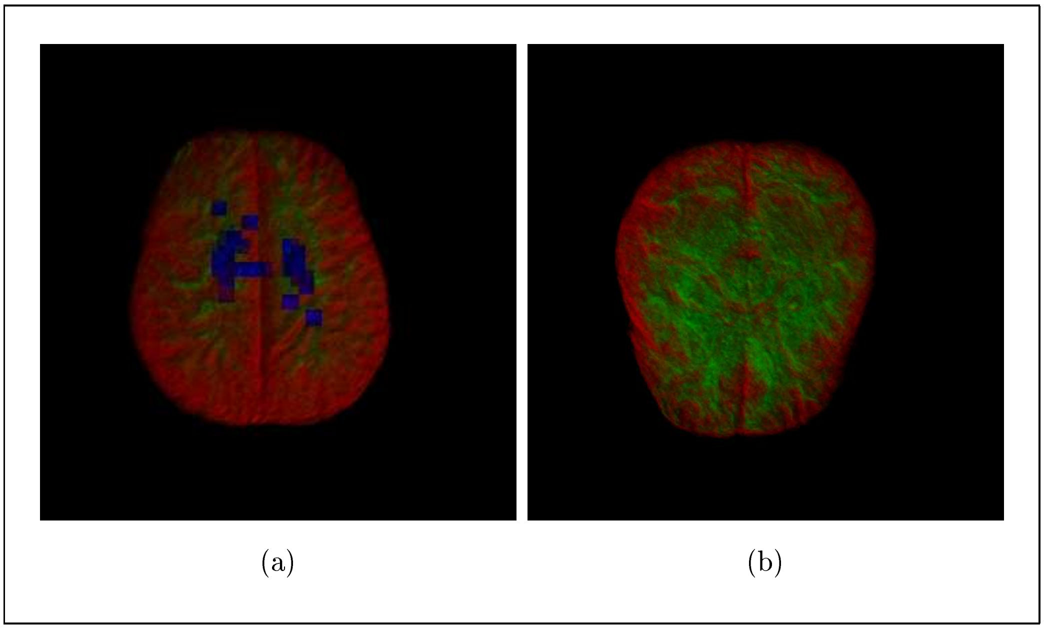


Figure 7.6: Two examples of a segmented brain visualized with functional activities using transfer-function volumes. (a) View from above and (b) view from below the brain for comparison of activation locations.

brain. In Figure 7.6 (b) the viewing direction was chosen oriented to the opposite, to generate a view from below the brain. One can easily see, that the functional activation occurs in the upper part of the human brain for the task of finger tipping as described in Chapter 1.

7.3 Multiple Views

In Section 5.3 multiple views, an extension to transfer-function volumes, is dealt with.

In this section, some examples of visualizations obtained with this method are shown, more images as well as also animations using this technique can be viewed at `http://www.cg.tuwien.ac.at/research/vis/vismed/MM/Results/MVI/`.

In Figure 7.7 an image generated with the technique of using multiple views is presented. It can be clearly seen, that adding more views generated

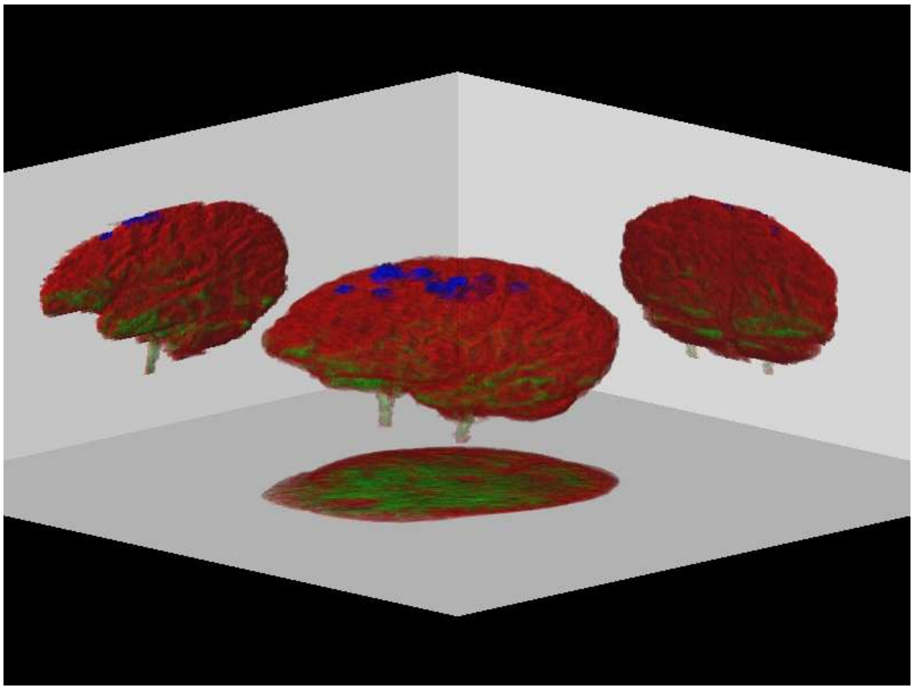


Figure 7.7: Multiple views improve the understanding of the three-dimensional structure as well as the spatial relationship of the anatomical and functional data sets.

from different viewing directions improves the understanding of the three-dimensional structure of the volumetric data, even when only a still image is investigated. Further information about the spatial relationship of multi-modal data sets can be gained, too.

When using transfer functions with low opacities to visualize the volumetric data sets (volume data is rather transparent), the problem of perceiving structural information and spatial relationships is even bigger, than in the case of higher opacities or surface renderings. In this case the use of multiple views added to the resulting image as done for example in Figure 7.8 is very valuable and proves to be useful.

Also in the cases of performing cuts of the anatomical data to reveal underlying functional data (see Figure 7.9), as well as occlusion problems

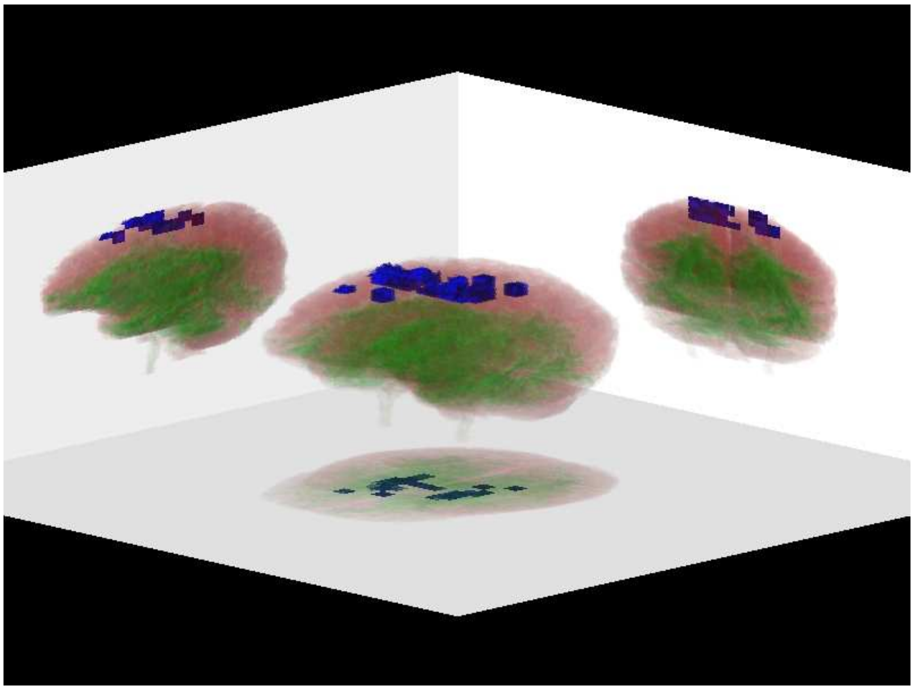


Figure 7.8: Multiple views are especially useful in the case of rather transparent representations of the volumes.

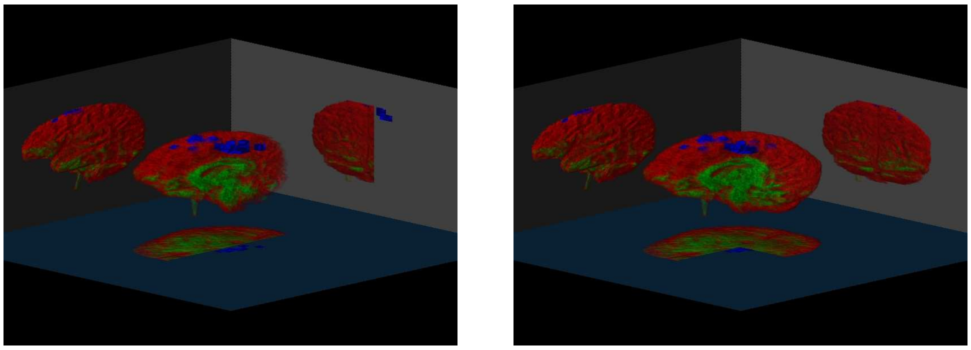


Figure 7.9: Two images featuring multimodal data sets. Parts of the anatomical data have been cut away (by the use of transfer-function volumes) to reveal the underlying functional data. The multiple views help to understand the structural relationship of functional and anatomical data.

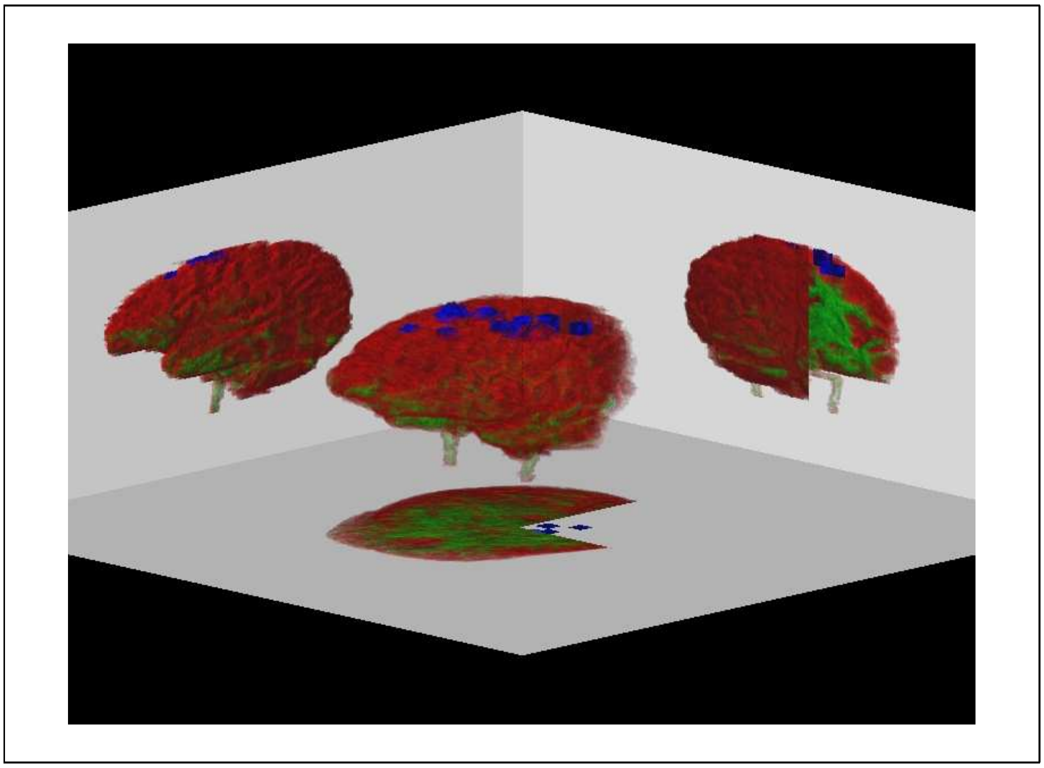


Figure 7.10: The problem of occlusion is solved satisfactorily by the usage of image planes to provide multiple views from different viewing directions. Although the cut away of anatomical data is not visible from the primary viewing direction, the spatial relationship of the functional data at this region becomes obvious.

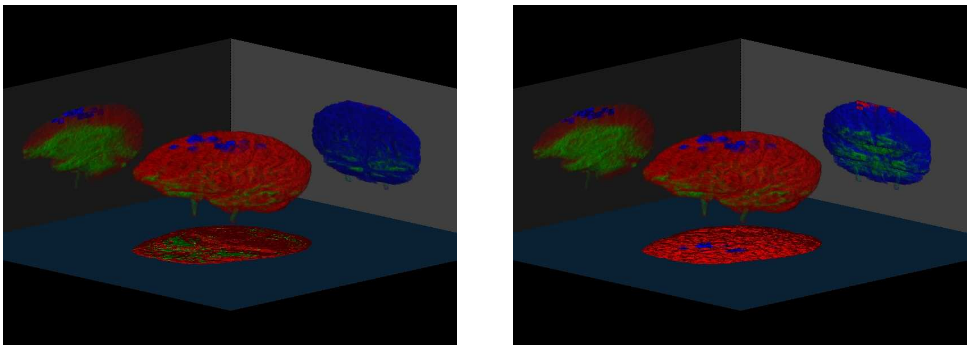
resulting from these cuts (see Figure 7.10), the employed multiple views technique improves visual impression and understanding abilities.

7.4 Magic Mirrors

In Section 5.4 magic mirrors have been discussed in detail.

Some sample images are presented in the following figures, more results provided electronically (images and animations) can be viewed at `http://www.cg.tuwien.ac.at/research/vis/vismed/MM/Results/MM/`.

In Figure 7.11 two different approaches described in Section 5.4, namely



(a) (b)

Figure 7.11: *Back projecting* (a) vs. *front projecting* (b).

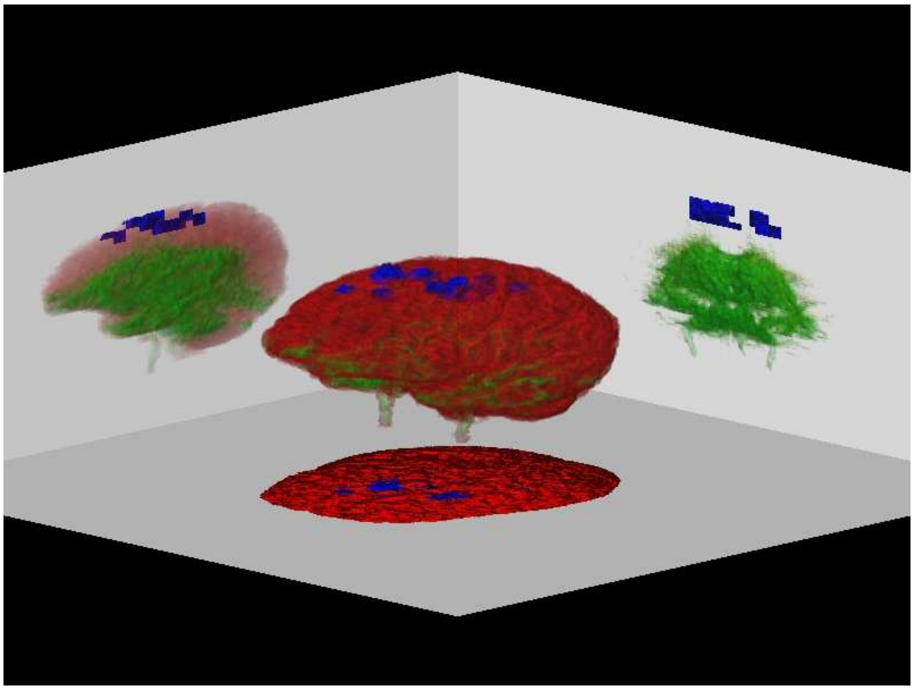


Figure 7.12: For the images on each of the additional image planes different transfer functions can be applied, to maximize information provided by the visualization. By applying different transfer functions, care has to be taken, that useful and non-irritating choices are made.

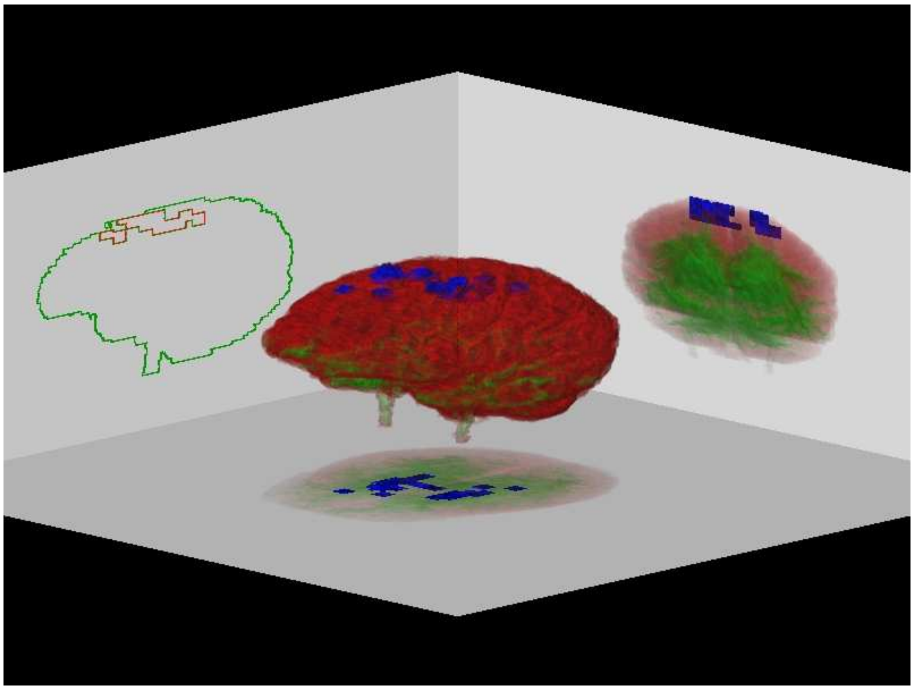


Figure 7.13: Additional information provided by displaying a texture generated by *contour tracing* on one of the image planes.

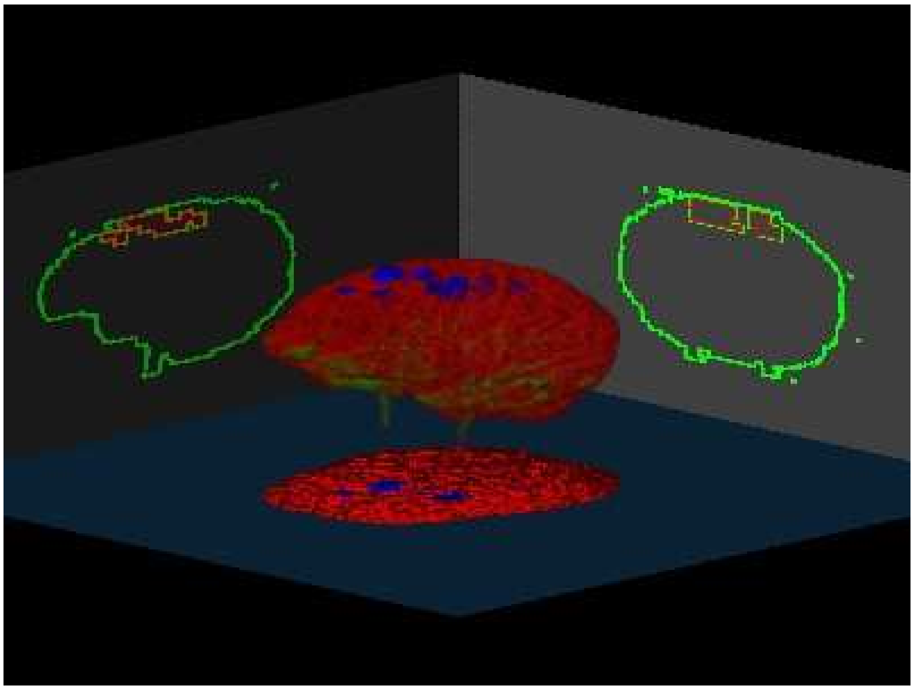


Figure 7.14: Another example of displaying contour-traced images on the image planes behind the volume data.

back projecting (a) vs. *front projecting* (b) are illustrated. In Figure 7.12, the capabilities to support different transfer functions are presented. The transfer functions on the two image planes behind are chosen in a way, that it can clearly be observed, whether the activated regions are in the gray matter of the brain (red color) or partly also in the white matter (green color). On the image plane below the objects *front projecting* is applied.

In Figure 7.13 and Figure 7.14 two images, using also the *contour tracing* approach are given.

7.5 Statistics and Future Work

A typical image, generated for the animations provided at `http://www.cg.tuwien.ac.at/research/vis/vismed/MM/Results/`, with an image resolution of 400 by 300 pixels needs approximately 3 to 4 minutes of calculation time on a Intel Pentium 2, 300 MHz. The calculation times depend on the transfer functions used for the different additional image planes. Using very opaque transfer functions is computationally much less demanding than rendering very transparent representations. This is due to early ray termination, when the maximum opacity is reached in the compositing step. Images with a resolution of 800 by 600 generated for this result section took between 12 and 18 minutes to calculate. With the data sets used in this case, as described in Chapter 1, the registration in the pre-processing step took less than 6 seconds. Therefore, speeding up the ray-casting process, which uses most of the computation time, has to be addressed.

In the future it is planned to use a faster ray-casting system, which will be *shear warp based* [31]. With using this other approach, image generation rates of about 3-10 seconds could be achieved, providing a considerable acceleration of the presented visualization techniques.

Concerning memory consumption, due to the use of transfer-function volumes for registering the data sets, the presented techniques (except for the very first one), are not highly memory demanding. Except for the two volume data sets (anatomical and functional data), only the transfer-function volume requires additionally a considerably amount of memory. In our implementation, 1 byte was used for each corresponding voxel in the anatomical data set. Compared to the two bytes per voxel in the anatomical data set itself, this is a 50% increase of memory usage.

Additional savings can be achieved, by restricting the transfer-function volume to use only 4 or 8 transfer functions simultaneously, and not to allow for 256 as was done in this implementation. This would further decrease memory needs, and not restrict the presented visualization techniques, as only 3 different transfer functions were employed.

In the future not only speed issues will have to be dealt with, also the usability of these techniques will have to be investigated by showing them to doctors, and asking them for feedback to evaluate the usability.

Chapter 8

Conclusions

This thesis discusses multimodal volume visualization in the medical field, which is an emerging and important area in recent medical applications. Volume visualization techniques in the medical context are especially of urgent need in treatment planning, early detection of health problems and surgery training.

Many problems of multimodal volume visualization in general as well as also in the special case of *anatomical-functional multimodal volume visualization of the human brain* are discussed. A set of arguments in favor of the use of this kind of visualization techniques in the medical context are presented.

A list of different currently available volume visualization systems (commercial and non-commercial ones) is discussed. The capabilities of these systems with respect to the given task of multimodal volume visualization are investigated.

As non of the given systems provided the functionality asked for in our case, four new different techniques with different approaches to the problem of anatomical-functional multimodal volume visualization are developed.

The first method, *multimodal ray casting* (see Section 5.1 for further details), works on the two volumes of the anatomical and functional data sets separately.

Due to some disadvantages of the first technique the second technique uses a completely different approach. *Transfer function volumes* (see Section 5.2) allow the simultaneous visualization of both data sets. The results seem to be quite promising.

In Section 5.3 the concept of *multiple views* implements a further useful enhancement of multimodal volume visualization by placing three additional image planes behind and below the visualized objects, to generate different views of the object simultaneously.

Finally, *Magic Mirrors* depict different information on the three additional image planes available. It is a useful enhancement of the previously developed technique, and will be further extended in the future.

The users, which were presented the results achieved by the different techniques developed during the work on this thesis claimed to be very pleased by the obtained results. They already provided us with feedback, concerning the direction of further desired improvements if possible. These could include the following:

- accelerating the techniques by using an optimized ray-casting algorithm (only standard ray casting is used here)
- finding further possibilities to enhance the information provided on the *Magic Mirrors*
- integrate the multimodal volume visualization of more than two modalities at a time (for example in the field of human brain visualization, anatomical data provided by MRI could be presented together with two or three functional data sets acquired by different modalities, eg. fMRI, EEG, SPECT, or others)

Chapter 9

Multiple Views and Magic Mirrors - Multimodal Volume Visualization of the Human Brain

Abstract

Multimodal visualization of functional and anatomical data of the human brain is an important field in medical volume visualization. The aim of this application is to provide the user with information on the location of functional activations in the different regions of the brain. When the approach of direct volume rendering is chosen, the visual impression of details usually suffers from accumulating the image from colors and opacities derived from the data set. We present extensions to overcome this problem: *Transfer function volumes* are used for the highlighting of activated regions. *Multiple views* simplify the task of localizing these parts of the brain. Complementary information is brought into the visualization by *Magic Mirrors* in order to enhance the comprehensive view of the multimodal arrangement of volume data sets.

keywords: Volume Visualization, Multimodality, fMRI

9.1 Introduction

Over the last decade, the use of digital modalities for clinical diagnosis and therapy planning has dramatically increased. New developmens in threedimensional volume acquisition techniques such as Spiral CT (*Computed Tomography*) scanning, faster and better pulse sequences for MRI (*Magnetic Resonace Imaging*) and improved functional imaging provided by modalities such as PET (*Positron Emmisson Tomography*), SPECT (*Single Photon Emission Computed Tomograph*), EEG (*Electro EncephaloGraphy*), MEG (*Magnetic EncephaloGraphy*), and fMRI (*functional Magetic Resonace Imaging*) are creating a rapidly increasing demand for integrated multimodality 3D visulization. It aims at efficiently presenting information from multiple modalities, in order to obtain a comprehensive picture of clinical cases. Therefore, depending on the clinical requirements, it might often not be sufficient to consider vascular, anatomical and functional information seperatly but to superimpose images from different modalities.

Functional MR (magnetic resonace) imaging is used to determine regions of the human brain, which are responding to sensory stimulation, motor or cognitive activity or emotional states. Neuroimaging experts investigate such scans usually by employing their talent to visually extract the characteristics of activated regions from the scanned volumes. This exploration task is usually done on twodimensional slices of the human brain, where the functionally activated regions are highlighted in some special way (see Fig. 9.1, top row). Viewing several slices at once (side by side) is refered to as the *image galleries* approach. This way of investigating the data sets provides detailed 2D anatomical context information. Nevertheless, this way of dealing with multiple data sets does not aid the user in forming a threedimensional model of the underlying spacial relationship of functional activations and the anatomical geometry. Other approaches try to increase the amount

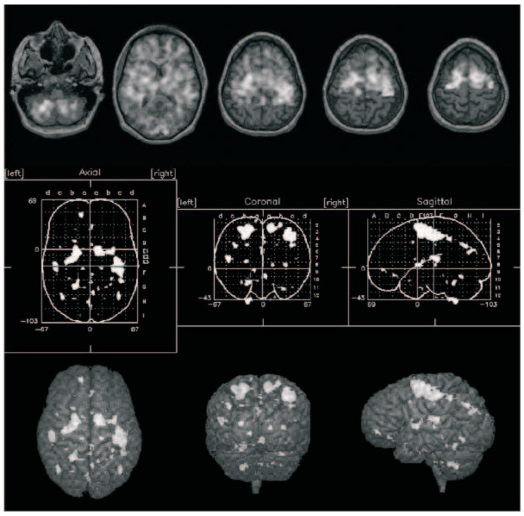


Figure 9.1: Conventional approaches to multimodal visualization [49]. Top, selected MRI slices are merged with functional data by pixel interleaving [50]. Middle, the maximum intensities of the data are projected onto axial, coronal and sagital planes bearing brain outlines and millimetric Talairach coordinates [56]. Bottom, the maximum data intensities are projected along the line of sight onto the cortical surface.

of presented information by utilizing *maximum intesity projections* (MIP), mapped onto 2D reference grids with brain outlines [15] (see Fig. 9.1, middle and bottom row). This representations aid the task of side by side comparisons, but nevertheless provide just very rough information of the anatomical context. Therefore it is very hard for the user to mentally reconstruct the threedimensional structure of the patients brain.

Threedimensional surface renderings of the brain, where the functional activation is encoded by coloring the cerebral cortex provides a misleading anatomical context. Activated subcortical foci below the surface of the brain are obscured. Furthermore, all foci seem to lie on the surface of the cortex, as no depth cues are provided.

Direct volume rendering would be able to overcome this problem. Assigning transparencies to data ranges would allow the deeper structures to be visualized. Therefore direct volume rendering would be the most usable approach to the solution of this problem. Usually, however, anatomical detail has to be sacrificed, when employing transparencies [22], decreasing the visual appearence of details significantly.

For these reasons K. Rehm and other authors recently proposed, that the anatomy of the brain should not be visualized threedimensionally at all [49]. They use symbolic representations known from other fields of visualization (like fluid or information visualization) to encode the location and size of foci. In order to place the foci into anatomical context, they are projected onto the walls of a cube like environment together with different representations of the brain. We are going to utilize a similar approach of adding information to the visualization in Section 9.4.

Direct volume rendering still seems to be the most useful approach to clearly give visible hints on the location and size of activated regions. We are going to present some extensions to overcome the problem of reduced visibility in the following sections.

First of all, the given problem of our special case of combined visualization of functional and anatomical data of the human brain will be stated in detail. The problems arising when using standard direct volume visualization techniques will be discussed in Section 9.3. The first improvement using multiple views will be discussed in Section 9.4. Additional aids for the investigating user (Magic Mirrors) are introduced in Section 9.5. A results section will provide images and statistics to prove the usabilities of our techniques. Finally conclusions will be drawn.

9.2 The problems encountered in multimodal volume visualization

In this work a volume visualization technique for multimodal data of the human brain is presented, consisting of functional data of the human brain (measured by fMRI) and anatomical data of the brain (measured by different methods of MRI). Other types of multimodality are out of the scope of this paper and are not discussed. The functional data represents the activation potential of the brain at a certain point in time.

The test situation used was the following: A patient has to perform the repetitive action of thumb and index finger tapping. The activation potential of the brain is measured. This gives the functional context of the reaction to the action of the fingers. The fMRI scans are acquired during stimulation as well as under resting conditions. By calculating the differences of these images the activated regions of the brain can be obtained. A review of the physics of MRI (and fMRI) resides outside of the scope of this paper. For a complete description of MRI specialized literature [6, 52] should be consulted.

The problems encountered with multimodal volume visualization of anatomical/functional data of the human brain include the following:

- *the data sets have different resolutions*: the resolution of the functional data sets is usually significantly lower than that provided by the anatomical data. This is due to the fact, that the functional data has to be measured interactively. An anatomical MRI scan is acquired using more time.

- *the spatial extend of the data sets is different*: again, due to the time constraints in fMRI measuring. Usually only parts of the brain (for example only 5-7 slices) are scanned for the functional data set.

Therefore the data sets first have to be spatially aligned through *registration* [36]. Afterwards visualization is used to identify the regions of the brain, where the changes in the activation potential occured. This visualization step is the main point of the presented paper.

9.3 Transfer-function volumes for multimodal visualization

As mentioned above, the usage of shaded surface displays does not serve our purpose of visualizing multimodal data very well. First of all, inner structures of the brain (like white matter) are obscured by the covering surface. When activated regions are encoded by coloring the according parts of the surface, incorrect information is presented: activations seem to appear only on the cortical surface. The parts within the brain responding to activation are not visible at all. For these reasons we are not able to use any surface based visualization techniques like for instance *isosurface extraction by marching cubes* [35].

Direct Volume Rendering (or probabilistic rendering) is concerned with the visualization of inner structures. Data values are mapped into color and opacity values by the usage of transfer funtions. During image rendering physical properties of light transportation in non-opaque media are simulated. This approach serves our problem of multimodal volume visualization very well. Different methods in direct volume rendering are available:

- *Splatting* [58] is an object space approach, which composes the image by accumulating the contribution of every single voxel in the data set.
- *Ray casting* [33] is an image space approach, which accumulates contributions from voxels encountered along viewing rays. As this method approximates techniques used in photorealistic rendering (global illumination), the simulation of physically correct phenomena like refraction or reflection is possible. Having a special enhancement to our technique in mind (described in Section 9.4), raycasting is the technique of our choice.

For the chosen approach of ray casting, a special scheme for the specification of transfer functions had to be employed. *Transfer-function volumes* are used to specify different transfer functions for seperate regions of the data set to be visualized. A transfer-function volume is a threedimensional array storing an index to a set of transfer functions for every single voxel. This index is used to access a transfer function for mapping the data value of the voxel into the physical properties color and opacity. When the number of transfer functions to be used is limited, e.g. to 4 or 8, the indices can be packed more densly and significant amounts of storage space are saved.

Assigning transfer functions on a voxel per voxel basis has serveral advantages:

- A special transfer function is defined, which should be used to encode the activated regions of the anatomy. During the *registration* process, the index of this transfer function is assigned to all voxels which are at least partially intersected by a voxel of the functional data set with activation potential. During rendering of the anatomical data set activated regions are colored using this special transfer function and are highlighted.

- Using a "NULL" transfer function (mapping every data value to zero opacity) *volume sculpting* facilities can be realized quite easily. Setting regions of the transfer function volume to the index of this special transfer function, various "cut-away" possibilities can be realized, because voxels values mapped by "NULL" transfer functions simply do not appear in the visualization. These regions can be specified by cutting planes, block cut operators, or even on a voxel by voxel basis. The anatomical data set used in the scope of this work has been segmented in this way. A segmentation mask gained by manual segmentation of the anatomical data set specified the voxels of the non-brain regions of the head scan, which were rendered using a "NULL" transfer function. The original data set has not been modified in any way.

- When this approach has to be optimized for the calculation time of the raycasting step, the data volume can be converted into a color/opacity volume using the transfer-function volume in a preprocessing step.

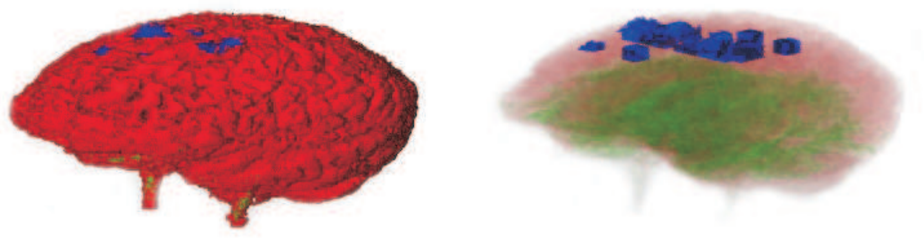


Figure 9.2: Different problems arise when using direct volume rendering. When surface rendering is employed, the regions of activation are hidden by the cerebral cortex (left hand side). Right, very low opacity allows deeper structures to be visualized. However, it is hard to tell, where the foci reside exactly.

> If rapid changes of the used transfer functions are desired (for instance when rendering several frames for an animated visualization), the transfer-function volume does not have to be changed at all. Simply the set of transfer functions referenced by the indices is modified.

Direct volume rendering is capable to overcome the problem of occlusion. Nevertheless, significant detail in image quality is sacrificed when employing low opacity contributions of the voxels. Figure 9.2 shows a comparison of the two extremal cases. When a transfer function mapping the voxel values of the anatomical data set into high opacity values is used, the visualization yields to a surface like representation (left hand side). When very low opacity values are used, hardly any anatomical context is visible (right hand side). It is not easy to describe the location of activated regions according to the visualizations in Fig. 9.2. A solution to overcome this problem is shown in the next section. To compensate for the loss of image details, additional information is brought into the visualization by enabling alternative views of the brain.

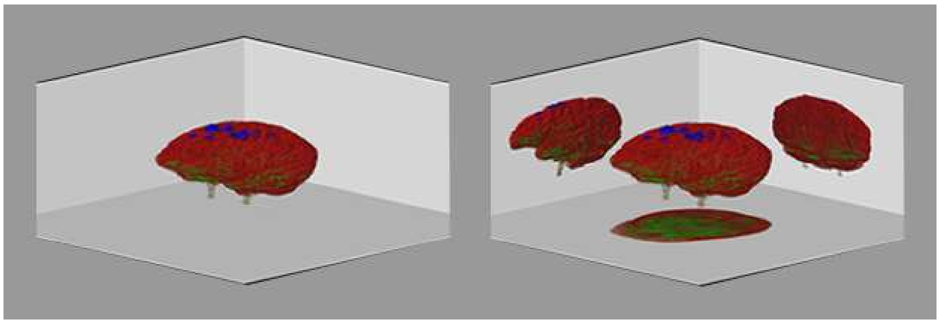


Figure 9.3: The principle of placing image planes as multiple views of the scene for enhancing volume visualization: without (left) and with (right) textures added.

9.4 Multiple Views - Enhancing the visual impression of direct volume renderings

From a single still image it is very hard to get the correct visual impression of depth. In the presented technique the image produced with the transfer-function volume is enhanced by adding three different views of the object to the final image. These multiple views are realized by positioning three image planes behind and below the object. Images displayed on this image planes are again generated by using the ray-casting technique [33]. These three image planes are added to the scene as shown in Fig. 9.3. The white parts on the three planes (see Fig. 9.3, left) depict the location, where the additional images are displayed as textures (Fig. 9.3, right). Each of these images has it's own set of camera parameters. In the example of Fig. 3 the viewpoints for the image planes are behind (below, respectivly) the planes. These image planes provide multiple views and are quite different in their properties from, e.g., mirrors or shadow walls. Placing a real mirror at the same location (an approach used by A. Kaufman [3]) would result in completely different images. Nevertheless, this different behaviour was aimed at for the further developments discussed in the next section.

The benefit gained from these multiple views is manifold:

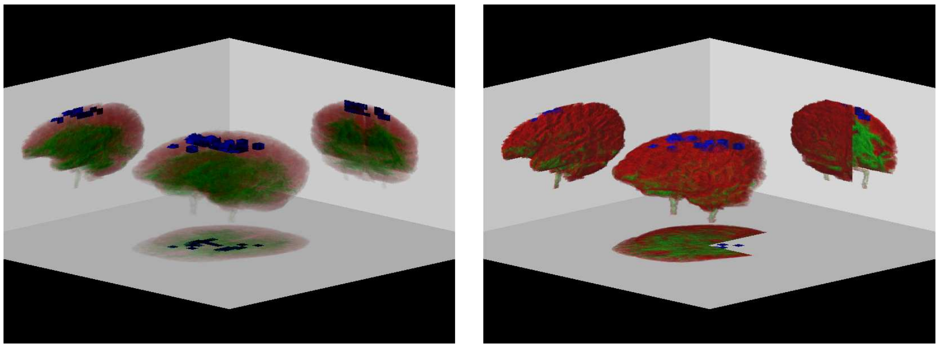


Figure 9.4: Left: Using multiple views, the exact localization of activations is also possible with transparent transfer functions; Right: Even occluded regions like the cutaway part become obvious with the help of multiple views

- The location of activated regions becomes obvious even in still images, as the threedimensional context of the two data sets can be recognized more easily when multiple views on the same scene are available. This is especially the case for highly transparent renderings (compare the image on the right hand side of Fig. 9.2 with Fig. 9.4, left hand side).
- The problem of occlusion is solved rather satisfactorily. In Fig. 9.4, right hand side, for example a block of the anatomical data has been cut out of the backside, but due to the multiple views provided by this technique, the locations of activations are still visible.

9.5 Magic Mirrors - Illustrating additional information

The benfits of multiple views of the brain generated by volume rendering can even be enhanced. As the employed image palnes of the multiple views do not show the real world physical behaviour of , e.g., reflecting planes, several quite different rendering techniques can be used to generate the textures. Leaving the domain of physical reflection completely, the extended version of the multiple views is called *Magic Mirrors*.

- *"Projection"* vs. *"Reflection"*: The multiple views textures used in the last section were created by raycasting the brain anatomy with an view point located behind (below, respectivly) the image planes. For this reason, the quasi-reflective property of the views was gained. It is easy to locate the view point for texture generation on the other side of the brain geometry. Utilizing this approach, projections of the brain in direction of the mirror can be generated. The main views onto the brain (front, top, right) can be used to enhance the visual impression of the threedimensional representation of the brain (Fig. 9.5, bottom mirror).

- Different transfer functions can be used for each of the Magic Mirrors. Various characteristics of the multimodal scenario can be emphasized in a single image. Fig. 9.5 demonstrates the relation of activated regions to white and gray brain tissue.

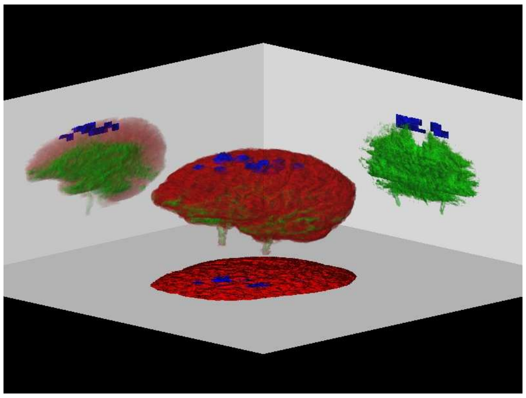


Figure 9.5: Magic mirrors employing different transfer functions and *"Projecting"*

- Additional views to the anatomy created by non-direct volume rendering techniques can be used to provide complementary information. In Fig. 9.6 a 3D *"contour map"* of the brain and activated regions (generated by the ray-tracing approach) was mapped onto the Magic Mirror at the right hand side of the scene. The usage of other techniques like *maximum intensity projections*, *orthonormal cuts*, *color encoded cuts*, or *segmentation maps* as Magic Mirrors is also possible.

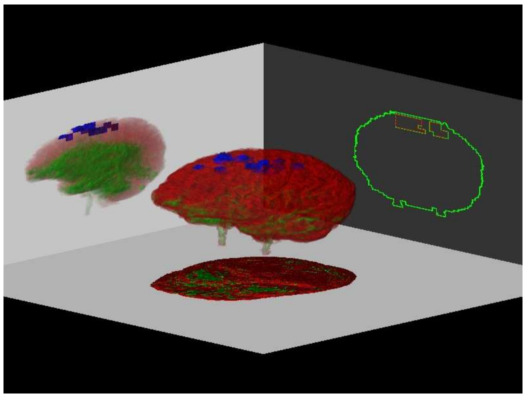


Figure 9.6: Magic mirrors with contour display

9.6 Results

The goal of the presented work was to develop a technique, which allows the identification of the location of regions with activation potential. In order to prove the usefulness of the presented Magic Mirrors approach a non-clinical test scenario was set up.

The data sets used for this work were gained from a 3-Tesla MRI scanner. The anatomical data sets were acquired using a pulse sequence called *MDEFT* (Modified Driven Equilibrium Fourier Transform). These images are T1-weighted. The functional data sets used in this work are $T2^*$-weighted, and have already been pre-processed, so that the regions of activations are emphasized in the volume, and all other regions are neglected.

The data sets featured the resolutions shown in Table 9.1. A small group of test persons was asked to interprete the given Magic Mirrors images and verbosly describe the location as well as the size of activated regions in relation to the brain anatomy. All evaluators reported, that it was an easy task to derive the location of activated regions from the multiple views and Magic Mirrors representations.

The volume of the activated regions was not estimated as easily. Although this information is implicitly know to the system from the properties of the functional data set, it might be desireable to get an instant impression of the size of the activated volume. We are going to deal with this problem in our future work (see Section 9.7).

Electronic versions of the presented images as well as animations can be found at our website: `http://www.cg.tuwien.ac.at/research/vis/vismed/MM/`.

9.7 Conclusions and future work

The problem of occlusion when utilizing direct volume rendering in the scope of multimodal volume visualization of functional/anatomical data of the human brain has been overcome.

Transfer-function volumes are used to realize highlighting and various volume sculpting tasks by assigning different transfer functions on a voxel by voxel basis. Multiple views of the visualized brain geometry in a single image provide additional information. Different representations of the multimodal scenario can be displayed by Magic Mirrors. Various combinations of data representations (utilizing different transfer functions and contour displays) provide the possibility to intuitivly derive information like location and size of activated regions from the visualization.

Current shortcomings are addressed in our future research work. A faster ray-casting system (shear warp based) is going to be utilized, which should

	anatomical data set	**functional data set**
resolution	256 x 256 x 64 slices	64 x 64 x 5 slices
voxel size	1mm x 1mm x 2mm	4mm x 4mm x 4mm

Table 9.1: Resolutions of the data sets used to get the results shown in this paper

allow the interactive investigation and manipulation of the visualization. Regions of activation are going to be treated as entities, allowing to visually encode the magnitude of volume size and other properties in our direct volume rendering approach. Furthermore functional data sources obtained from different modalities are going to be included, in order to compare the results.

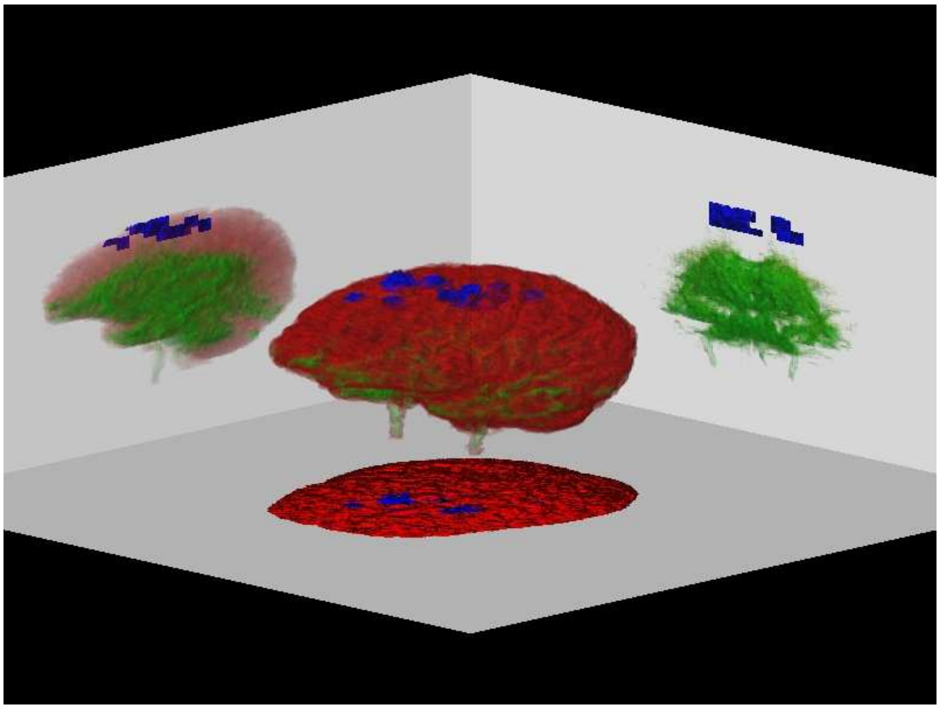


Figure 9.7: Magic Mirrors with different transfer functions (left and right mirrors) and projective view (bottom mirror)

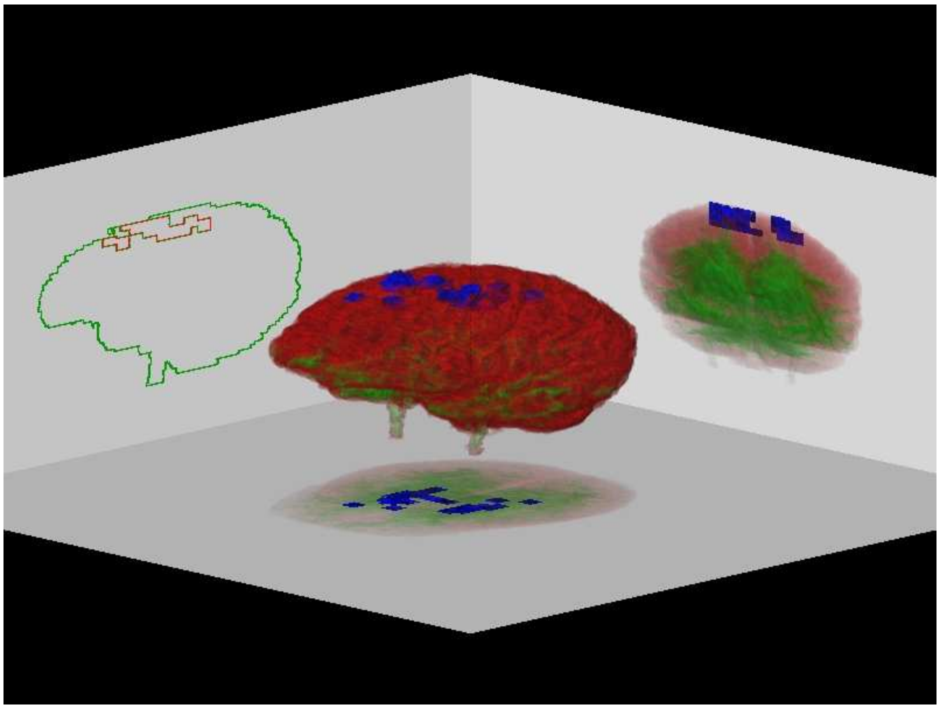


Figure 9.8: Magic Mirrors with contour display (left mirror)

Chapter 10

Acknowledgments

First of all, I want to thank our beloved Master Eduard Gröller, who always gave me hope, when something didn't work out the way, I wanted it to work out, and who gave me inspirations in this project.

Of course my thanks also go to Dipl.-Ing. Andreas König, who did most of the supervision of this diploma thesis and the underlying project. He was to me much more than a supervisor.

Than I have to thank Helwig, for even after I did both of my computer graphics projects under his supervision (and was not only successful), he still believed in me, and wanted me to do a diploma thesis under his supervision. Also my (little) knowledge of LaTeXwas quite often improved by some genious, helping remarks from him.

Now for the official part:

The work presented in this diploma thesis has been funded by the $\mathrm{V^{is}M^{ed}}$ project. $\mathrm{V^{is}M^{ed}}$ is supported by *Tiani Medgraph*, Vienna, `http://www.tiani.com`, and the *Forschungsförderungsfonds für die gewerbliche Wirtschaft*, Austria, `http://www.telekom.at/fff/`. Please refer to `http://www.vismed.at` for further information on this project.

This work has also also been funded by a *"Förderungsstipendium der Technisch-Naturwissenschaftlichen Fakultät der TU Wien"*.

This diploma thesis was written at the Institute of Computergraphics at the Vienna University of Technology. The project was carried out in very tight cooperation with the Institute of Medical Physics at the University of Vienna. The main cooperation partners there were Dipl.Ing. Dr. Markus Barth and a.o. Univ. Prof. Dr. Ewald MOSER. The cooperation included also the University Clinic of Neurology at the General Hospital of Vienna, the main contact person there was Ass.-Arzt Dr. Roland Beisteiner.

Dipl.-Ing. Dr. Fritz Tafill provided me with a lot of the background knowledge about MRI and related topics.

Then, of course I have to thank my Computer, `dance@cg.tuwien.ac.at`, for it did only crash approximately once a week, when it was the worst moment to crash. Additional remark: the installed OS is not the most stable available at the moment ;-).

I also want to thank my brother Herbert, who arranged a short publication of the work done in this thesis at the *Uni Managment Club*-Newspaper *Straight Forward*.

Last, but not least I want to thank my parents and Susi, for they were so patient, even when I worked for more than 70 hours a week, and supported me during my studies the best they could.

And now enough for the thanks,
I would say: Let's dance!

Bibliography

[1] Advanced Visual System Inc. *AVS Developers Guide, Release 4*, May 1992.

[2] Foster Findlay Associates. C_Images 3D homepage. See URL `http://www.demon.co.uk/ffaltd/3d.html`.

[3] R. Avila, T. He, L. Hong, A. Kaufman, H. Pfister, C. Silva, L. Sobierajski, and S. Wang. VolVis: A diversified volume visualization system. In R. Daniel Bergeron and Arie E. Kaufman, editors, *Proceedings of the Conference on Visualization*, pages 31–39, Los Alamitos, CA, USA, oct 1994. IEEE Computer Society Press.

[4] Silvia Castro, Andreas König, Helwig Löffelmann, and Eduard Gröller. Transfer function specification for the visualization of medical data. Technical Report TR-186-2-99-12, Institute of Computer Graphics, Vienna University of Technology, Austria, March 1998.

[5] International AVS Center. International AVS center module repository catalog. See URL `http://testavs.ncsc.org/ IAC/ catalog/ catalog.html`.

[6] D. W. Chakares and P. Schmalbrock. *Fundamentals of Magnetic Resonance Imaging*. Williams & Wilkins, 1992.

[7] L. Chen, GT. Herman, RA. Reynolds, and JK. Udupa. Surface shading in the cuberille environment. *IEEE Computer Graphics and Applications*, 5:33–43, 1985.

[8] Yoonsuck Choe and Risto Miikkulainen. Self-organization and segmentation with laterally connected maps of spiking neurons. In *Proceedings of WSOM'97, Workshop on Self-Organizing Maps, Espoo, Finland, June 4-6*, pages 26–31. Helsinki University of Technology, Neural Networks Research Centre, Espoo, Finland, 1997.

[9] Paolo Cignoni, Paola Marino, Claudio Montani, Enrico Puppo, and Roberto Scopigno. Speeding Up Isosurface Extraction Using Interval Trees. *IEEE Transactions on Visualization and Computer Graphics*, 3(2):158–170, April 1997.

[10] HE. Cline, WE. Lorensen, S. Ludke, CR. Crawford, and BC. Teeter. Two algorithms for the reconstruction of tomograms. *Med. Phys.*, 15:320–327, 1988.

[11] Chris Davies. Analyze homepage. See URL `http://www.cnsoft.com/Info/Analyze.html`, CNSoftware.

[12] K.E.W. Eberhardt, B.Tomandl, P. Hastreiter, T. Ertl, and R. Tröscher-Weber. Virtual endoscopic ct-angiography (vecta) in patients with intracranial aneurysms. In *1st Int. Conf. on Comp. Integr. Surg. (ISCAS), CIS '97, Linz, Austria.* Int. Soc. for Comp. Aid. Surg., Wiley and Sons Inc., 1997.

[13] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics: Principles and Practice.* Addison-Wesley Publishing Co., 2nd edition, 1990.

[14] P.T. Fox and M.E. Raichle. Stimulus rate determines regional brain blood flow in striate cortex. *Ann. Neurol.*, 17:303, 1985.

[15] K. J. Friston. Statistical Parametric Mapping. In *Functional Neuroimaging*, pages 79–93, Academic Press, San Diego, CA, 1994. R. W. Thatcher and M. Hallett and T. Zeffiro and E. R. John and M. Heurta.

[16] Hayden Image Processing Group. Alice homepage. See URL `http://www.perceptive.com`, Perceptive Systems, Inc.

[17] Hayden Image Processing Group. Cheshire homepage. See URL `http://www.perceptive.com/CHESHIRE.HTM`, Perceptive Systems, Inc.

[18] Medical Image Processing Group. 3DVIEWNIX homepage. See URL `http://www.mipg.upenn.edu/~Vnews/index.html`, Department of Radiology, University of Pennsylvania, Philadelphia.

[19] P. Hastreiter, H. K. Çakmak, and Th. Ertl. Intuitive and interactive manipulation of 3D data sets by integrating texture mapping based volume rendering into the openinventor class hierarchy. In Th. Lehman, I. Scholl, and K. Spitzer, editors, *Bildverarbeitung für die Medizin: Algorithmen, Systeme, Anwendungen*, pages 149–154. Inst. f. Med. Inf. u. Biom. d. RWTH, Aachen, Verl. d. Augustinus Buchhandlung, 1996.

[20] P. Hastreiter, T. Ertl, B. Tomandl, and K.E.W. Eberhardt. Vecta - virtual endoscopic ct-angiography. In W. Kalender, editor, *Proc. 4th International Workshop on Rapid Prototyping in Medicine and Computer-Assisted Surgery, Erlangen*, 1997. Abstract.

[21] GT. Hermann and HK. Liu. Three dimensional display of human organs from computed tomograms. *Computer Graphics Image Processing*, 9:1–21, 1979.

[22] K. H. Höhne and A. Pommert. *Volume Visualization. Brain Mapping: The Methods*, chapter 17. A. W. Toga and J. C. Mazziotta, Academic Press, San Diego, CA, 1996.

[23] X. Hu. Volumetric rendering of multimodality, multivariable medical imaging data. In *Computer Graphics (Chapel Hill Workshop on Volume Visualization)*, volume 23, pages 45–49, 1989.

[24] Advanced Visual System Inc. AVS homepage. See URL `http://www.avs.com`.

[25] Victoria Interrante, Henry Fuchs, and Stephen Pizer. Enhancing transparent skin surfaces with ridge and valley lines. In *IEEE Visualization '95*, pages 52–59, 1995.

[26] Victoria Interrante, Henry Fuchs, and Stephen Pizer. Illustrating transparent surfaces with curvature-directed strokes. In *IEEE Visualization '96*. IEEE, October 1996. ISBN 0-89791-864-9.

[27] Victoria L. Interrante. Illustrating surface shape in volume data via principal direction-driven 3D line integral convolution. In Turner Whitted, editor, *SIGGRAPH 97 Conference Proceedings*, Annual Conference Series, pages 109–116. ACM SIGGRAPH, Addison Wesley, August 1997. ISBN 0-89791-896-7.

[28] T. N. Jones and D. N. Metaxas. Automated 3D segmentation using deformable models and fuzzy affinity. *Lecture Notes in Computer Science*, 1230:113–??, 1997.

[29] Arie Kaufman. *Volume visualization.* IEEE Computer Society Press tutorial. IEEE Computer Society Press, 1109 Spring Street, Suite 300, Silver Spring, MD 20910, USA, 1991.

[30] Andreas König. Vismed - homepage. See URL `http://www.vismed.at/`, Institute of Computer Graphics, Vienna University of Technology, Austria.

[31] Philippe Lacroute and Marc Levoy. Fast volume rendering using a shear–warp factorization of the viewing transformation. In Andrew Glassner, editor, *Proceedings of SIGGRAPH '94 (Orlando, Florida, July 24–29, 1994)*, Computer Graphics Proceedings, Annual Conference Series, pages 451–458. ACM SIGGRAPH, ACM Press, July 1994. ISBN 0-89791-667-0.

[32] David Laur and Pat Hanrahan. Hierarchical splatting: A progressive refinement algorithm for volume rendering. In Thomas W. Sederberg,

editor, *Computer Graphics (SIGGRAPH '91 Proceedings)*, volume 25, pages 285–288, July 1991.

[33] Marc Levoy. Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, 8(3):29–37, May 1988.

[34] Marc Levoy. Efficient ray tracing of volume data. *ACM Transactions on Graphics*, 9(3):245–261, July 1990.

[35] WE. Lorensen and HE. Cline. Marching cubes: a high resolution 3D surface reconstruction algorithm. *Computer Graphics*, 21:163–169, 1987.

[36] J. B. Antoine Maintz and Max. A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):p.1–36, 1998.

[37] Alina N. Moga and Moncef Gabbouj. Parallel marker-based image segmentation with watershed transformation. *Journal of Parallel and Distributed Computing*, 51(1):27–45, May 1998.

[38] Torsten Möller, Raghu Machiraju, Klaus Mueller, and Roni Yagel. A comparison of normal estimation schemes. In Roni Yagel and Hans Hagen, editors, *IEEE Visualization 97*, pages 19–26. IEEE, November 1997.

[39] O. Monga. An optimal region growing algorithm for image segmentation. *International Journal of Pattern Recognition and Artificial Intelligence*, 1:351–375, 1987.

[40] C. Montani, R. Scateni, and R. Scopigno. Discretized marching cubes. In R. Daniel Bergeron and Arie E. Kaufman, editors, *Proceedings of the Conference on Visualization*, pages 281–287, Los Alamitos, CA, USA, October 1994. IEEE Computer Society Press.

[41] Heinz Morneburg. *Bildgebende Systeme für die medizinische Diagnostik*. Publicis MCD Verlag, 1995.

[42] Eric N. Mortensen and William A. Barrett. Intelligent scissors for image composition. In Robert Cook, editor, *SIGGRAPH 95 Conference Proceedings*, Annual Conference Series, pages 191–198. ACM SIGGRAPH, Addison Wesley, August 1995. held in Los Angeles, California, 06-11 August 1995.

[43] Department of Computer Science. VolVis homepage. See URL `http://www.cs.sunysb.edu/~vislab/volvis_home.html`, State University of New York at Stony Brook.

[44] Institut of Mathematics and Computer Science in Medicine. Voxel-Man homepage. See URL `http://www.springer-ny.com/medicine/voxelman/index.htm`, University of Hamburg.

[45] Kwang-Man Oh and Kyu Ho Park. A type-merging algorithm for extracting an isosurface from volumetric data. *The Visual Computer*, 12(8):406–419, 1996. ISSN 0178-2789.

[46] Philips Medical Systems, The Nederlands. *Principles of MR Imaging*, 1984.

[47] Riccardo Poli and Guido Valli. Optimum segmentation of medical images with hopfield neural networks. Technical Report CSRP-95-12, University of Birmingham, School of Computer Science, October 1995.

[48] P. Radeva, A. A. Amini, and J. Huang. Deformable B-solids and implicit snakes for 3D localization and tracking of SPAMM MRI data. *Computer Vision and Image Understanding: CVIU*, 66(2):163–??, ???? 1997.

[49] K. Rehm, K. Lakshminaryan, S. Frutiger, K. A. Schaper, D. W. Sumners, S. C. Strother, J. R. Anderson, and D. A. Rottenberg. A symbolic environment for visualizing activated foci in functional neuroimaging datasets. *Medical Image Analysis*, 2(3):215–226, 1998.

[50] K. Rehm, S. C. Strother, J. R. Anderson, K. A. Schaper, and D. A. Rottenberg. Display of merged multimodality brain images using interleaved pixels with independent color scales. *Journal of Nuclear Medicine*, 35:1815–1821, 1994.

[51] Georgios Sakas, Marcus Grimm, and Alexandros Savopoulos. Optimized maximum intensity projection (MIP). In *Eurographics Rendering Workshop 1995*. Eurographics, June 1995.

[52] R. Sammouda, N. Niki, and H. Nishitani. Neural network based segmentation of magnetic resonance images. *IEEE Nuclear Science Symposium*, 4:1827–1831, 1995.

[53] R. Shu, C. Zhou, and M. S. Kankanhalli. Adaptive marching cubes. *The Visual Computer*, 11(4):202–217, 1995. ISSN 0178-2789.

[54] J. Sijbers, P. Scheunders, M. Verhoye, A. Van der Linden andD. Van Dyck, and E. Raman. Watershed-based segmentation of 3D MR data for volume quantization. *Magnetic Resonance Imaging*, 15(4), 1997.

[55] Friedrich Tafill. *Automatic Classification of Non-Pathological Double-Echo Magnetic Resonance Images by Neural Networks*. PhD thesis, Vienna University of Technology, 1998.

[56] J. Talairach and P. Tournoux. *Co-planar Stereotaxic Atlas of the Human Brain*. Thieme Medical Publishers, Inc., New York, 1988.

[57] A. Ultsch. Self-organizing neural networks for visualization and classification. In O. Opitz, B. Lausen, and R. Klar, editors, *Information and Classification*, pages 307–313, Berlin, 1993. Springer.

[58] Lee Westover. Footprint evaluation for volume rendering. In Forest Baskett, editor, *Computer Graphics (SIGGRAPH '90 Proceedings)*, volume 24, pages 367–376, August 1990.

[59] Karel J. Zuiderveld, Anton H. J. Koning, Rik Stokking, J. B. Antoine Maintz, Fred J. R. Appelman, and Max A. Viergever. Multimodality visualization of medical volume data. *Computers and Graphics*, 20(6):775–??, November–December 1996.

[60] Karel J. Zuiderveld, Peter M. A. van Ooijen, John W. C. Chin-A-Woeng, Pieter C. Buijs, Marco Olree, and Frits H. Post. Clinical evaluation of interactive volume visualization. In *IEEE Visualization '96*. IEEE, October 1996. ISBN 0-89791-864-9.