

## Interaktive medizinische Volumenvisualisierung – ein Überblick

Bernhard Preim, Wolf Spindler, Heinz-Otto Peitgen

MeVis – Centrum für Medizinische Diagnosesysteme und Visualisierung,  
Universitätsallee 29, 28359 Bremen, Email: {preim, spindler, peitgen}@mevis.de

**Zusammenfassung.** Die Techniken zur Volumenvisualisierung sind in den letzten Jahren, vor allem durch dedizierte Hardwareunterstützung, so verbessert worden, dass sie in der klinischen Routine genutzt werden. Dieser Beitrag bietet einen Überblick über die Verfahren zur Visualisierung regelmäßiger Volumendaten, die in medizinischen Anwendungen vorherrschen. Schwerpunkte sind die direkte Visualisierung von Volumendaten durch bildbasierte, objektbasierte und texturbasierte Verfahren sowie Interaktionstechniken zum Umgang mit den Volumendaten.

Neue Anwendungen ergeben sich durch die Überlagerung mehrerer Volumendatensätze, die durch eine Registrierung aufeinander abgebildet werden. Die Exploration derart fusionierter Volumen wird ebenfalls behandelt. Der Überblick wird durch eine Übersicht über die Produkte zur Volumenvisualisierung abgerundet.

**Schlüsselwörter:** Volumenvisualisierung, Interaktionstechniken, Transferfunktionen

### 1 Einleitung<sup>1</sup>

Die Visualisierung von Volumendaten ist von hohem Interesse in der Forschung und in der praktischen Anwendung. Das seit 1992 jährlich stattfindende ACM Symposium „Volume Visualization“ sowie die hohe Zahl an Beiträgen auf Computergraphik- und Visualisierungskonferenzen sowie die wachsende Verbreitung kommerzieller Werkzeuge belegen dies. Die am weitesten verbreiteten Anwendungen für die Volumenvisualisierung sind medizinischer Natur.

Medizinische Volumendaten resultieren aus der Anwendung bildgebender Verfahren – vor allem Computertomographie (CT), Magnetresonanztomographie (MRT) und 3D-Ultraschall. Diese Verfahren erzeugen (eindimensionale) Intensitätswerte. Die in einzelnen Schichtbildern enthaltenen aber räumlich schwer vorstellbaren Strukturen (z.B. ineinander greifende Gefäßsysteme, krankhafte Veränderungen) werden in ihrer Ausdehnung und Lage besser beurteilbar durch eine interaktiv handhabbare 3D-Darstellung. 3D-Darstellungen patientenindividueller Daten unterstützen die Diagnosestellung sowie Therapieentscheidungen, wie die Bestrahlungsplanung und die Planung chirurgischer Eingriffe. In der medizinischen Ausbildung, z.B. in der Anatomie, werden idealisierte Volumenmodelle wie die Visible Human-Datensätze zur Exploration räumlicher Verhältnisse genutzt.

Bei MEVIS wird seit 1993 ein Volumenrenderer entwickelt, der eine Vielzahl von Visualisierungsverfahren integriert und vielfältig eingesetzt wird. Anhand dieses Volumenrenderers und anderer Arbeiten wird ein Überblick über das Gebiet gegeben. Dieser Überblick unterscheidet sich von anderen Übersichtsartikeln [E1v92], [YLT96] durch zwei Aspekte: die Integration von Interaktionstechniken und die Berücksichtigung kommerzieller Werkzeuge.

---

<sup>1</sup> Proc. of Simulation und Visualisierung (Magdeburg, März 2000), SCS-Verlag Ghent

## 2 Visualisierung regelmäßiger Volumendaten

In der medizinischen Visualisierung kommen Schichtdaten zum Einsatz, die z.B. den Bauchbereich, den Kopf oder die Extremitäten des Menschen darstellen. Die Auflösung der Datensätze beträgt häufig  $512 \times 512$  pro Schicht; die Anzahl der Schichten reicht von etwa 40 bis etwa 200, z.B. bei CT-Aufnahmen der Lunge. Daraus resultieren Datensätze mit einer Größe von 10–50 MegaVoxel (MVoxel). Durch die Weiterentwicklung der bildgebenden Verfahren (höhere Auflösung) steigen die Anforderungen an eine effiziente Visualisierung. Die Grauwertauflösung ist oft sehr hoch: bei CT-Bildern in der Regel 12 Bit, wobei diese für die Visualisierung aus Effizienzgründen oft auf 8 Bit-Daten transformiert werden. Medizinische Daten sind regelmäßig – das heißt, die Voxel sind in einem orthogonalen Gitter angeordnet, wobei jede Schicht die gleiche Anzahl an Voxeln hat und jedes Voxel das gleiche Volumen repräsentiert. Unregelmäßige Volumendaten, wie sie z.B. bei geographischen Volumendaten vorkommen, werden hier nicht betrachtet.

Die wesentlichen Kriterien für die Güte der Volumenvisualisierung sind die Qualität (detailgetreue Abbildung, räumlicher Eindruck), die Laufzeit, die Speichereffizienz und die Vielfalt der Datenrepräsentation (z.B. die Möglichkeit, mehrdimensionale Voxel darzustellen). Welches Verfahren besonders günstig ist, hängt von den Eigenschaften der Daten, dem Ziel der Visualisierung und von der verwendeten Hardware ab. Eine weitere Anforderung besteht darin, dass Volumendaten und Oberflächendaten in einer Visualisierung gemischt werden können. Auf diese Weise können z.B. modellierte Instrumente und medizinische Daten in einer chirurgischen Anwendung kombiniert werden.

Es gibt zwei große Klassen von Visualisierungsverfahren: direkte und die indirekte Volumenvisualisierung. Bei der direkten Volumenvisualisierung (DVR) tragen alle Voxel zu einem Bild bei, während bei der indirekten Visualisierung Teile des Volumens in eine Oberflächenrepräsentation überführt und als Oberfläche dargestellt werden.

## 3 Direkte Volumenvisualisierung

Bei DVR-Verfahren werden die Daten projiziert und unter Berücksichtigung ihrer Entfernung und der Transparenz überlagert. Es gibt drei Klassen von DVR-Algorithmen:

- bildbasierte Verfahren, die für jedes Pixel einen Strahl in die Szene (zurück) verfolgen und die Farbe des Pixels aus den getroffenen Voxeln – gewichtet mit der Transparenz – zusammensetzen (*back to front*),
- objektbasierte Verfahren, die den Objektraum (die Voxel) abtasten und ermitteln, welchen Beitrag die Voxel zu dem Bild (*front to back*) leisten, und
- texturbasierte Verfahren, die einen 3D-Texturspeicher und Hardwareunterstützung für das Texture Mapping nutzen.

Die Qualität der Bilder und die Darstellungsgeschwindigkeit hängen davon ab, mit welcher Frequenz das Volumen abgetastet wird und wie die Abtastpunkte berücksichtigt werden. Eine relativ niedrige Qualität liefert das Nearest-Neighbour-Verfahren, bei dem für jeden Abtastpunkt nur der nächstliegende Voxel berücksichtigt wird. Deutlich bessere Ergebnisse können durch eine Interpolation der benachbarten Voxel erreicht werden (z.B. trilineare Interpolation aus den 8 umgebenden Voxeln, d.h. es wird in  $x$ -,  $y$ - und  $z$ -Rich-

tung linear interpoliert). Bei der Erkundung von Volumendaten wird ausgenutzt, dass während der Interaktion (z.B. bei einer Rotation) nicht die volle Auflösung benötigt wird.

### 3.1 **Bildbasierte DVR-Verfahren**

Bildbasierte DVR-Verfahren basieren darauf, dass Strahlen durch die Bildebene in die Szene verfolgt werden. In Analogie zu den Verfahren zur Visualisierung von Oberflächenmodellen werden diese (diskretes) Raycasting oder – bei Verfolgung des Lichtstrahls über den ersten Schnittpunkt hinaus – (diskretes) Raytracing genannt. Das Attribut diskret verdeutlicht den Unterschied zur Visualisierung von Oberflächenmodellen: die diskrete Beschreibung der Modelle. Im diskreten wird die Schnittpunktberechnung erleichtert, da es nur eine Art von Objekt gibt – das Voxel. Diskretes Raytracing führt dazu, dass Beleuchtungseffekte, wie Lichtbrechung, Schatten und Mehrfachreflexion berücksichtigt werden (Abb. 2). Diskretes Raytracing ist z.B. im RASTERRAYTRACER implementiert, wobei polygonale Daten und Volumendaten gemischt werden [YCK92]. Dazu werden die kontinuierlichen Oberflächendaten mittels Scankonvertierung diskretisiert.

Bei bildbasierten Verfahren wird das Volumen nicht in der Reihenfolge traversiert, in der es im Speicher vorliegt, da die Sichtstrahlen das Volumen in beliebiger Richtung schneiden. Daher werden häufig Daten benötigt, die sich nicht im Cache befinden, so dass ein hoher Aufwand entsteht. Bildbasierte Verfahren lassen sich beschleunigen, indem die Strahlverfolgung abgebrochen wird, wenn die Summe der entlang des Strahls gesammelten Opazitäten oberhalb eines Schwellwertes nahe 1 ist (*early ray termination*) bzw. dadurch, dass die Auflösung in z-Richtung bei größerer Entfernung verringert wird [Lev90]. Bei einem Schwellwert von 0,95 ist dadurch bei typischen medizinischen Datensätzen eine Geschwindigkeitssteigerung um einen Faktor zwischen 1,6 und 2,2 möglich.

Zur Beschleunigung bildbasierter Verfahren kann die Kohärenz von Pixeln im Bild genutzt werden. Das Bild wird zunächst in einem groben Raster berechnet und zwischen Pixeln mit ähnlichem Grauwert wird interpoliert. Nur bei großen Grauwertunterschieden wird adaptiv verfeinert. Diese für das klassische Raytracing entwickelte Strategie wurde von LEVOY auf die Volumenvisualisierung angewendet [Lev90b].

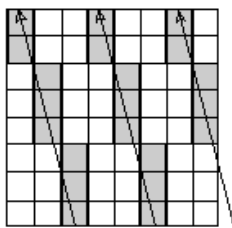


Abbildung 1: Strahlentemplates beim parallelen Raycasting, nach [YK92]



Abbildung 2: Realistische Beleuchtungswiedergabe durch volumetrisches Raytracing, Screenshot vom VOLVIS-System aus [Avi92]

Bei bildbasierten Verfahren ist die Parallelprojektion besonders schnell. Dabei werden einmalig die Voxeln berechnet, die bei der Verfolgung eines Strahls entstehen (3D-Bresenham-Algorithmus). Diese Voxel werden in einem *ray template* gespeichert, das für die weitere Berechnung über das Bild geschoben wird [YK92] (Abb. 1). Da die Volumen-

visualisierung aufwändig ist, wird die Idee des Progressive Refinement – die schrittweise Verfeinerung eines zunächst sehr grob – gerenderten Bildes verfolgt [LH91]. Dadurch kann das entstehende Bild frühzeitig beurteilt und das Rendern ggf. abgebrochen werden.

### **3.2 Objektbasierte DVR-Verfahren**

Bei objektbasierten Verfahren wird über die Voxel des Datensatzes iteriert und errechnet, welchen Einfluss sie auf das Bild haben. Dazu wird ein Filter benutzt, der den Anteil eines Voxels entsprechend der Filtergewichte auf die Pixel des Bildes verteilt: Die Voxel hinterlassen einen „Fußabdruck“ auf dem Bild, d.h. sie beeinflussen einen Bereich des Bildes. Der Anteil eines Voxels ist dabei für ein zentrales Pixel sehr hoch und für die Pixel in der Nachbarschaft entsprechend geringer. In [Wes90] wird das Splatting-Verfahren (to splat – etwas gegen etwas klatschen) vorgestellt, das durch die Nutzung hierarchischer Datenstrukturen verbessert wurde (Hierarchical Splatting, [LH91]).

Objektbasierte Verfahren können davon profitieren, dass die verarbeiteten Voxel in einem Cache gehalten werden. Hierarchische Datenstrukturen wie Pyramiden, Octrees (Zerlegung in quaderförmige Teilvolumen), Zerlegungen in Tetraeder oder Wavelets werden genutzt, um evtl. große Bereiche, in denen das Volumen nicht besetzt oder die Opazität (nahe) 0 ist, schnell zu überspringen. Der Anteil der Voxel, die auf diese Weise übersprungen werden kann, liegt meist zwischen 70 und 90 Prozent [LL94]. Eine derartige Ausnutzung der Kohärenz erfordert allerdings eine Vorverarbeitung, die sich erst bei mehrfachem Rendern rentiert und die bei jeder Veränderung der Transferfunktion (und damit der Opazitätszuordnung) wiederholt werden muss.

Bildbasierte Verfahren können nur geringfügig durch hierarchische Datenstrukturen beschleunigt werden, da der Strahl immer wieder unterschiedliche Ebenen der Hierarchie traversiert. Die Shear-Warp-Transformation [LL94] verbindet die Möglichkeiten zur Effizienzsteigerung von bild- und objektbasierten Verfahren durch eine mehrstufige Transformation. Dabei wird das Volumen im ersten Schritt so geschert, dass die Schichten des Volumens genau senkrecht zur Sichtrichtung stehen, so dass sie sehr effizient auf die Bildebene abgebildet werden können. Unter den drei möglichen Richtungen das Volumen in Schichten zu zerlegen, wird diejenige gewählt, bei der der Winkel zwischen der Sichtrichtung und den Schichten 90 Grad am nächsten ist. Bei einer perspektivischen Transformation kommt zu der Scherung eine Skalierung, die die Verkleinerung der entfernten Schichten berücksichtigt. Die Schichten des Volumens werden im zweiten Schritt von vorn nach hinten semitransparent überlagert, so dass ein Zwischenbild entsteht. Dieses Zwischenbild wird durch eine Verzerrung (warp) in das gerenderte Bild überführt, wobei die ursprüngliche Scherung ausgeglichen wird. Auf dieser Grundlage kann die Kohärenz im Volumen und im Bild bestmöglich ausgenutzt werden. Dieses Verfahren ist das schnellste in Software realisierte Visualisierungsverfahren.

### **3.3 Texturbasiertes Volumenrendering**

Bei dieser neueren Klasse von Volumenvisualisierungsverfahren wird das Volumen in einen 3D-Texturspeicher geladen, der in hochwertigen Graphikworkstations zur Verfügung steht. Ursprünglich wurden diese Speicher für das Rendering texturierter Oberflächenmodelle entwickelt; sie können aber auch zur Volumenvisualisierung genutzt werden. Dazu werden parallele, äquidistante Schnittebenen durch das Volumen gelegt und von hinten nach vorn – entsprechend der Opazitätswerte – überblendet. Für eine hohe Qualität der Visualisierung kommt eine trilineare Interpolation zum Einsatz. Sowohl das

Überblenden als auch die trilineare Interpolation sind in Hardware realisiert und dementsprechend schnell. Diese Technik wurde in [CCF94] vorgestellt und ist in der Bibliothek VOLUMIZER von Silicon Graphics realisiert. Die dadurch mögliche Geschwindigkeitssteigerung hängt entscheidend von der Größe des Datensatzes und der des Texturspeichers (typisch sind 4 MByte) ab. Wenn das Volumen nicht komplett in den Texturspeicher passt oder die Größe des Datensatzes nicht einer Zweierpotenz entspricht, wird es in quaderförmige Blöcke (sogenannte *Bricks*) eingeteilt, die beim Rendern nach ihrer Entfernung zum Betrachter sortiert werden. Dabei werden die texturierten Polygone eines Bricks von hinten nach vorn gezeichnet, ehe der nächste, näher am Betrachter liegende Brick abgearbeitet wird. Bricks überlappen an den Enden etwas (oft um ein Voxel), damit eine korrekte trilineare Interpolation möglich ist. Das Laden des Texturspeichers ist zeitaufwändig, so dass die Geschwindigkeit stark abnimmt, wenn viele Blöcke erzeugt werden müssen. Mit texturbasierten Verfahren sind gegenwärtig 100 MVoxel/s möglich, so dass bei typischen Datensätzen 2-10 Frames/Sekunde erreicht werden. In Abb. 3 wird das Vorgehen bei der bildbasierten und texturbasierten Visualisierung gegenübergestellt.

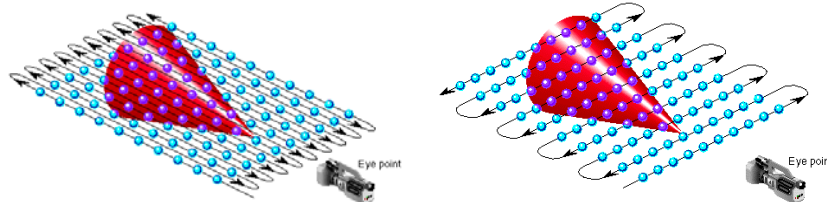


Abbildung 3: Vergleich der Vorgehensweisen bei der bildbasierten Volumenvisualisierung (ray-casting, links) und bei der Überlagerung von Schichten bei texturbasierten Verfahren (rechts), Screenshot von Silicon Graphics®

### 3.4 Projektionsmethoden in der direkten Volumenvisualisierung

Die zuvor beschriebenen DVR-Verfahren sind hinsichtlich der erzeugten Bilder äquivalent, was durch Transformation der zugrunde liegenden Gleichungen gezeigt werden kann. Im Unterschied zu diesen Verfahren dienen Projektionsmethoden dazu, bestimmte Strukturen hervorzuheben. Die bekanntesten Vertreter sind die Maximum- und Minimum-Intensity-Projektion (MIP und mip). Projektionsmethoden können durch jede der oben beschriebenen Volumenvisualisierungsverfahren realisiert werden. Bei bildbasierten Projektionsverfahren wird für jeden Bildpunkt ein Strahl in die Szene verfolgt und jeweils der Voxel mit dem minimalen bzw. maximalen Intensitätswert dargestellt. Auf diese Weise werden häufig Blutgefäße in kontrastierten Bildern visualisiert (siehe Abb. 4).

MIP-Darstellungen heben bestimmte Strukturen hervor; geben aber die Tiefenverhältnisse nicht korrekt wieder. So werden große Gefäße, die bei einer Kontrastierung sehr hell erscheinen, dargestellt, obwohl vor ihnen kleinere (dunklere) Gefäße liegen. Eine Modifikation von MIP-Darstellungen ist die Closest Vessel Projection (CVP) [ZKV94] (Abb. 4, rechts). Dabei wird ein Schwellwert ermittelt, der unterhalb der Intensität aller Gefäße liegt. Für jeden Strahl wird der erste Voxel dargestellt, der oberhalb des Schwellwertes liegt – die Tiefenverhältnisse der Gefäße werden korrekt wiedergegeben und das Verfahren ist schneller, da die Strahlfollowung oft abgebrochen werden kann, ehe das Volumen traversiert wurde. Allerdings sind CVP-Projektionen stark von der Wahl eines geeigneten Schwellwertes abhängig, damit keine Gefäße unterdrückt, aber auch keine

irrelevanten Strukturen eingeblendet werden. In Lehrsystemen wird häufig die average-Projektion eingesetzt, die eine Röntgenprojektion simuliert. Dabei wird für jedes Pixel die Intensität der Voxel entlang des entsprechenden Strahls arithmetisch gemittelt.

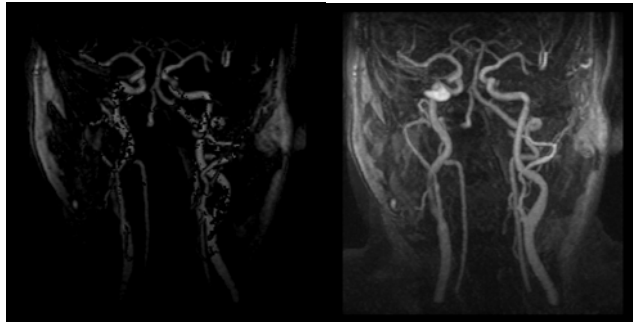


Abbildung 4: Vergleich einer MIP-Darstellung (links) und einer CVP-Darstellung (rechts), die Blutgefäße im Kopfbereich darstellen, mit freundl. Genehmigung von SVEN KOHLE, MEVIS

### 3.5 Shading bei der direkten Volumenvisualisierung

Bei DVR-Verfahren hängt die Farbe eines Pixels nur von den Intensitäten der beteiligten Voxel (und nicht von der Orientierung relativ zu einer Lichtquelle) ab. Dadurch geben die entstehenden Bilder nur wenig Tiefenhinweise und feine Strukturen sind kaum erkennbar. Eine perspektivische Darstellung gibt räumliche Verhältnisse besser wieder als eine Parallelprojektion. Für eine gute Erkennung der räumlichen Verhältnisse (Tiefe, Abstände der Objekte voneinander und vom Betrachter) ist es wichtig, die Orientierung von Oberflächen zu einer Lichtquelle zu berücksichtigen. Diese Orientierung ergibt sich bei der Oberflächendarstellung aus den Oberflächennormalen. In einem Volumendatensatz kann eine Normale geschätzt werden.

Die erste Variante dieser Schätzung, das Depth-Shading, beruht auf den Tiefenwerten: aus den Tiefenwerten der sichtbaren Voxel wird ein Gradient geschätzt [G85]. Aufwändiger, aber qualitativ besser ist das Graylevel Gradient-Shading. Dabei wird der Gradient ausgehend von den Intensitätswerten geschätzt [HB86]. Dazu werden die Intensitätswerte in einer  $3 \times 3 \times 3$ -Nachbarschaft genutzt. Der Gradient für das zentrale Voxel  $v_{i,j,k}$  ergibt sich durch Differenz der Intensitäten der benachbarten Voxel ( $v_{i+1}-v_{i-1}$ ,  $v_{j+1}-v_{j-1}$ ,  $v_{k+1}-v_{k-1}$ ). Abb. 5 zeigt das Prinzip und Abb. 6 eine Darstellung, die mit diesem Shading-Verfahren erzeugt wurde. Problematisch daran ist, dass die Berechnung die Interpolation von Voxeln nicht berücksichtigt, ob die Voxel zu einem Objekt gehören. Als Verbesserung wurde in [YKZ91] das kontextsensitive Shading eingeführt. Dabei wird in einem ersten Schritt ermittelt, welche Voxel zu einem Objekt gehören. Diese bilden einen Kontext. Bei der Schätzung der Normalen eines Voxels werden nur solche Nachbarn benutzt, die den gleichen Kontext haben. Die Gradienten in einem Vorverarbeitungsschritt berechnet und im Volumen gespeichert. Dabei wird der errechnete Normalenvektor so gerundet, dass er in etwa 6-8 Bit gespeichert werden kann. So können z.B. in einem 16-Bit-Datenvolumen 9 Bit für die Intensität (512 Werte) und 7 Bit für den Gradienten verwendet werden. Für eine schnelle Visualisierung wird in einer 16-Bit-Lookup-Tabelle für jede Kombination aus Intensität und Gradient berechnet, welcher Wert für ein so beleuchtetes Voxel entsteht. Diese Berechnung muss bei jeder Transformation des

Volumens oder der Lichtquelle gestartet werden. Da die Lookup-Tabelle in diesem Beispiel nur 64 K Einträge hat, spart man damit viel Zeit gegenüber der Berechnung für die 10-50 MVoxel eines typischen Datensatzes.

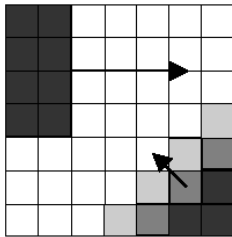


Abbildung 5: Approximation der Oberflächennormalen in einem Volumenmodell, nach [YKT96]



Abbildung 6: Beleuchtete Darstellung eines MRT-Datensatzes des Kopfes. Die Details an den Ohren, an den Augen und an der Nase werden erst durch die Berücksichtigung der Gradienten erkennbar. (Screenshot vom MEVIS VOLUMERENDERER)

## 4 Indirekte Volumenvisualisierung

Beim indirekten Volume Rendering wird anhand eines Schwellwertes eine Oberfläche gebildet, die sich aus den Grenzflächen von Voxeln zusammensetzt, die diesen Wert über- oder unterschreiten. Dazu werden vor allem der Marching Cubes-Algorithmus [LC87] oder der Dividing Cubes-Algorithmus [CLL88] angewendet. Beide Verfahren überführen die (Volumen)-Daten in maximaler Genauigkeit in eine Oberflächenrepräsentation. Dazu werden jeweils Würfel, die aus  $2 \times 2 \times 2$  Voxeln bestehen (4 Voxel einer Ebene  $k$  und die entsprechenden 4 Voxel der Ebene  $k+1$ ), daraufhin analysiert, welche der acht Voxel Werte oberhalb des Schwellwertes aufweisen. Dabei gibt es prinzipiell  $2^8$  Möglichkeiten, von denen die meisten durch Transformation auseinander hervorgehen. Daher reicht es die 15 in Abb. 7 dargestellten Fälle zu unterscheiden. Davon ausgehend wird beim Marching Cubes-Algorithmus ein Dreiecksnetz konstruiert, das die Mittelpunkte der Würfelseiten verbindet. Die Qualität der Approximation kann verbessert werden, wenn statt der Mittelpunkte zwischen zwei Voxeln, zwischen denen die Oberfläche verläuft, linear interpoliert wird. Wenn also Voxel  $v_1$  eine Intensität  $i_1$  aufweist und Voxel  $v_2$  eine Intensität  $i_2$  aufweist, wobei  $i_1$  oberhalb des Schwellwertes  $i$  und  $i_2$  unterhalb dieses Schwellwertes liegt, wird die Strecke zwischen  $v_1$  und  $v_2$  im Verhältnis  $(i-i_2)/(i_1-i_2)$  geteilt.

Der Dividing Cubes-Algorithmus unterteilt die Würfel, die aus  $2 \times 2 \times 2$  Voxeln bestehen, solange, bis Würfel entstehen, die vollständig innerhalb oder außerhalb eines Objektes liegen. Im Ergebnis werden 3D-Punkte und Normalen erzeugt. Der Dividing Cubes-Algorithmus ist für medizinische Anwendungen schneller als der Marching Cubes-Algorithmus. Abb. 8 zeigt eine indirekte Volumenvisualisierung, wobei der Schwellwert so gewählt wurde, dass die Knochen gut dargestellt werden.

Da die resultierende Menge an Polygonen für die interaktive Handhabung bei typischen medizinischen Datensätzen zu groß ist (Größenordnung  $10^6$ ) werden die Polygonnetze anschließend reduziert. Dabei besteht das Ziel darin, die Zahl der Polygone stark zu dezimieren und den Fehler (Abweichung von der originalen Geometrie) klein zu halten. Etablierte Verfahren dazu sind *Mesh Decimation* [SZL92] und *Progressive Me-*

shes [Hop96], mit denen eine Reduktion um eine Größenordnung möglich ist. Die resultierende Oberfläche kann auf Graphik-Workstations effizient gerendert werden, weil das Rendern (Transformieren, Projizieren, Beleuchten) von Dreiecksnetzen hardwaremäßig unterstützt wird.

Der große Nachteil indirekter Visualisierungen besteht darin, dass für jedes Voxel eine binäre Entscheidung getroffen werden muss, ob es zu einer Oberfläche gehört oder nicht und die allermeisten Voxel dabei verworfen werden. Die Berechnung muss erneut gestartet werden, wenn ein anderer Schwellwert eingestellt wird. Dadurch ist die indirekte Volumenvisualisierung ungeeignet, um sich mit den Volumendaten vertraut zu machen, sondern eher für das Erzeugen fertiger Präsentationen nützlich. Wenn keine abgrenzbaren Oberflächenkonturen existieren, ist die indirekte Volumenvisualisierung ungeeignet.

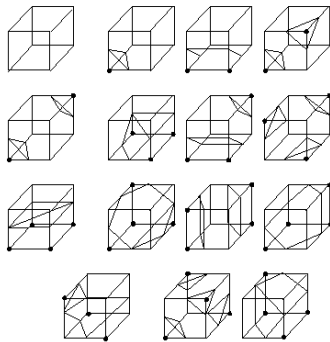


Abbildung 7: Prinzip der Isooberflächen-Generierung nach dem Marching Cubes-Algorithmus



Abbildung 8: Isooberflächen-Darstellung der Brustwirbelsäule (Ansicht von vorn), (Screenshot vom MEVIS VOLUMERENDERER)

## 5 Interaktionsaufgaben und -techniken zur Exploration von Volumendaten

Zur Erkundung von Daten ist eine interaktive Handhabung von Volumendaten nötig, so dass graduelle Übergänge zwischen verschiedenen Sichten möglich sind. Neben den grundlegenden geometrischen Transformationen (Rotation, Translation, Skalierung) sind die folgenden Interaktionsaufgaben wesentlich:

- Selektion in Volumendaten,
- Manipulation von Transferfunktionen,
- Multiplanare Reformatierung (MPR),
- Einfügen von Schnittebenen,
- Deformation von Volumenmodellen,
- Exploration von Ausschnitten eines Volumens und
- Distanz-, Abstands- und Volumenmessungen

Die Interaktionstechniken, mit denen diese Interaktionsaufgaben realisiert werden können, werden im folgenden beschrieben.

### 5.1 Selektion in Volumendaten

Die Selektion von Teilen eines Volumens (Regionen oder einzelnen Voxeln) ist Voraussetzung für viele weitere Interaktionen (z.B. um Startpunkte für eine Segmentierung von



Bildbereichen zu setzen). Präzise, aber wenig intuitiv ist eine numerische Angabe der Voxelwerte. Eine derartige Spezifikation kann durch Einfügen eines geeignet skalierten und beschrifteten Koordinatensystems erleichtert werden. Wesentlich natürlicher ist die Selektion in der Visualisierung. Dazu werden häufig mehrere Ansichten dargestellt: in einer Ansicht können einzelne Schichten und darin (in 2D) bestimmte Bereiche gewählt werden; in einer zweiten Sicht wird die entsprechende Stelle in der 3D-Ansicht hervorgehoben. Diese direkt-manipulative Selektion (Picking) kann erleichtert werden, indem das Picking auf bestimmte Intensitätsbereiche eingeschränkt wird. Dabei wird an der vom Benutzer selektierten Stelle im Viewport ein Strahl in die Szene verfolgt und der erste Voxel zurückgegeben, der in diesem Intensitätsbereich liegt.

## 5.2 Manipulation von Transferfunktionen

Für die Exploration unsegmentierter Volumendaten ist eine intuitive und zugleich präzise Einstellung der Transferfunktion die wichtigste Interaktion. Die Transferfunktion ist eine Abbildung der Intensitätswerte auf Transparenz-, Grau- und/oder Farbwerte. Einfache Transferfunktionen definieren eine Fensterung, bei der die Darstellung auf Voxel aus einem bestimmten Intensitätsbereich eingeschränkt wird (*windowing*). Die Breite des eingestellten Intervalls bestimmt dabei den Kontrast der Darstellung und der Mittelpunkt des Intervalls bestimmt die Helligkeit. Ausgefeiltere Transferfunktionen sind lineare Funktionen oder Polynome höherer Ordnung (oft kubische Funktionen, siehe Abb. 9). Bei DVR-Verfahren kann durch eine Veränderung der Transferfunktion die Sichtbarkeit von Strukturen graduell verändert werden (Abb. 10). Bei segmentierten Daten kann diese Einstellung separat auf einzelne anatomische Strukturen angewendet werden. Sehr hilfreich ist es, wenn bei der Einstellung einer Transferfunktion das Histogramm des Datensatzes hinterlegt wird, so dass erkennbar ist, wie groß der Anteil der Daten ist, der von der aktuellen Einstellung betroffen ist.

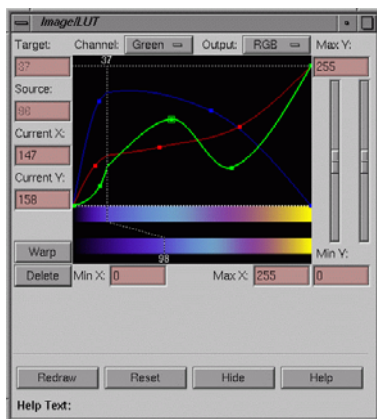


Abbildung 9: Manipulation von Transferfunktionen für die RGB-Ausgabe. Eine Transferfunktion (Rot, Grün, Blau über der Intensität ist jeweils aktiv und kann verändert werden. Der Wertebereich liegt zwischen 0 und 255. (Screenshot vom MEVIS VOLUMERENDERER)

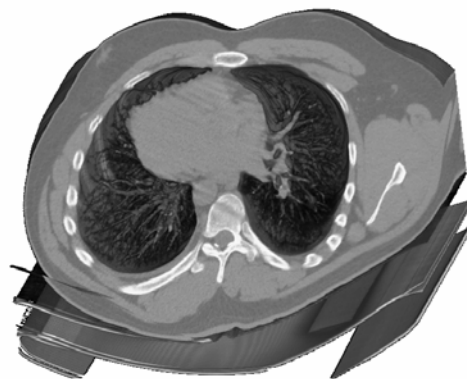


Abbildung 10: Direkte Volumenvisualisierung mit einer Grauwert- und Transparenzfensterung. Die Transferfunktion ist so gewählt worden, dass die Gefäße in der Lunge besonders gut zum Vorschein kommen. (Screenshot vom MEVIS VOLUMERENDERER)

Um dem Benutzer eine sinnvolle initiale Visualisierung anzubieten, ist es günstig, vordefinierte Funktionen zur Hervorhebung spezieller Gewebeklassen anzubieten sowie das Abspeichern zusätzlicher Funktionen zu ermöglichen. Gerade bei CT-Bildern können vordefinierte Funktionen bei anderen Datensätzen oft wieder genutzt werden oder müssen nur geringfügig skaliert und verschoben werden.

#### **Gradientenbasierte Transferfunktionen**

Transferfunktionen im engeren Sinn sind Abbildungen von Intensitätswerten auf Farb- und Transparenzwerte. Durch eine solche Transferfunktion ist es nicht möglich, Kanten zu betonen, die sich durch einen hohen Gradienten auszeichnen (vgl. Abschn. 3.4). Ein sehr guter räumlicher Eindruck entsteht häufig, wenn die Transparenz an den Gradienten bzw. an den Winkel zwischen geschätzter Normale und Lichtquelle angepasst wird. Um diese Effekte erzielen zu können, ist es nötig, ein zusätzliches Volumen mit Gradienten zu erzeugen und Transferfunktionen anzubieten, mit denen eine solche Abbildung möglich ist [FBT98].

#### **Bildbasierte Gestaltung von Transferfunktionen**

Die Definition von Transferfunktionen durch die Manipulation von Polynomen ist aufwändig und steht in keinem für den Benutzer sichtbaren Zusammenhang zu den Eigenschaften eines Bildes und den Zielen der Visualisierung. Diese Ziele sind z.B. ein kontrastreiches Bild zu erstellen, das Rauschen in den Bildern zu unterdrücken, die Kanten in den Bildern zu verstärken, die Daten zu glätten oder Strukturen hervorzuheben. Wie diese Effekte durch eine bestimmte Transferfunktion erreicht werden können, ist nicht offensichtlich. Ein erster Ansatz, um dieses Problem zu lösen, besteht darin, eine Menge von Bildern mit unterschiedlichen Transferfunktionen zu generieren und den Benutzer ein Bild (und damit eine Funktion) auswählen zu lassen [HHK96]. Dabei kann in mehreren Schritten auch eine schrittweise Verfeinerung vorgenommen werden. Allerdings ist dieser Prozess sehr (rechenzeit-)aufwändig. FANG *et al.* beschreiben, wie Bildverarbeitungsfilter in den Entwurf von Transferfunktionen integriert werden können, so dass Benutzer direkt die Parameter dieser 3D-Bildverarbeitungsfilter manipulieren [FBT98].

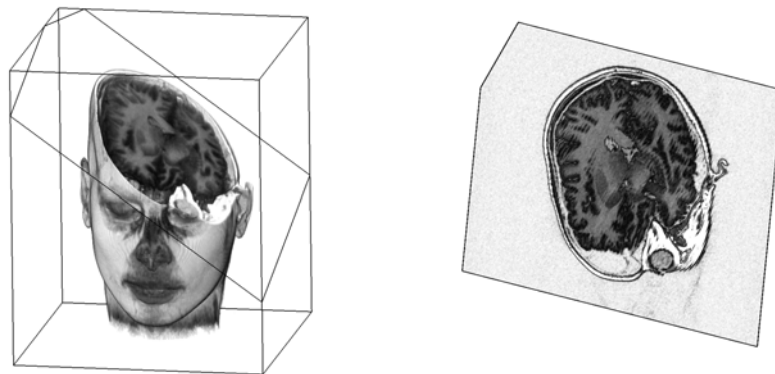


Abbildung 11: MPR-Darstellung eines MRT-Datensatzes des Kopfes. Links ist durch eine Schnittebene angedeutet, welche Schicht aus den Originaldaten herausgeschnitten wird. Rechts ist die dazugehörige Schicht dargestellt mit senkrecht zur Papirebene stehendem Normalenvektor. (Screenshot vom MEVIS VOLUMERENDERER)

### 5.3 *Multiplanare Reformattierung*

In medizinischen Anwendungen ist eine MPR-Darstellung zur Exploration der Daten weit verbreitet. Dabei wird eine Ebene in beliebiger Orientierung eines Volumendatensatzes dargestellt (Abb. 11). Im einfachsten Fall werden die der Ebene am nächsten liegenden Voxel dargestellt; eine Verbesserung ist wiederum durch eine trilineare Interpolation möglich. Auf dem so erzeugten Bild können Bildverarbeitungsoperationen (vor allem Kantenverstärkung, Glättung) angewendet werden. Für eine effiziente Darstellung kann eine modifizierte Version des Bresenham-Algorithmus genutzt werden.

### 5.4 *Einfügen von Schnittebenen*

Das Einfügen von Schnittebenen ist eine verbreitete Technik zur Exploration von Volumendaten, z.B. um tiefer liegenden Strukturen zu erkunden oder eine MPR-Ansicht zu kontrollieren. Schnittebenen können selektiv auf einzelne Strukturen oder Intensitätsbereiche angewendet werden, was zu sehr lehrreichen Visualisierungen führen kann. Schwierig ist es, dem Benutzer eine präzise und intuitive Interaktion mit Schnittebenen anzubieten. Die Visualisierung der Schnittebene selbst, die Darstellung der Handles zu deren Manipulation und die Integration exakter numerischer Eingaben sind dabei wesentlich. Eine Benennung von Schnittebenen ist Voraussetzung dafür, dass diese einzeln gelöscht, ausgeblendet oder anderweitig modifiziert werden können.

### 5.5 *Deformation von Volumenmodellen*

Für viele Anwendungen, z.B. in der rechnergestützten Simulation von Operationen, ist es nötig, Volumenmodelle mit Eigenschaften über die Elastizität anzureichern und sie zu deformieren. So können z.B. Biege-, Schneide- und Sägevorgänge simuliert werden. Dabei kommen verschiedene Verfahren zum Einsatz. Finite-Elemente-Methoden können die Deformationen sehr exakt nachbilden, sind aber sehr zeitaufwändig, so dass die Modellkomplexität für eine interaktive Handhabung stark reduziert werden muss. Thin-Plate-Splines ermöglichen eine wesentlich schnellere Approximation der Deformation, die nicht viel schlechter ist als bei Finite-Elemente-Methoden [Sch98].



Abbildung 12: Ein Voxelmodell eines Schädels (links) wird durch diskontinuierliche Deflektoren visualisiert, so daß „Einblicke“ in den Schädel entstehen (Mitte in den Mund, rechts ins Gehirn). Dabei wurde nichts herausgeschnitten, Quelle Kurzion, Yagel [1996], S. 108 bzw. S. 109

Eine elegante Methode zur Visualisierung deformierter Modelle stellen KURZION und YAGEL vor [KY96]. Sie deformieren die ins Modell gerichteten Strahlen. Diese Strahlen werden *Deflektoren* genannt. Es werden kontinuierliche und diskontinuierliche Deflek-

toren unterschieden. Kontinuierliche Deflektoren verändern das Bild so, dass etwas verbogen oder verformt wird. Diskontinuierliche Deflektoren bewirken dagegen einen Sprung im Modell, so dass ein Blick ins Innere möglich wird – ähnlich wie bei einer Cutaway-Ansicht in technischen Illustrationen (siehe Abb. 12).

### 5.6 Exploration von Ausschnitten eines Volumens

Für die Exploration großer Voxelmodelle sind mehrere Techniken entwickelt worden, in denen eine semi-transparente Darstellung für die Integration von Detail und Kontext genutzt wird [Has99]. Dazu zählt der semi-transparente Cursor, in Form eines Quaders, [ZBM96] und die Magische Kugel [CMS94]. Innerhalb eines VOIs (Volume of Interest), das durch diese Werkzeuge definiert wird, kann z.B. die Transferfunktion geändert werden und so zusätzliche Details eingeblendet werden (Abb. 13 und Abb. 14). Bei einer Isooberflächen-Darstellung kann der Teil des Volumens innerhalb des semi-transparenten Objektes in maximaler Genauigkeit dargestellt werden. Eine derart detaillierte Betrachtung von interessierenden Ausschnitten unterstützt z.B. die Selektion von Strukturen und die Abstandsmessung. Wichtig ist, dass die Detail- und die Überblicksansicht hinsichtlich der Sichtrichtung miteinander synchronisiert sind. Darüber hinaus kann es sinnvoll sein, wenn der Benutzer Sichten auf unterschiedliche Teilmengen der Daten generieren und vergleichen kann. Dies wird in [PSS00] anhand der Planung von Operationen diskutiert.

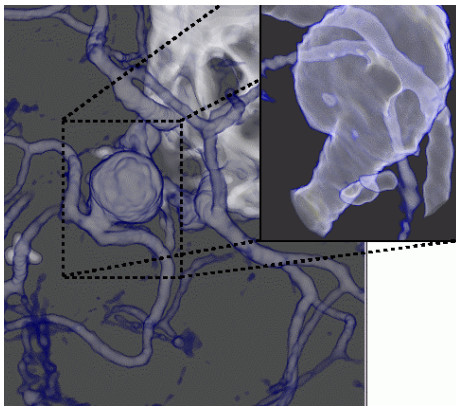


Abbildung 13: Darstellung eines Aneurysmas (einer Gefäßerweiterung) in einer CT-Angiographie des Kopfes. Die Kombination von Detail- und Kontextdarstellung erleichtert die Beurteilung der Situation. Mit freundl. Genehmigung von Peter Hastreiter, Universität Erlangen.

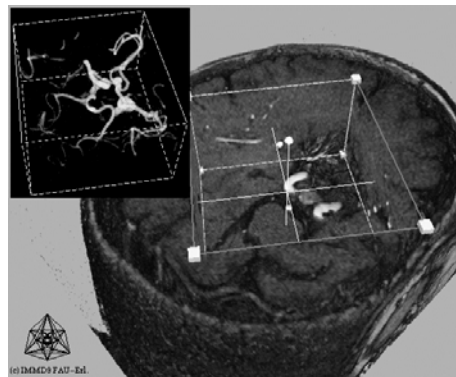


Abbildung 14: Generierung einer Detailansicht durch interaktives Boxclipping in 3D zur Beurteilung von Blutgefäßen im Gehirn. Mit freundl. Genehmigung von Peter Hastreiter, Universität Erlangen.

### 5.7 Distanz-, Abstands- und Volumenmessungen

Volumenvisualisierungen dienen sehr häufig der Präzisierung einer Diagnose oder der Planung einer Therapie. Für diese Anwendungen ist es oft essenziell, wenn quantitative Werte, insbesondere Abstände zwischen anatomischen Strukturen, Längen, Querschnitte und Volumen einzelner Strukturen zur Verfügung stehen. So ist bei dem in Abb. 13 dargestellten Aneurysma die Ausdehnung entscheidend für das weitere Vorgehen. Diese

Maße sollten direkt in die Visualisierung integriert sein, damit der Bezug deutlich wird. Abstände ergeben sich daraus, dass die Größe eines Voxels mit der Distanz zwischen den vom Benutzer selektierten Voxeln multipliziert wird. Aus der Größe eines Voxels ergibt sich der Fehler, der durch Rundung entsteht – dieser sollte ebenfalls angegeben werden (Abb. 14). Für die Volumenmessung sind verschiedene Verfahren möglich. Wenn eine Struktur explizit segmentiert wurde, kann das Volumen durch „Zählen“ der Voxel ermittelt werden. Dieser Wert ist relativ präzise, wenn die Struktur so groß ist, dass die Zahl der Randvoxel klein ist. Eine sehr grobe Volumenabschätzung ist aber auch ohne die (aufwändige) Segmentierung möglich, in dem um das zu vermessende Objekt ein umschließender Hüllkörper interaktiv so verfeinert wird, dass er relativ gut mit diesem übereinstimmt. Eine solche Volumetrie ist z.B. für die Beurteilung eines Tumors wesentlich.

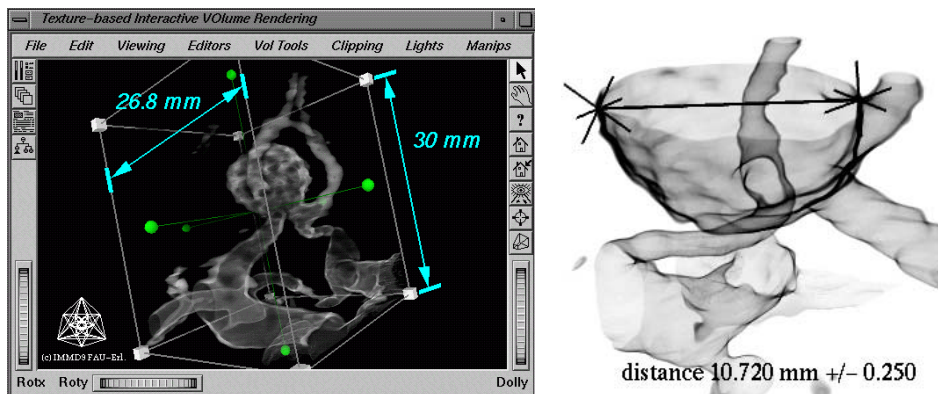


Abbildung 15: Integration von Bemaßungen in eine Volumenvisualisierung zur Präzisierung einer Diagnose, Mit freundl. Genehmigung von Peter Hastreiter, Universität Erlangen

## 6 Visualisierung registrierter Volumendatensätze

Für komplexe medizinische Fragestellungen ist es oft notwendig, die Stärken mehrerer bildgebender Verfahren zu kombinieren, z.B. CT für die Knochendarstellung und MRT für die Darstellung von Weichteilen. Um die dabei entstehenden Volumendatensätze zu beurteilen, ist es entscheidend, dass diese überlagert werden können. Dies setzt voraus, dass die gemeinsamen Merkmale in den Datensätzen erkannt werden und so ermittelt werden kann, durch welche Transformation ein Datensatz bestmöglich auf den anderen abgebildet werden kann. Dieser Vorgang wird Registrierung oder Matching genannt. Die auftretenden Transformationen und die Durchführung der Registrierung, wird u.a. in [Has99] diskutiert. Eine Möglichkeit besteht z.B. darin, dass Benutzer Landmarken definieren, die die Korrespondenz zwischen Voxeln in den Bildern bestimmen.

Bei der Überlagerung (Fusion) der so aufeinander abgebildeten Datensätze ist die Frage, wie der Benutzer einen solchen Datensatz erkunden kann. HASTREITER präsentiert dazu mehrere Ideen [Has99]. So kann der eine Datensatz dem anderen semi-transparent überlagert werden, wobei die Überlagerung auch auf einen Ausschnitt begrenzt werden kann. Eine andere Variante der Exploration der Daten besteht darin, je eine Schnittebene für jeden Datensatz zur Verfügung zu stellen, so dass der Benutzer beeinflussen kann, wie groß der Anteil ist, der von jedem Datensatz dargestellt wird.

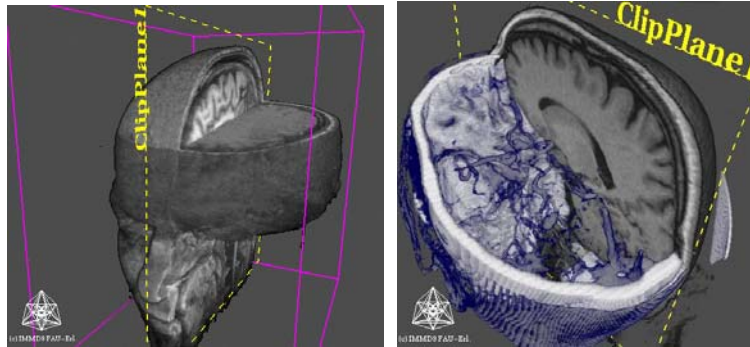


Abbildung 16: Fusion eines CT- und eines MRT-Datensatzes (links), Fusion einer MR-Angiographie zur Gefäßdarstellung mit einer MR-Tomographie (rechts). In beiden Bildern wird durch Schnittebenen die Überlagerung im Bild gesteuert. Mit freundl. Genehmigung von Peter Hastreiter, Universität Erlangen.

Eine wichtige Anwendung für die Registrierung von Volumendaten betrifft die Unterstützung von Chirurgen während einer Operation durch Augmented Reality-Techniken. Dazu müssen Videobilder der laufenden Operation mit den Daten aus der OP-Vorbereitung, z.B. CT-Daten, registriert werden. Wenn dies gelingt, kann z.B. der Tumor, der entfernt werden soll, in das Operationsgebiet projiziert werden, so dass der Chirurg erkennt, wie er sich zu einem solchen Tumor vorarbeiten kann.

## 7 Werkzeuge zur Volumenvisualisierung

Die Übersicht über die Werkzeuge zur Volumenvisualisierung stützt sich vor allem auf Demonstrationsversionen und berücksichtigt folgende Systeme: das frei verfügbare VOLVIS-System, den VOLUMIZER (von Silicon Graphics), die VOXLIB von VOXAR, VGL von VOLUME GRAPHICS, VOXELVIEW von VITAL IMAGES, ANALYZE von der Mayo-Clinik und die Graphikkarte VOLUME PRO 500 von MITSUBISHI. Dabei werden die verwendeten Renderingverfahren und die Stärken der einzelnen Systeme betrachtet. Aufgrund der starken Abhängigkeit der Performance von der konkreten Hardware (Vorhandensein und Größe von Texturspeichern, hardwareunterstützten Lookup-Tabellen) etc. werden keine Benchmarks integriert.

Alle im Folgenden kurz vorgestellten Bibliotheken sind recht ausgereift. Bezüglich der Rendering-Verfahren ist ein gemeinsamer Kern in allen Bibliotheken – mit Ausnahme der VOLUME PRO 500-Karte – realisiert. Dazu zählen: MIP, MPR, direktes Volumenrendering semi-transparenter Daten (Rendering von Grauwert- und Farbbildern), Darstellung der Volumendaten in Parallel- und perspektivischer Projektion. Wie die direkte Volumenvisualisierung realisiert ist, wird nicht in allen Fällen beschrieben; in den meisten Fällen handelt es sich um Raycasting-Verfahren. Alle Bibliotheken unterstützen auch die Extraktion und das Rendering von Isooberflächen. In allen Bibliotheken ist außerdem die Manipulation von Lookup-Tabellen möglich. Aufgrund dieser Gemeinsamkeiten wird im folgenden nur erwähnt, welche Verfahren über diesen gemeinsamen Kern hinausgehen.

Bei allen beschriebenen Bibliotheken ist die Qualität der entstehenden Bilder hoch. Die Unterschiede sind recht subtil und beziehen sich auf die Flexibilität, mit der die Verfahren parametrisiert werden können und auf zusätzliche Möglichkeiten zur Visualisie-

rung bzw. Analyse von Volumendaten. Unterschiede gibt es auch in der Effizienz der Visualisierung, wobei diese relativ schwer zu beurteilen ist, da die Effizienz von einer Vielzahl von Parametern der Hardware abhängt. Für den praktischen Einsatz sind weitere Merkmale, wie die Verfügbarkeit auf bestimmten Plattformen, die unterstützten Datenformate, die Exportmöglichkeiten und die Kosten wesentlich. Da sich diese Merkmale besonders schnell ändern, wird dazu auf die WWW-Seiten der Hersteller verwiesen. Alle Systeme sind dort ausführlich beschrieben. Auf den meisten Web-Sites können (kostenlose) Demoversionen angefordert oder heruntergeladen werden.

### 7.1 VolVis

**Anbieter:** Visualization Laboratory at the State University of New York at Stony Brook  
**WWW-Adresse:** <http://www.mayo.edu/bir/>

Das frei verfügbare VOLVIS-System, das von der Gruppe um Prof. KAUFMAN entwickelt wurde (AVILA *et al.* [ASK92] und [AHH94]) umfasst eine breite Palette von Rendering-Verfahren. VOLVIS besteht aus fünf Bibliotheken für die Ein- und Ausgabe von Volumendaten, für Filteroperationen, das Rendering und mathematische Operationen auf Volumen. Oberflächengeometrie und Volumendaten können beliebig kombiniert werden; Außerdem ist die (direkte) Visualisierung unregelmäßiger, nicht quaderförmiger Gitter möglich. VOLVIS ist sehr flexibel bzgl. der Qualität und Geschwindigkeit der Visualisierung, aber generell langsam, da alle Verfahren in Software realisiert sind.

**Renderingverfahren.** Es werden folgende Renderingverfahren unterstützt: Raytracing, Radiosity und Splatting und direktes Rendern komprimierter Voxeldaten.

**Datenstrukturen.** VOLVIS ist objektorientiert realisiert, wobei die Low-Level-Klassen (Punkte, Vektoren, Matrizen) und die High-Level-Klassen (Lichtquellen, Sichten, Volumen) wesentlich sind. Diese Datenstrukturen sind Bestandteil der meisten APIs zur Volumenvisualisierung.

**Interaktion.** VOLVIS steht auch als interaktives System zur Verfügung. Es erlaubt die Eingabe mit verschiedensten physischen Zeigergeräten, z.B. (3D-) Space-Maus und Datenhandschuh, wobei auch mehrere Geräte parallel genutzt werden können. Die Modellierung mit volumenhaften Daten geschieht mit CSG (Constructive Solid Geometry)-Operationen, so dass Daten „umgehängt“ und kopiert werden können. Die Manipulationskomponente dient der Einstellung einer beliebigen Sicht und Beleuchtung sowie der Transformation des Volumens. Für die interaktive Komponente wird eine Isoberflächen-Darstellung des Volumens adaptiv in eine niedrigere Auflösung konvertiert. Diese Konvertierung wird auch zur Beschleunigung von Raytracing und Radiosity genutzt.

### 7.2 Volumizer

**Anbieter:** Silicon Graphics **WWW:** <http://www.sgi.com/software/volumizer>

**Konzepte.** Der VOLUMIZER basiert auf der extensiven Nutzung der Hardwareunterstützung für das Texturemapping und stellt eine Schicht auf OpenGL dar. Wenn eine Hardwareunterstützung für 3D-Texturemapping vorhanden ist, wird diese genutzt; andernfalls muss eine Unterstützung für 2D-Texturemapping verfügbar sein.

So wie beim polygonalen Rendering eine Zerlegung in Dreiecke stattfindet, werden beim VOLUMIZER Volumen in Tetraeder zerlegt und das Volumenrendering auf die Texturierung von Polygonen, die die Tetraeder schneiden, zurückgeführt. Die Repräsentation in Tetraedern hat den Vorteil, das Volumen deformiert werden können, indem die Eckpunkte der Tetraeder manipuliert werden, wobei die Originaldaten unverändert bleiben.

**Rendering-Verfahren.** Der VOLUMIZER, aktuelle Version 1.2, unterstützt die Visualisierung der 3D-Texturen, wobei unterschiedliche Interpolationen und Shading möglich sind. Eine beliebige Kombination mehrerer Volumen sowie von Volumen und (transparenten) Oberflächenobjekten ist möglich.

**Trennung von Geometrie und Erscheinung.** Beim VOLUMIZER wird eine *Geometrie* durch Primitive, wie Kugeln oder Tetraeder, und die *Erscheinung* durch Texturen definiert. Die Texturen (definiert durch die Voxel) können auf beliebige Geometrien abgebildet werden. So können Volumen (die Geometrie) durch Strecken, Stauchen und anderweitiges Verformen der 3D-Textur deformiert werden, ohne dass die Voxel manipuliert werden. Dieses Konzept hat viele Vorteile; ist aber gewöhnungsbedürftig, weil die Voxel damit nur in Kombination mit einer Geometrie darstellbar sind. Teile der Geometrie können in unterschiedlicher Qualität (z.B. feinere Unterteilung) und mit unterschiedlichen Materialeigenschaften dargestellt werden.

**API.** Der VOLUMIZER kann zusammen mit anderen SGI-Bibliotheken, wie OPEN INVENTOR und PERFORMER benutzt werden. Die jeweils günstigsten Parameter für das interne Datenformat, die Abtastrate, die Bricks, in die das Volumen zerlegt wird, sind stark abhängig von der Größe des Texturspeichers bzw. der Interpolation. Die Funktion *getBestParameters*, wählt die auf der jeweiligen Hardware günstigsten Parameter.

Die API ist umfangreich und gut dokumentiert. Die Programmierung kann auf der Basis von OpenGL (eher prozedural) oder OPENINVENTOR (eher objekt-orientiert) erfolgen. Für die Definition von Texturen sind eine Reihe von Klassen vorhanden, deren Benutzung gewöhnungsbedürftig ist. Konzeptionell sind die API der VGL und von der VOXLIB einfacher. Der OpenGL Textur-Manager verwaltet die Texturkoordinaten.

### 7.3 **Analyze und AVW**

**Anbieter:** Mayo-Klinik, Rochester

**WWW-Adresse:** <http://www.mayo.edu/bir/>

Analyze (aktuelle Version: 7.5.4)

- interaktives System zur Exploration, Analyse und Visualisierung mehrdimensionaler Daten, Schwerpunkt: Bildverarbeitung (nicht Volumenvisualisierung),

AVW (A Visualization Workshop)

- Sammlung von Bibliotheken, die die Algorithmen als API zur Verfügung stellen

Grundfunktionen (beider Produkte):

- Registrierung von mehreren Bildern und Volumen (auf der Basis von Landmarken),
- Kombinierte Darstellung von Regionen, die mit unterschiedlichen Rendering-Parametern generiert wurden,
- Glättung und Kontrastverstärkung auf der Basis adaptiver lokaler Verfahren,
- Segmentierung und Vermessung,
- Histogrammoperationen und Filter auf Bildern und Volumendaten



**Renderingverfahren.** Es werden folgende Verfahren angeboten: grey level gradient shading, MIP (mit optionaler Wichtung der Tiefe, so dass auch räumliche Effekte erkennbar werden), average projection und Stereorendering. Außerdem ist eine Maskierung von Bereichen beim Rendering möglich.

**Zusammenfassung.** Das Analyze-Tool und die dazugehörige API AVW sind besonders umfassend. Die Stärke dieser Systeme liegt in den umfassenden Analysemöglichkeiten und der Integration von Bildverarbeitung und Volumenvisualisierung. Die Volumenvisualisierung ist dabei nur ein Teil unter einer Vielzahl von Modulen. Das System wirkt insgesamt ausgereift und deckt eine breite Palette von Anwendungen ab. Die API ist nicht objekt-orientiert und ist softwaretechnologisch nicht auf dem neuesten Stand.

#### 7.4 VoxLib

**Anbieter:** Voxar, Edinburgh      **WWW:** <http://www.voxar.com>

##### **Rendering-Verfahren.**

- Extraktion von scharfen Oberflächen und von sogenannten Fuzzy-Oberflächen, die nicht auf einen exakten Intensitätswert beschränkt sind, sondern ein Intervall von Intensitätswerten darstellen,
- Rendern von Punktwolken,
- Kombination von Oberflächen- und Volumendaten sowie von mehreren Volumen durch Rendern in einen gemeinsamen Z-Buffer,
- Voxelization von Polygonen (einschließlich von Normalen und Materialeigenschaften)

**Highlights.** Die Klasse *VxrContinuation* kapselt eine langwierige Aktion in kleine Schritte (sollten höchstens 0,5 Sekunden dauern), die sukzessive abgearbeitet werden und nach denen eine Unterbrechung möglich ist. Dabei wird der Stand der Abarbeitung verwaltet. Diese Klasse stellt eine Alternative zur Nutzung mehrerer Threads dar, weil auf diese Weise bei einer längeren Aktion auf Benutzereingaben reagiert werden kann. Eine Rendering-Spezifikation kann um die Parameter *Quality* und *TimeBudget* ergänzt werden. Der *Quality*-Parameter ist ein Wert zwischen 0 und 1 (Rendering in der besten Qualität), wobei die Performance etwa linear von diesem Wert abhängig ist. Der *TimeBudget*-Parameter ermöglicht es, eine Framerate einzustellen (z.B. 10 fps), wobei die Rendering-Parameter so angepasst werden, dass dieser Wert etwa erreicht wird.

**Zusammenfassung.** Die VOXLIB basiert auf umfangreichen Erfahrungen im Bereich der medizinischen Visualisierung und ist über Jahre hinweg verfeinert worden. Die VOXLIB ist gut für eine Kombination mit anderen Bildverarbeitungsalgorithmen geeignet und enthält bereits einige dieser Algorithmen. Das Vorhandensein von Klassen für die Vermessung von Volumendaten ist ein besonderes Plus. Die objekt-orientierte API und ihre Beschreibung sind vorbildlich strukturiert. Einige aufwändige, aber qualitativ hochwertige Verfahren, wie Schattenprojektion und Stereodarstellung, fehlen.

#### 7.5 Volume Graphics Library VGL

**Anbieter:** Volume Graphics, Heidelberg,  
**WWW:** <http://www.volumegraphics.com/vgl.html>

**Renderingverfahren.** Die VGL, aktuelle Version 2.1, unterstützt folgende Verfahren: Darstellung mehrerer Volumen auch in Kombination mit Oberflächenmodellen, Ray-tracing, Erzeugung realistischer Schatten, Rendering in mehreren Threads und Stereo-rendering. Alle Rendering-Algorithmen können in ihrer Auflösung durch die Methode *setDepthResolution* (Zahl der Abtastwerte in z-Richtung) beeinflusst werden.

**Ankündigungen.** Die für das 1. Quartal 2000 angekündigte Version 3 bietet neue Verfahren auf der Basis von Texturemapping-Hardware, eine Nutzung von OpenGL-Hardware-Implementierungen und eine Unterstützung für die VOLUMEPRO 500-Graphikkarte.

### 7.6 *VoxelView*

**Anbieter:** Vital Images, Minneapolis      **Version:** 2.5.4

**WWW:** <http://www.vitalimages.com/products/voxelview.shtml>

**Renderingverfahren.** Shading der Daten, Antialiasing, Kombination und Koordinierung von 2D- und 3D-Ansichten.

**Highlights.** Umfangreiche 3D-Bildverarbeitung zur Vorverarbeitung der Daten stehen zur Verfügung. Der Zugriff und die Manipulation von Voxeldaten durch mehrere Prozesse wird durch eine Shared-Memory-Verwaltung und Semaphore unterstützt. Es können zwei Monitore angeschlossen werden.

### 7.7 *Volume Pro 500*

**Anbieter:** Mitsubishi      **WWW:** <http://www.rtviz.com>

**Rendering-Verfahren.** Die im Mai 1999 vorgestellte VOLUME PRO 500-Karte ist eine dedizierte Hardware für die Volumenvisualisierung, die auf PCs als Ergänzung zu einer „normalen“ Graphik-Karte genutzt wird. Die Rendering-Geschwindigkeit liegt mit 500 MVoxel/s etwa Faktor 10 über der der besten vergleichbaren Systeme. Damit ist erstmals eine Realtime-Darstellung hochauflöser Datensätze möglich. Diese Geschwindigkeit wird auch erreicht, wenn die Volumendaten durch Lichtquellen beleuchtet werden.

**Einschränkungen.** Die Darstellung ist auf Parallelprojektion und auf DVR-Verfahren beschränkt. Durch das Fehlen einer perspektivischen Projektion ist die Graphikkarte vor allem für die virtuelle Endoskopie (Navigation durch Teile des menschlichen Körpers) nicht geeignet. Eine Überabtastung zur Steigerung der Qualität ist zwar möglich, verringert aber die Darstellungsgeschwindigkeit unverhältnismäßig.

## 8 Zusammenfassung und Ausblick

Die Verfahren zur medizinischen Volumenvisualisierung haben in der jüngsten Vergangenheit Einzug in die radiologischen Workstations der großen Hersteller und damit in die medizinische Praxis gehalten. Die vor wenigen Jahren noch vorherrschenden indirekten Verfahren, bei der als Zwischenrepräsentation eine Oberflächendarstellung generiert wurde, verlieren mehr und mehr an Bedeutung, da eine Hardwareunterstützung für die direkte Volumenvisualisierung Verbreitung findet. Dadurch werden die DVR-Verfahren schneller und die Qualität der entstehenden Bilder (Beleuchtung) besser. Das Bestreben die Effizienz der Verfahren – vor allem durch dedizierte Hardwareunterstützung – zu ver-

bessern, wird auch die kommenden Jahre prägen, da sich die Auflösung der bildgebenden Verfahren und damit die anfallende Datenmenge deutlich erhöht. Ein absehbarer Trend ist die verstärkte Überlagerung mehrerer Volumendatensätze, wie sie z.B. zur Beurteilung von Krankheitsverläufen wichtig ist.

Zugleich gewinnen die Interaktionstechniken zur Handhabung von Volumendaten an Bedeutung. Die Herausforderung besteht dabei darin, eine intuitive 3D-Interaktion mit präzisen numerischen Eingaben zu verbinden. Sinnvolle Voreinstellungen, vor allem für die sensible Einstellung von Transferfunktionen, und die Möglichkeit benutzerspezifische Präferenzen für relevante Parameter zu speichern, werden eine Schlüsselrolle dabei spielen, Volumenvisualisierungsverfahren in der klinischen Praxis effizient einzusetzen.

## Danksagung

Wir möchten uns bei unseren Kollegen für die Beiträge zur Implementierung von Komponenten des Volumenrenders bedanken: bei Andrea Schenk (Realisierung der MPR), bei Holger Bettag (Realisierung eines Editors für Lookup-Tabellen) und bei Sven Kohle (Realisierung der Closest-Vessel-Projection).

## Literatur:

- [ASK92] R. Avila, L. Sobieraski und A. Kaufman. „Towards a Comprehensive Volume Visualization System“, *IEEE Visualization*, IEEE Computer Society Press, Oktober, S. 13-20
- [AHH94] R. Avila, T. He, L. Hong, A. Kaufman et al. „VolVis: A Diversified Volume Visualization System“, *IEEE Visualization*, IEEE Computer Society Press, Washington, DC, Oktober, S. 31-43
- [CCF94] B. Cabral, N. Cam und J. Foran. „Accelerated Volume Rendering and Tomographic Reconstruction Using Texture Mapping Hardware“, *1994 Symposium on Volume Visualization*, ACM SIGGRAPH (Washington, Oktober), S. 91-98
- [CLL88] H. E. Cline, W. E. Lorensen, S. Ludke, C. R. Crawford und B. C. Teeter. „Two algorithms for three-dimensional reconstruction of tomograms“, *Medical Physics*, Band 15(3), S. 320-327
- [CMS94] P. Cignoni, C. Montani und R. Scopigno. „Magic Sphere: an Insight Tool for 3D data visualization“, *Proc. of Eurographics* (Oslo, 12.-16. September), Computer Graphics Forum, Band 13 (3), S. 317-328
- [Elv92] T. T. Elvins. „A Survey of Algorithms for Volume Visualization“, *Computer Graphics*, Band 26 (3), 23 Seiten
- [FBT98] S. Fang, T. Biddlecome und M. Tuceryan. „Image-Based Transfer Function Design for Data Exploration in Volume Visualization“, *Proc. of IEEE Visualization*, S. 319-326
- [GR85] D. Gordon und R. A. Reynolds. „Image-Based Shading of 3-Dimensional Objects“, *Computer Graphics and Image Processing*, Band 29 (3), S. 361-376
- [Has99] P. Hastreiter. *Registrierung und Visualisierung medizinischer Bilddaten unterschiedlicher Modalitäten*, Dissertation, Technische Fakultät, Friedrich-Alexander-Universität Erlangen-Nürnberg
- [HHK96] T. He, L. Hong, A. Kaufman und H. P. Pfister. „Generation of transfer functions with stochastic search techniques“, *Proc. of IEEE Visualization*, Oktober, S. 227-234
- [Hop96] H. Hoppe. „Progressive Meshes“, *Proc. of SIGGRAPH* (New Orleans, LA, Juli), ACM SIGGRAPH, S. 99-108

- [Lev90] M. Levoy. „Efficient ray tracing of volume data“, *ACM Transactions on Graphics*, Band 9 (3), Juli 1990, S. 245-261
- [Lev90b] M. Levoy. „Volume Rendering by Adaptive Refinement“, *The Visual Computer*, Band 6 (1), Februar, S.2-7
- [LL94] P. Lacroute und M. Levoy. „Fast Volume Rendering Using a Shear-Warp Factorization of the Viewing Transformation“, *Proc. of SIGGRAPH* (Orlando, Florida, 24.-29. Juli), ACM SIGGRAPH, S. 451-458
- [LH91] D. Laur und P. Hanrahan. „Hierarchical Splatting: A Progressive Refinement Approach for Volume Rendering“, *Proc. of SIGGRAPH*, ACM SIGGRAPH, S. 285-288
- [LC87] W. E. Lorensen und H. E. Cline. „Marching Cubes: A high Resolution 3D Reconstruction Algorithm“, *Proc. of SIGGRAPH*, ACM SIGGRAPH, S. 163-169
- [PSS00] B. Preim, W. Spindler, D. Selle und H. O. Peitgen. „Visuelle Simulation und Analyse Planung onkologischer Operationen“, in diesem Band.
- [PTH99] B. Pflesser, U. Tieder und K.H. Höhne. „Simulation von Schnittoperationen in medizinischen Volumenmodellen“, *Bildverarbeitung für die Medizin – Algorithmen, Systeme, Anwendungen* (Heidelberg, März), Informatik Aktuell, Springer-Verlag, S. 182-186
- [Sch98] T. Schiemann (1998) *Interaktive Verfahren für deformierende Eingriffe an volumenbasierten digitalen Körpermodellen*, Dissertation, Universitätskrankenhaus Hamburg-Eppendorf
- [SZL92] W. J. Schröder, J. A. Zarge und W. E. Lorensen. „Decimation of Triangle Meshes“, *Proc. of SIGGRAPH'92* (Chicago, Juli), ACM SIGGRAPH, S. 65-70
- [WE98] R. Westermann und T. Ertl. „Efficiently Using Graphics Hardware in Volume Rendering Applications“, *Proc. of SIGGRAPH*, ACM SIGGRAPH, S. 169-179
- [Wes90] L. Westover. „Footprint Evaluation for Volume Rendering“, *Proc. of SIGGRAPH* (Dallas, Juli), ACM SIGGRAPH, S. 367-376
- [YKZ91] R. Yagel, A. Kaufman und Q. Zhang. „Realistic Volume Viewing“, *Proc. of Visualization '91*, San Diego, S. 226-231
- [YCK92] R. Yagel, D. Cohen und A. Kaufman. „Discrete Raytracing“, *IEEE Computer Graphics and Applications*, Band 12 (5), S. 19-28
- [YK92] R. Yagel und A. Kaufman. „Template-Based Volume Viewing“, *Proc. of Eurographics* (Cambridge UK, September), Computer Graphics Forum, Band 11 (3), S. 153-157
- [YKT96] R. Yagel, A. Kaufman und U. Tiede. *Classification and Survey of Algorithms for Volume Viewing*, Tutorial Notes, Visualization of Biomedical Computing (Hamburg, September)
- [ZBM96] S. Zhai, W. Buxton und P. Milgram. „The partial occlusion effect: utilizing semi-transparency in 3D human computer interaction“, *ACM Transactions on HCI*, Band 3 (3), S. 254-284
- [ZKV95] K. Zuiderveld, A. H. Konig und M. Viergever. „Techniques for Speeding Up high-quality perspective maximum-intensity projection“, *Pattern Recognition Letters*, Band 15, S. 507-517