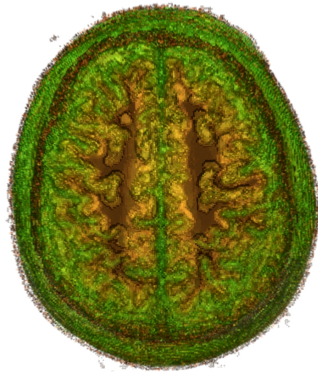




— Diplomarbeit —

Entwurf eines multimodalen Volumenvisualisierungssystems

Bakkalaureus für Computervisualistik Matthias Keil
Matrikelnummer 160490



4. November 2005

1. Gutachter: Prof. Dr.-Ing. Bernhard Preim*

2. Gutachter: Prof. Dr.-Ing. Georgios Sakas**

Betreuer: Dipl.-Math. Evelyn Firle** und

Dipl.-Phys. Stefan Wesarg**

Bearbeitungszeitraum: 01.07.2005 – 15.11.2005

*Institut für Simulation und Graphik, Fakultät für Informatik, Otto-von-Guericke Universität,
Magdeburg

**Cognitive Computing & Medical Imaging, Fraunhofer Institut für Graphische Datenverarbeitung (IGD), Darmstadt

Danksagung

Bedanken möchte ich mich bei Evelyn Firlé und Stefan Wesarg vom Fraunhofer IGD in Darmstadt, die mir diese Diplomarbeit ermöglicht haben. Vor allem Evelyn gebührt Dank dafür, dass sie trotz ihrer Schwangerschaft und später ihren drei kleinen Kindern Zeit für mich und meine Probleme hatte. Stefan möchte ich für die Hilfe zur Selbsthilfe danken.

Danke auch an alle Lehrenden der Fakultät für Informatik an der Otto-von-Guericke Universität in Magdeburg. Ganz besonders möchte ich Prof. Bernhard Preim dafür danken, dass er meine externe Diplomarbeit betreut hat. Danke auch an meine Kommilitonen für interessante Unterhaltungen, fröhliche Lernstunden und freundschaftliche Abende.

Ich möchte mich ganz herzlich bei meiner Verlobten Carina Deutsch bedanken. Sie hat mich unterstützt wo es nur ging und mir immer Mut gemacht. Ich liebe Dich!

Weiterhin danke ich meinen Eltern und Großeltern, die mir dieses Studium ermöglicht haben sowie meinen Schwiegereltern für die ermahnenden Worte zu Beginn des Studiums und die Unterstützung über die fünf Jahre.

Vielen Dank, Matthias Keil

Inhaltsverzeichnis

1. Motivation und Aufbau	1
2. Bildgebende Verfahren für die medizinische Registrierung	3
2.1. Anforderungen an die verwendeten Volumendatensätze und den Nutzer	4
2.2. Grundlagen der Computertomographie	4
2.2.1. Bildgebung mittels Computertomographie	4
2.2.2. Verwendete Computertomographiedaten	7
2.3. Grundlagen der Positronen Emissions Tomographie	10
2.3.1. Bildgebung mittels Positronen Emissions Tomographie	10
2.3.2. Verwendete Positronen Emissions Tomographie Daten	11
2.4. Grundlagen der Magnetresonanztomographie	13
2.4.1. Bildgebung mittels Magnetresonanztomographie	13
2.4.2. Verwendete Magnetresonanztomographiedaten	15
2.5. Vom Messsignal zum Volumendatensatz	19
2.5.1. Bildrekonstruktion	19
2.5.2. Partialvolumeneffekt	21
2.6. Registrierung von medizinischen Datensätzen	22
3. Multimodale Visualisierung medizinischer Volumendatensätze	26
3.1. Anforderungen an die zu erzeugende Visualisierung	26
3.2. Schichtbilder und deren Eignung für das Visualisierungssystem	27
3.3. Indirekte Volumenvisualisierung und deren Eignung für das Visualisierungssystem	30
3.4. Direkte Volumenvisualisierung	34
3.4.1. Direkte monomodale Volumenvisualisierung	34
3.4.2. Direkte multimodale Volumenvisualisierung	35
3.4.3. Pipelines der direkten multimodalen Volumenvisualisierung	36
3.4.4. Eignung der direkten multimodalen Volumenvisualisierung für das Visualisierungssystem	40
3.5. Rendering	41
3.5.1. Renderingverfahren	41
3.5.2. Beleuchtungs- und Schattenberechnung	45
3.5.3. Eignung der Renderingverfahren für multimodale Daten	46

4. Transferfunktionen	48
4.1. Anforderungen an die verwendeten Transferfunktionen	48
4.2. Eindimensionale Transferfunktionen	49
4.3. Gradientenbasierte mehrdimensionale Transferfunktionen	50
4.3.1. Verwendetes Kantenmodell	51
4.3.2. Definition der Gradienteninformation	52
4.3.3. Histogramme über Intensität und Gradientenstärke	54
4.4. Spatialized Transfer Functions	60
4.4.1. Automatische Generierung der Ortsinformation	61
4.4.2. 2D-Histogramme für die Spatialized Transfer Functions	62
4.4.3. Interaktion mit der Transferfunktion	65
4.4.4. Spatialized Transfer Functions für multimodale Visualisierungen .	66
4.5. Spezifikation von Transferfunktionen	67
4.5.1. Passive Hilfsmittel für die Spezifikation	67
4.5.2. Interaktive Hilfsmittel für die Spezifikation	69
5. Entwurf und Implementierung des Visualisierungssystems	70
5.1. Entwurf des multimodalen Visualisierungssystems	70
5.2. Spatialized Transfer Functions für die multimodale Volumenvisualisierung	73
5.3. Implementierung der Visualisierungskomponente	76
5.3.1. Allgemeines	76
5.3.2. Vorverarbeitung der Daten	84
5.3.3. Generierung der 2D-Histogramme und der Ortsinformationen	88
5.3.4. Klassifikation und Darstellung der 2D-Histogramme	92
5.3.5. Implementierung der Spatialized Transfer Functions	94
5.3.6. Anwendung der Transformation und Berechnung der sich über-	
schneidenden Regionen	97
5.3.7. Rendering	100
6. Ergebnisse	101
6.1. Gradientenbasierte Visualisierungen einzelner Volumene	101
6.1.1. Monomodale Visualisierung eines MRT-Datensatzes	102
6.1.2. Monomodale Visualisierung eines PET-Datensatzes	103
6.2. Vergleich 3D-Visualisierung mit Schachbrettmuster-Schichtdarstellung . .	104
6.3. Darstellung von Strukturen mit und ohne anatomischen Kontext	107
6.4. Vergleich von Visualisierungen mit und ohne Überabtastung	111
6.5. Vergleich multiintensitätsbasierte und gradientenbasierte Visualisierung	115
6.5.1. Visualisierung mit gradientenbasierten STFs	116
6.5.2. Visualisierung mit multiintensitätsbasierter STF	117
7. Diskussion und Ausblick	119
A. Anhang	122
Verzeichnis der verwendeten Abkürzungen	129
Abbildungsverzeichnis	132
Literaturverzeichnis	138
Index	141

1. Motivation und Aufbau

The demand for ... [multimodal visualizations] is rapidly increasing in scientific applications such as medicine in which the visual integration of multiple modalities allows a better comprehension of the anatomy and a perception of its relationships with activity.

MARIA FERRÉ, ANNA PUIG AND
DANI TOST, AUS [FPT02]

Motivation

Durch medizinische Bildgebung ist es den Ärzten und Forschern möglich geworden, Einblick ins Innere des menschlichen Körpers zu bekommen. Ungeachtet der in den letzten Jahren erzielten Fortschritte haben die in der klinischen Diagnostik verwendeten bildgebenden Verfahren, wie Computertomographie, Magnetresonanztomographie und Positronen Emissions Tomographie nach wie vor grundlegende Vor- und Nachteile. Eine gemeinsame Darstellung von Datensätzen des gleichen Objekts, die zudem mit verschiedenen Modalitäten aufgenommen wurden (multimodale Visualisierung), ermöglicht die Nutzung der Vorteile und ebenso eine Kompensation der Nachteile. Weiterhin können komplementäre Informationen, die sich aus den speziellen Abbildungseigenschaften der Verfahren ergeben, gemeinsam präsentiert werden. Eine dreidimensionale Visualisierung bietet zudem einen natürlichen Blick auf das Objekt.

Eine multimodale Abbildung kann selbstverständlich auch zwei Volumen der gleichen Modalität zeigen (sog. Multivolumen-Visualisierung), die zu unterschiedlichen Zeitpunkten aufgenommen wurden. Unabhängig von den verwendeten bildgebenden Verfahren müssen die Datensätze zunächst durch eine Registrierung zueinander im Raum ausgerichtet werden. Die Vereinigung der Datensätze erfolgt in der anschließenden Multivolumen-Visualisierung. Die Genauigkeit der Registrierung hat dabei einen hohen Einfluss auf das Visualisierungsergebnis.

Im Rahmen dieser Arbeit soll für die am Fraunhofer Institut für Graphische Datenverarbeitung (IGD) entwickelte Software für die medizinische Bildverarbeitung MIP (Medical Imaging Platform) ein multimodales Volumenvisualisierungssystem entworfen und

implementiert werden. Bisher stand innerhalb dieser Anwendung keine 3D-Darstellung der Registrierungsergebnisse zur Verfügung. Mit Hilfe einer Visualisierung sollen relevante Strukturen der Datensätze hervorgehoben und vergleichbar gemacht werden. Weiterhin soll die Identifikation sowie die Unterscheidbarkeit und Klassifikation der Organe möglich sein. Für diese Arbeit ergeben sich außerdem spezielle Anforderungen an die Visualisierung, die im Folgenden formuliert werden.

Das zu entwickelnde Visualisierungstool ist an die Anforderungen von Registrierung und multimodalen Volumendaten anzupassen. Ziel ist es zwei Datensätze unterschiedlicher bildgebender Verfahren unabhängig von der aufgenommenen Körperregion darzustellen. Anhand einer dreidimensionalen Visualisierung soll das Ergebnis der Registrierung veranschaulicht werden. Dazu ist es zum einen erforderlich, dass nur die relevanten Strukturen eines Datensatzes isoliert darstellbar sind. Zum anderen müssen auch Kontextinformationen aus dem gleichen Datensatz für eine Lagebestimmung hinzugefügt werden können. Fehlende Information soll mit Hilfe von komplementären Strukturen aus einem Datensatz anderer Modalität kompensiert werden.

Aufbau der Diplomarbeit

In Anlehnung an die Arbeit von Puig, Ferré et al. [FPT02] werden für den Entwurf des multimodalen Volumenvisualisierungssystems die folgenden Punkte untersucht. Anhand der bildgebenden Verfahren werden zunächst die wichtigen Eigenschaften der verwendeten Daten herausgearbeitet, die das Ergebnis der multimodalen Visualisierung beeinflussen. In diesem Zusammenhang erfolgt die Analyse der Vor- und Nachteile der jeweiligen Modalität sowie der in den Bilddaten auftretenden Artefakte. Da die Visualisierungsmethoden vielfältig und für multimodale Visualisierungen unterschiedlich gut geeignet sind, wird im Anschluss eine Auswahl von Methoden der multimodalen Visualisierung vorgestellt. Die verschiedenen Ansätze werden analysiert und anhand der gestellten Anforderungen für die Umsetzung ausgewählt. Daraufhin wird ein Überblick über Transferfunktionen und deren Spezifikation gegeben und im Besonderen auf die speziellen Charakteristika der in dieser Arbeit verwendeten *Spatialized Transfer Functions* eingegangen. Transferfunktionen bilden die Messwerte eines Datensatzes oder daraus berechnete Merkmale auf optische Eigenschaften wie Farbe und Transparenz ab. Nach einer kurzen Darstellung möglicher Renderingalgorithmen werden der Entwurf für die Systemerweiterung und die Implementierung im Detail vorgestellt. Die mit dem implementierten Visualisierungssystem erzeugten multimodalen Volumenrenderings werden analysiert und bewertet. Anhand der Ziele erfolgt außerdem eine Diskussion über die Leistungsfähigkeit des Systems. Weiterhin werden Anregungen für die Optimierung und Weiterentwicklung der Komponente in folgenden Arbeiten gegeben.

2. Bildgebende Verfahren für die medizinische Registrierung

Imaging of this quality, sharpness, and speed gives us the opportunity to study the human anatomy at a level that has only been dreamt about.

PROF. DR. WERNER A. BAUTZ,
RADIOLOGISCHES INSTITUT,
UNIVERSITÄT
ERLANGEN-NÜRNBERG

Einer Visualisierung von medizinischen Bilddaten geht immer eine Bildaufnahme voraus. Es ist wichtig, die Eigenschaften der einzelnen bildgebenden Verfahren bei der späteren Visualisierung zu beachten. Je nach Verfahren können einige Gewebearten unterschiedlich gut dargestellt werden. Zum Beispiel ist es nicht möglich Knochen in Magnetresonanztomographiebildern darzustellen, andererseits ist dafür der Weichteilkontrast um ein vielfaches höher als zum Beispiel bei der Computertomographie. Auch die Auflösung und damit die zu verarbeitende Datenmenge variiert zwischen den Modalitäten. Jedes Verfahren hat somit charakteristische Merkmale, die vor allem im Hinblick auf eine multimodale Visualisierung der Daten beachtet werden sollten. Aus diesem Grund beschäftigt sich dieses Kapitel mit den drei für die Registrierung am häufigsten verwendeten Bildgebungsverfahren Computertomographie (CT), Positronen Emissions Tomographie (PET) und Magnetresonanztomographie (MRT). Weiterhin sollen die Eigenschaften der Datensätze, welche für die Darstellung durch das zu entwickelnde multimodale Visualisierungssystem verwendet wurden, erläutert werden. Die Besonderheiten der verwendeten Modalitäten sollen dabei ebenso, wie die Merkmale der Strukturen, die in den Bilddaten enthalten sind, vorgestellt werden. Ein weiterer wichtiger Punkt ist die Analyse welche Strukturen innerhalb der Bilddaten für die Visualisierung von Interesse sind.

2.1. Anforderungen an die verwendeten Volumendatensätze und den Nutzer

Für die Darstellung der Volumendatensätze mit der Visualisierungskomponente ist es erforderlich, dass die Transformationsparameter für die Ausrichtung der Datensätze bereits mit der vorhandenen Registrierungskomponente berechnet worden sind. Weiterhin müssen diese Translations- und Rotationsparameter sowie die Bilddaten im Bildverarbeitungssystem abrufbar sein. In den Datensätzen müssen Informationen wie Auflösung und Größe der Voxel gespeichert sein, um der Visualisierungskomponente für Anpassungen der verwendeten Funktionen zur Verfügung zu stehen. Es werden keinerlei Einschränkungen bezüglich der Ausprägung dieser Eigenschaften gemacht. Um interessierende Strukturen hervorheben zu können, sollte der Nutzer die Modalität und den abgebildeten Körperbereich kennen und sich über die Eigenschaften der Strukturen, wie z.B. Grauwert und Abgrenzung zum umgebenden Gewebe, bewusst sein. Desweiteren sollte der Nutzer mit der Interpretation von in den Bilddaten auftretenden Artefakten vertraut sein.

Die im Rahmen dieser Arbeit verwendeten Datensätze erfüllen die oben genannten Anforderungen. Insgesamt liegen 19 Volumendatensätze vor, die drei verschiedene Körperbereiche abbilden. Neben 11 zur Verfügung stehenden Aufnahmen des Schädels und sechs Thoraxaufnahmen, sind zwei Ganzkörperdatensätze vorhanden. Sämtliche darzustellende Datensätze sind gruppenweise 9 verschiedenen Patienten zugeordnet.

2.2. Grundlagen der Computertomographie

2.2.1. Bildgebung mittels Computertomographie

¹ Die Bildgebung der Computertomographie basiert auf der 1895 von Wilhelm Conrad Röntgen entdeckten Röntgenstrahlung. Für die Erfassung der Bildinformation muss das Volumen durchstrahlt werden. Somit bedeutet jede CT-Untersuchung eine Strahlenbelastung für den Patienten. In der klinischen Diagnose werden moderne Spiral-CTs verwendet.

Aufbau eines Computertomographen Die wichtigsten Komponenten eines Spiral-CTs sind die sogenannte Gantry und die Patientenliege. Die Gantry setzt sich aus einer Röntgenröhre und den gegenüberliegenden Detektoren zusammen. Die Detektoren bestehen aus einem Szintillationskristall und einer dahinter liegenden Photodiode. Der Szintillationskristall wandelt die Röntgenstrahlung durch Interaktion des γ -Quanten mit einem Elektron in Licht um. Die Photodiode erzeugt entsprechend der Lichtmenge ein elektrisches Signal, welches für die Rekonstruktion der Schicht genutzt wird (vgl. [LOPR97]).

¹Die in diesem Abschnitt vorgestellten Merkmale der Computertomographie (CT) sind folgenden Quellen entnommen: [CTH, GCT, CTA]

Während einer Aufnahme rotiert die Gantry eines Spiral-CTs um den Patienten. Dieser wird dabei mit der Liege kontinuierlich vorangeschoben. Die Gantry beschreibt somit eine Spiralbewegung um das abzubildende Volumen. Abbildung 2.1 zeigt den Aufbau und die Funktion eines Spiral-CTs.

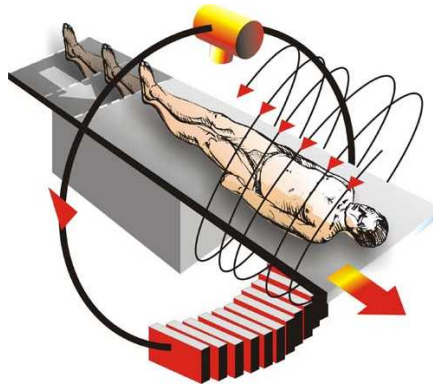


Abbildung 2.1.: Scanning eines Patienten mit einem Spiral-CT. Während die Gantry eine Dauerrotation beschreibt, wird der Untersuchungstisch mit dem Patienten kontinuierlich vorgeschoben. Es resultiert ein spiralförmiger Rohdatensatz, der durch Interpolation zu planaren Schichten umgerechnet wird. Quelle: [Lau99].

Bildrekonstruktion Das von den Detektoren gemessene, analoge Signal muss digitalisiert und in eine Schichtaufnahme umgerechnet werden. Durch die Spiralbewegung kann die Messung der einzelnen Schichten nicht aus allen Winkeln erfolgen. Durch Interpolation werden die fehlenden Messpunkte berechnet und so die Schicht vollständig rekonstruiert. Da die einzelnen Schichten aus den Daten berechnet werden, kann deren Lage so gewählt werden, dass sie sich überlappen, was zu einer Verbesserung der Bildqualität führt. In der klinischen Routine werden meist Überlappungen von 30-50% angewendet (vgl. [CTH]).

Für die Rekonstruktion einer Schicht wird eine gefilterte Rückprojektion verwendet. Hierbei werden die gemessenen Signale zunächst mit einem Hochpassfilter gefaltet, um die Bildschärfe zu erhöhen. Das gefilterte Signal wird anschließend in den Bildraum zurückprojiziert. Die Daten aus allen Winkeln werden zu einem Schichtbild addiert. Die (gefilterte) Rückprojektion wird durch Abbildung 2.2 verdeutlicht. Das Ergebnis der Bildrekonstruktion sind Volumendatensätze, die sich aus Schichtbildern zusammensetzen. In der klinischen Routine werden diese Schichten meist mit einer Auflösung von 512 x 512 Voxeln und einer Pixelgröße von ca. 0,5 x 0,5 mm erzeugt. Die Schichtdicke beträgt je nach Fragestellung ca. 2 mm.

Hounsfield Einheiten Die Werte eines CT-Bildes werden in Hounsfield Einheiten (engl. *Hounsfield Units*, Abk. HU) angegeben. Sie repräsentieren die Schwächungswerte von Geweben relativ zur Röntgendichte von Wasser und werden wie folgt berechnet:

$$CTMesswert = 1000 * (\mu - \mu_{Wasser}) / \mu_{Wasser} \quad (2.1)$$

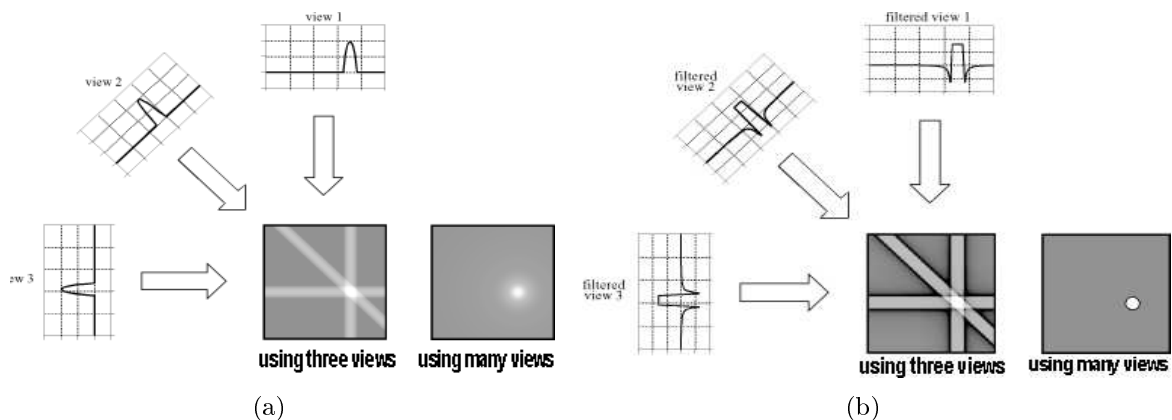


Abbildung 2.2.: Bei der Rekonstruktion einer Schicht aus den gemessenen CT-Daten kommt die gefilterte Rückprojektion, hier im rechten Bild (b) dargestellt, zum Einsatz. Wie unschwer zu erkennen ist, wird durch die Hochpassfilterung, im Gegensatz zur Rückprojektion ohne Filterung (a), die Bildschärfe stark erhöht. Nach der Faltung werden die gemessenen Signale in den Bildraum zurückprojiziert. Dies gleicht einer Verwischung des Bildsignals über die zu rekonstruierende Schicht. Quelle: [FBP].

Wasser hat per Definition 0 HU. Luft liegt bei ca. -1000 HU, während es nach oben keine Grenze gibt. Medizinische Computertomographen arbeiten auf einer Skala von -1024 bis +3071 Hounsfield Einheiten. Somit stehen für eine Visualisierung in Form von Grauwertbildern 4096 Grauwerte zur Verfügung. Eine Übersicht der Hounsfield Einheiten verschiedener Gewebetypen zeigt Abbildung 2.3.

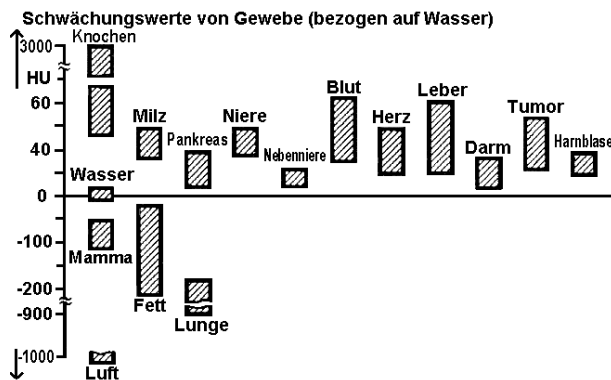


Abbildung 2.3.: Für einige Gewebetypen, die typischerweise mit der CT aufgenommen werden, sind die dazugehörigen Schwächungswerte in Hounsfield Einheiten (*Hounsfield Units*, HU) abgetragen. Aus der Abbildung ist ersichtlich, dass ein Gewebe immer einen Bereich der Hounsfield Skala belegt. Weiterhin wird deutlich, dass sich die Bereiche mehrerer Gewebe überschneiden können. Dies ist z.B. bei tumorösem Gewebe, der Leber und Blut der Fall. Quelle: [LOPR97].

Die CT-Messwerte der Gewebe erstrecken sich jeweils über einen Bereich der Hounsfield Skala. Auch in verschiedenen CT-Aufnahmen haben gleiche Gewebe annähernd die

gleichen Intensitätswerte. Somit sind die Bilddaten gewissermaßen standardisiert. Dies erleichtert die Visualisierung von CT-Datensätzen, da z.B. die Definition und die Wiederverwendung von Transferfunktionen, zur Analyse von Datensätzen mit einer bestimmten Fragestellung, realisierbar werden (vgl. [HST⁺04]). Die Bereiche unterschiedlicher Gewebe können sich überschneiden. Dies kann bei einer späteren Visualisierung zu Problemen führen, da unterschiedliches Gewebe auf gleiche Grauwerte abgebildet wird. Bei der Definition von Transferfunktionen muss daher eine Möglichkeit gefunden werden diese Grauwertbereiche voneinander zu trennen (vgl. Abschnitt 4.2).

Bewertung der Computertomographie Die CT wird in der klinischen Diagnostik sehr häufig verwendet. Dies liegt vor allem daran, dass Computertomographen in fast jeder Klinik verfügbar und die Untersuchungen sehr günstig geworden sind. Die Auflösung einer modernen CT-Aufnahme und der sehr gute Knochenkontrast sind weitere Punkte, die für eine CT sprechen. Die kurzen Aufnahmezeiten ermöglichen Volumendatensätze, die frei von Bewegungsartefakten sind. CT-Daten bilden die Anatomie der aufgenommenen Körperregion ab und ermöglichen so die Suche nach morphologischen Abweichungen, die z.B. Ergebnis krankhafter Veränderungen sind. Ein weiterer wichtiger Einsatzbereich ist die Bestrahlungsplanung in der Onkologie.

Obwohl sich die einzelnen Organe durch Fettgewebe voneinander abgrenzen und somit für den betrachtenden Arzt gut zu unterscheiden sind, macht sich der fehlende Weichteilkontrast innerhalb eines Organs bei einigen klinischen Fragestellungen negativ bemerkbar. Aus diesem Grund werden CT-Aufnahmen im neurologischen Bereich zunehmend durch MRT-Aufnahmen ersetzt. Weiterhin spricht die Strahlenbelastung des Patienten während einer Aufnahme gegen ein CT. Für bestimmte Fragestellungen, wie z.B. die Darstellung der Blutgefäße (Angiographien), müssen Kontrastmittel verabreicht werden, die Unverträglichkeiten beim Patienten auslösen können.

2.2.2. Verwendete Computertomographiedaten

Für die Darstellung mit der multimodalen Visualisierungskomponente, die in dieser Arbeit entwickelt wurde, stehen neun CT-Datensätze zur Verfügung (s. Abb. 2.4). Von diesen sind fünf Schädelaufnahmen, die den in Abschnitt 2.4.2 beschriebenen MRT-Aufnahmen zugeordnet sind. Einer der Schädel Datensätze hat sowohl einen dazugehörigen MRT-, als auch einen PET-Datensatz. Die vier anderen Datensätze gehören zu PET-Daten (vgl. Abschnitt 2.3.2). Davon zeigen drei Datensätze den Thorax des Patienten und einer bildet den gesamten Körper ab.

Eigenschaften der Datensätze Die Unterschiede in den Eigenschaften der zur Verfügung stehenden CT-Datensätze liegen hauptsächlich in den Bereichen Auflösung, Voxelgröße und Anzahl der Bits, mit denen die Messwerte gespeichert werden. Alle vorliegenden Datensätze sind anisotrop, d.h. die Voxelausdehnung in x-/y- und z-Dimension ist unterschiedlich. Die Verarbeitung der Daten innerhalb der zu entwickelnden Visualisierungskomponente muss an die Eigenschaften der Datensätze angepasst werden und erfolgt stets mit 16 Bit Datentypen. Sämtliche zur Verfügung stehenden Datensätze wurden ohne Kontrastmittel aufgenommen. Tabelle 2.1 zeigt die Eigenschaften der CT-Daten

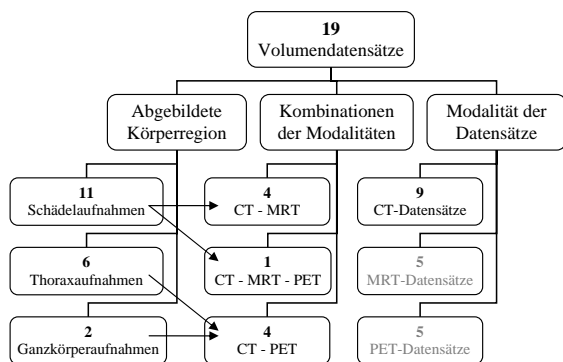


Abbildung 2.4.: Einteilung der verwendeten CT-Datensätze - die für diesen Abschnitt nicht relevanten Datensätze sind grau unterlegt.

beispielhaft anhand von drei Datensätzen.

	CT MRT Head 1	CT MRT PET	CT MRT Head 2
Spannung	120 KV	keine Info	130 KV
Schichtdicke	3,0 mm	4,0 mm	5,0 mm
Auflösung	512 x 512 x 50	512 x 512 x 40	512 x 512 x 36
Pixelgröße	0,59 x 0,59 mm	0,41 x 0,41 mm	0,70 x 0,70 mm
Bits pro Pixel insgesamt/ verwendet	16/16	16/12	16/12
Größe des Datensatzes brutto/ netto	25/25 MB	20/15 MB	18/13,5 MB
Kontrastmittel	ohne	ohne	ohne
Scanner	General Electrics - Genesis Foreign	Siemens - Sensation 16	Siemens - Emotion

Tabelle 2.1.: Eigenschaften von drei verwendeten CT-Schädel Datensätzen.

Abgebildete Körperregionen Die CT-Datensätze bilden sowohl den gesamten Körper bzw. Thorax und Schädel, als auch nur den Schädel ab. Je kleiner die abgebildete Region ist, desto besser kann die Auflösung der Daten werden, ohne die Strahlenbelastung des Patienten zu erhöhen. Aus diesem Grund variiert die Pixelgröße im Bereich von weniger als einem halben Millimeter bei Schädelaufnahmen bis hin zu 2,4 mm bei Ganzkörperdatensätzen.

Die dargestellten Körperregionen unterscheiden sich in ihrem Aufbau sehr stark. So bildet der Schädel eine kompakte Struktur mit wenigen Grenzflächen zwischen Geweben, die von Luft umgeben ist. Im Thorax und Abdomen treten komplexere Formen auf. Die Strukturen sind z.T. viel größer als im Schädel und es gibt viele Grenzflächen. Die Anzahl der zu unterscheidenden Organe ist viel höher als im Schädel. Im Thorax befindet sich Luft in den Lungen und somit auch im Inneren des Volumens. Luft kann auch innerhalb von abdominalen Strukturen auftreten. In jedem Fall erzeugt sie eine sehr kontrastreiche Grenzfläche innerhalb des Datensatzes.

Interessierende Strukturen innerhalb der Daten Aus den CT-Datensätzen sollen vor allem die Knochen dargestellt werden. Sie bilden in der Visualisierung meist den anatomischen Kontext für die Lokalisierung von interessierenden Strukturen aus MRT- oder PET-Daten. Knochen werden aber ggf. auch separat dargestellt z.B. für die Visualisierung von Frakturen. Knochen heben sich im CT aufgrund ihres hohen Intensitätswertes stark vom umgebenden Gewebe ab. Die Größe und vor allem der Durchmesser der Knochenstrukturen ist abhängig von der aufgenommenen Körperregion. Im Schädelbereich sind die Knochen viel dünner als z.B. bei Thoraxaufnahmen. Während Knochen im Thoraxbereich von viel Weichgewebe wie Muskulatur und Haut umgeben sind, werden die Schädelknochen nur von einer dünnen Hautschicht bedeckt und schließen das Weichgewebe (Gehirn) vollständig ein.

Artefakte Als Artefakte in Bilddaten werden alle Merkmale und Strukturen bezeichnet, die zwar im Bild erkennbar aber im abgebildeten Volumen nicht vorhanden sind. In der CT werden drei häufig auftretende Artefakte unterschieden.

Zunächst treten in den vorliegenden Volumendatensätzen vor allem in Ganzkörperaufnahmen Bewegungsartefakte auf. Die Ursache hierfür ist die längere Aufnahmezeit, in der der Patient ruhig liegen und evtl. den Atem anhalten muss. In Thoraxaufnahmen ist auch der Herzschlag für Bewegungsartefakte verantwortlich. Bewegungsartefakte bewirken ein Verwischen der Messwerte. Daraus resultiert die Verschlechterung der Kanteninformation.

Ein weiteres Artefakt entsteht durch die sogenannte Strahlaufhärtung. Hierbei wird die im Röntgenstrahl enthaltene Strahlung geringerer Energie beim Durchgang durch Gewebe absorbiert und der Strahl so aufgehärtet. Dieses Artefakt kann zum einen durch Filterung des Röntgenstrahls (z.B. durch Aluminiumplatten) und zum anderen durch Filterung der berechneten Messwerte ausgeglichen werden. Leider liegt keine Information über verwendete Filter vor.

Das letzte Artefakt entsteht an Materialien, die nicht von der Röntgenstrahlung durchdrungen werden können. Dies sind vor allem Metalle (z.B. Zahnfüllungen). An solchen Materialien entstehen Schatten, die nach der Rekonstruktion der Daten als Streifenartefakte zu erkennen sind. Die hier beschriebenen Artefakte sind dem Vorlesungsskript [CTA] entnommen.

Standardisierung Durch die Zuordnung von Hounsfield Einheiten zu den Geweben ist eine Standardisierung der Messwerte gegeben. Über die Rekonstruktion der Messwerte lag keine Information vor. Daher kann bei einer Untersuchung der Übertragbarkeit von generierten Transferfunktionen auf vergleichbare Datensätze nicht auf diesen Parameter eingegangen werden. Die Parameter Auflösung, Voxelgröße und die Bits pro Messwert können durch geeignete Interpolation vereinheitlicht werden. Daher ist die CT aufgrund ihrer Abbildungseigenschaften von allen in dieser Arbeit genutzten Modalitäten am ehesten für eine solche Standardisierung geeignet.

2.3. Grundlagen der Positronen Emissions Tomographie

Krankhafte Veränderungen zeigen sich oft in einer erhöhten Stoffwechselaktivität. Die PET macht sich dies zu nutze, indem sie körpereigene Radionuklide als Strahlenquelle für die Bildgebung verwendet. Demnach arbeitet PET ebenso wie CT mit Strahlung, allerdings ist die Strahlenquelle hierbei innerhalb des Körpers. Die zu messende Strahlung tritt nur an den Stellen aus dem Körper aus, an denen sich die Radionuklide anreichern. Somit erfolgt zwar eine Darstellung genau dieser Bereiche, morphologische Informationen umgebender Gewebe können aus diesem Grund mit der PET jedoch nicht abgebildet werden. Die PET ist ein wichtiges Mittel z.B. bei der Tumorsuche.

2.3.1. Bildgebung mittels Positronen Emissions Tomographie

² Die Grundlage für die PET bilden Radionuklide die Positronen emittieren. Die Strahlenquelle wird, im Gegensatz zur CT, in den Organismus eingebracht. Hierfür werden Isotope körpereigener Stoffe wie Sauerstoff (^{15}O), Fluor (^{18}F), Kohlenstoff (^{11}C) und Stickstoff (^{13}N) verwendet. Diese Radionuklide müssen in Ringbeschleunigern, sogenannten Zyklotronen hergestellt werden. Die Isotope haben meist eine kurze Halbwertszeit von wenigen Minuten. Die Strahlenbelastung des Patienten für eine Untersuchung wird dadurch gering gehalten. Die kurze Halbwertszeit macht jedoch eine Kopplung eines Positronen Emissions Tomographen an das Vorhandensein eines Zyklotrons notwendig, da die Stoffe nach der Herstellung nicht weit transportiert werden können.

Das von einem Radionuklid emittierte Positron vereinigt sich spätestens nach 5 mm mit einem Elektron. Die beiden Teile werden vernichtet und es entstehen zwei γ -Quanten mit einer Energie von je 511 keV. Aufgrund des Impulserhaltungssatzes bewegen sich die Quanten exakt in einem Winkel von 180° auseinander.

Die Detektorelemente, sogenannte Szintillationszähler, sind bei einem Positronen Emissions Tomographen auf einem Ring angeordnet (s. Abb. 2.5). Zwei sich gegenüberliegende Detektorelemente müssen die entstandene Vernichtungsstrahlung fast gleichzeitig registrieren (innerhalb von maximal 12 ns), um den Positronenstrahler zu lokalisieren. Es werden bis zu 15 solcher Ringe verwendet, um mehrere Schichten gleichzeitig aufnehmen zu können. Die räumliche Verteilung der Positronen Emitter lässt sich über ein Rekonstruktionsverfahren, die sogenannte gefilterte Rückprojektion, berechnen. Die Auflösung zwischen den Schichten ist mit bis zu 1 mm besser als die örtliche Auflösung innerhalb einer Schicht. Diese beträgt zwischen 1-2 mm bei speziellen Hirntomographen und 2-3 mm für Ganzkörpertomographen. Diese Auflösungsbeschränkung ist von mehreren Ursachen abhängig. Die Tatsache, dass die Messung der Positronen nicht am Ort der Emission, sondern am Ort der Vernichtung der Teilchen geschieht und diese sich eine kurze Strecke durch das Gewebe bewegen können, schränkt die örtliche Auflösung am stärksten ein.

²Die in diesem Abschnitt vorgestellten Merkmale der Positronen Emissions Tomographie (PET) sind folgenden Quellen entnommen: [PET, PMD]

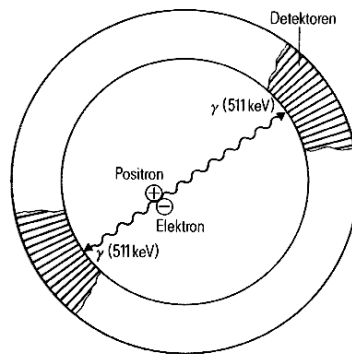


Abbildung 2.5.: Die nach der Vereinigung eines Positrons und Elektrons entstandene Vernichtungsstrahlung bewegt sich in einem Winkel von 180° auseinander. Zwei sich gegenüberliegende Detektoren müssen die Strahlung fast gleichzeitig erfassen, um den Ort der Vereinigung zu registrieren. Quelle: [Poh].

Bewertung der Positronen Emissions Tomographie Aufgrund der schlechten Auflösung und dem fehlenden anatomischen Kontext wird zu jedem PET immer auch ein CT angefertigt, um eine genaue Lokalisierung der erhöhten Stoffwechselaktivität zu ermöglichen. Um krankhafte Veränderungen, die in PET-Aufnahmen zu erkennen sind, in CT-Aufnahmen zu lokalisieren, ist vor der multimodalen Visualisierung eine Registrierung der Datensätze notwendig. Die Registrierung entfällt, wenn die für diesen Zweck entwickelten PET-/ CT-Kombinationssysteme wie der SIEMENS biographTM verwendet werden. Neben der schlechten Auflösung tritt in PET-Daten ein starkes Rauschen auf. Dieses schlechte Signal- zu Rausch-Verhältnis verlangt eine geeignete Vorverarbeitung z.B. durch Medianfilter, um die Qualität der Visualisierung zu verbessern. Ein weiterer Nachteil ist die Strahlenbelastung des Patienten durch die verabreichten Radionuklide. Diese können auch Unverträglichkeiten, wie z.B. Übelkeit oder Geschmacksverlust auslösen. Trotz der vielen Nachteile rechtfertigt die Darstellung der krankhaften Stoffwechselveränderungen die Verwendung der PET vor allem bei der Tumorsuche.

2.3.2. Verwendete Positronen Emissions Tomographie Daten

Insgesamt stehen fünf PET-Datensätze für die multimodale Visualisierung zur Verfügung. Von diesen wurden alle mit den in Abschnitt 2.2.2 vorgestellten CT-Daten registriert. Drei Aufnahmen zeigen den Schädel und Thorax des Patienten und eine den gesamten Körper. Weiterhin liegt ein Datensatz aus „CT MRT PET“ vor, der sowohl mit einem MRT- als auch mit einem CT-Schädel Datensatz visualisiert werden soll (s. Abb. 2.6).

Eigenschaften der Datensätze Die vorliegenden PET-Datensätze haben, aufgrund der großen Voxelausdehnung und der daraus resultierenden schlechten Auflösung, den geringsten Speicherbedarf. Für die gemeinsame Darstellung mit anderen Modalitäten ist eine starke Überabtastung notwendig. Je nach aufgenommener Körperregion sind die Voxel annähernd isotrop oder stark anisotrop. Auch die Anzahl der Bits für die Speicherung der Bildinformation variiert stark (vgl. Tabelle 2.2).

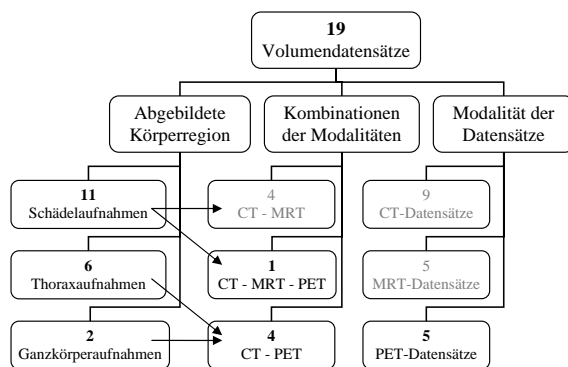


Abbildung 2.6.: Einteilung der verwendeten PET-Datensätze - die für diesen Abschnitt nicht relevanten Datensätze sind grau unterlegt.

	CT MRT PET	CT PET Body
Schichtdicke	4,25 mm	2,42 mm
Auflösung	128 x 128 x 35	128 x 128 x 351
Pixelgröße	4,296875 x 4,296875 mm	5,15 x 5,15 mm
Bits pro Pixel insges./ verwendet	16/16	16/8
Größe des Datensatzes brutto/ netto	1,09/1,09 MB	10,96/5,48 MB
Kontrastmittel	FDG - Fluorodeoxyglucose	k.A.
Scanner	General Electrics - Advance	k.A.

Tabelle 2.2.: Eigenschaften zweier PET-Datensätze. „CT MRT PET“ ist ein Schädel- datensatz und „CT PET Body“ eine Ganzkörperaufnahme.

Abgebildete Körperregionen Wie auch bei den CT-Datensätzen liegen PET-Volumen- daten des gesamten Körpers sowie in der Kombination Thorax und Schädel und reine Schädel- daten vor. Somit variiert die Größe der abgebildeten Regionen stark. Da die Anatomie in PET-Daten nicht vollständig abgebildet wird und die dargestellten Akti- vierungen abhängig von der Schwere der krankhaften Veränderungen sind, können keine Aussagen über die Größe oder Form der dargestellten Strukturen gemacht werden. Nur das Gehirn und in Ganzkörperdatensätzen die Blase zeigen in allen zur Verfügung stehen- den Volumen eine Anreicherung des Kontrastmittels und werden daher hell dargestellt.

Interessierende Strukturen innerhalb der Daten In reinen Schädel- datensätzen zeigt das Gehirn insgesamt eine hohe Stoffwechselaktivität. Um in dieser Region auffällige Be- reiche zu visualisieren, müssen relativ geringe Unterschiede in der Helligkeit erkannt wer- den. Im Gegensatz dazu zeigen große Bereiche von Thorax- und Ganzkörperaufnahmen kaum messbare Aktivierung während sich die interessanten Strukturen durch ihre Hellig- keit stark von der Umgebung abheben. Interessant sind in allen Datensätzen sehr helle Bereiche, die auf eine hohe Stoffwechselaktivität hinweisen. Die Größe der interessanten Strukturen lässt sich nicht allgemein festlegen. Dies liegt daran, dass die dargestellten krankhaften Veränderungen wie z.B. Tumore oft einen Wachstumsprozess repräsentieren

und je nach Fortschreiten der Krankheit verändert sich auch die Größe der Struktur. Weiterhin kann auch die Anzahl interessanter Strukturen in den Datensätzen variieren. So kann ein Tumor in einem Datensatz einzeln oder in Verbindung mit mehreren Metastasen auftreten.

Artefakte Ein wichtiger Punkt ist der hohe Rauschanteil in den Daten. Dieser muss durch eine geeignete Vorverarbeitung reduziert werden. Weiterhin sind neben den interessierenden Strukturen hoher Aktivität in den Datensätzen, je nach aufgenommener Körperregion und dem verwendeten Kontrastmittel, Organe mit einer hohen Grundstoffwechselaktivität und Ausscheidungsorgane sichtbar. Diese müssen durch den Betrachter von den krankhaften Veränderungen unterschieden werden.

Standardisierung Die vorliegenden PET-Datensätze sind nicht standardisiert, da das gemessene Signal nicht in einer Relation zu einer eingesendeten Strahlung steht, wie es beim CT der Fall ist. Die gemessene Strahlung ist zum einen von der Stärke der Anhäufung des Kontrastmittels in bestimmten Organen bzw. krankhaft veränderten Strukturen abhängig. Je nach Fortschreiten oder Art der Erkrankung treten große Unterschiede in der Stoffwechselaktivität und damit in der Intensität in den aufgenommenen Bilddaten auf. Zum anderen wirkt sich der verwendete Positronenemitter und der Zeitraum zwischen Gabe des Kontrastmittels und der Aufnahme auf die Stärke des Signals aus. Diese Unterschiede in den Bilddaten verhindern die Anwendung von Vorgaben (engl. *Presets*) für Transferfunktionen auch bei gleicher Fragestellung.

2.4. Grundlagen der Magnetresonanztomographie

Ziel der MRT ist die Abbildung der Morphologie im Inneren des Patienten. Der Hauptunterschied zwischen der MRT und den zuvor beschriebenen Bildgebungsverfahren ist die Art der Signale, welche für die Berechnung der Bildinformation genutzt werden. Anders als z.B. bei der CT wird hier keine Röntgenbestrahlung des Körpers durchgeführt. Die MRT misst stattdessen die Menge an Wasserstoff-Protonen im Gewebe anhand derer kernmagnetischen Eigenschaften und der daraus resultierenden Magnetisierung.

2.4.1. Bildgebung mittels Magnetresonanztomographie

³ Für die Aufnahme eines Datensatzes wird der Patient in ein Dauermagnetfeld gebracht, an dem sich der größte Teil der im Gewebe enthaltenen Protonen parallel ausrichtet. Durch drei sogenannte Gradienten wird die Stärke des Magnetfeldes lokal verändert, so dass eine Kodierung der Positionen innerhalb des Magnetfeldes möglich wird. Hierzu wird zunächst eine Schicht aus dem Volumen durch einen Gradienten ausgewählt. Eine Besonderheit der MRT ist, wie in Abbildung 2.7 gezeigt, dass die Lage dieser Schnittebenen frei gewählt werden kann.

³Die in diesem Abschnitt vorgestellten Merkmale der Magnetresonanztomographie (MRT) sind folgenden Quellen entnommen: [HK03, MRIa, GMR]

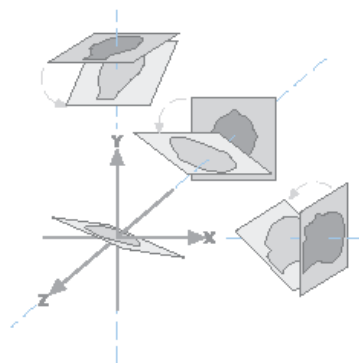


Abbildung 2.7.: Da die Wahl der Schichten in der MRT über sogenannte Gradienten erfolgt, welche ein statisches Magnetfeld beeinflussen, können beliebige Schnittebenen durch das Volumen gewählt werden. Quelle: [HK03].

Über die Anregung durch einen elektromagnetischen Impuls werden die Protonen der gewählten Schicht anschließend um einen bestimmten Winkel (Flipwinkel) aus ihrer Ausrichtung gekippt wodurch eine Quermagnetisierung entsteht. Während der sogenannten Längsrelaxation richten sich die Protonen je nach Gewebe unterschiedlich schnell wieder an dem Magnetfeld aus und die Quermagnetisierung zerfällt. Dabei geben sie die zuvor aufgenommene Energie in Form von elektromagnetischen Wellen ab, welche für die Berechnung des Signals gemessen werden. Die Position des Signals innerhalb der Schicht wird dabei durch die anderen beiden Gradienten kodiert. Durch die Wahl des Messzeitpunktes, kann das Signal anhand der Relaxationszeiten (T_1 = Zeit bis zur Ausrichtung und T_2 = Zeit bis zum Zerfall der Quermagnetisierung) gewichtet werden, was zu unterschiedlichen Messwerten für dasselbe Gewebe führen kann. Die Gehirnflüssigkeit, das sogenannte Liquor, sendet in der T_1 Wichtung ein schwaches Signal, während sie bei einer T_2 Wichtung viel Signal abgibt. In einem T_1 gewichteten Bild erscheint der Liquor daher dunkel und in einer T_2 Wichtung hell. Abbildung 2.8 zeigt eine MRT-Aufnahme eines Gehirns in beiden Wichtungen (vgl. [HK03]).

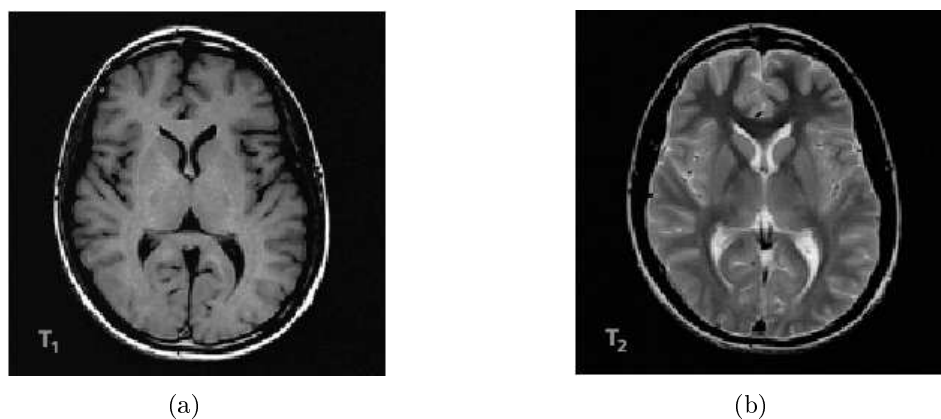


Abbildung 2.8.: Aus T_1 und T_2 Wichtung resultierende MRT-Bilder des menschlichen Gehirns. Auffällig sind die Unterschiede in der Darstellung der Gehirnflüssigkeit (T_1 - dunkel (a); T_2 - hell (b)). Quelle: [HK03].

Bewertung der Magnetresonanztomographie Die MRT kommt völlig ohne schädliche radioaktive Strahlung aus. Somit ist die Belastung des Patienten, bis auf die Lautstärkeentwicklung während der Aufnahme, sehr gering. Aus diesem Grund können Aufnahmen beliebig oft wiederholt werden, was vor allem bei Verlaufskontrollen vorteilhaft ist. Für die Diagnose ist vor allem der sehr gute Weichteilkontrast wichtig. Dieser ermöglicht es, unterschiedliche Gewebetypen zu klassifizieren, Veränderungen zu lokalisieren und so z.B. die Tumorsuche zu erleichtern. Auch bei der MRT müssen für bestimmte Fragestellungen Kontrastmittel verwendet werden. Eingesetzt werden häufig Gadolinium-Präparate, welche die Relaxationszeiten stark reduzieren und so in einer T1 Wichtung hell erscheinen. Es können Unverträglichkeiten gegen die Kontrastmittel auftreten, diese sind allerdings weniger häufig als bei Röntgenkontrastmitteln (vgl. [MRIb] und [MED]). Neben der Abbildung der Morphologie ermöglicht die MRT auch einen Einblick in funktionelle Abläufe z.B. im Gehirn. Funktionelle Daten lagen für diese Arbeit allerdings nicht vor.

Im Gegensatz zum guten Weichteilkontrast fällt negativ auf, dass Knochen im MRT nur als dunkle Bereiche erkennbar sind. Eine Diagnose der Knochenstruktur ist damit nicht möglich. Aus diesem Grund sind Kombinationen von MRT- und CT-Aufnahmen sinnvoll, in denen die Knochenstrukturen aus dem CT und der Weichteilkontrast aus dem MRT dargestellt werden. Auch die langen Aufnahmezeiten von mehreren Minuten für hochaufgelöste anatomische Datensätze sind ein Nachteil dieses Bildgebungsverfahrens. Während den Aufnahmen können Bewegungen Artefakte verursachen. Weiterhin sind Magnetresonanzaufnahmen noch immer sehr teuer und nicht an allen Kliniken verfügbar.

2.4.2. Verwendete Magnetresonanztomographiedaten

Bei der MRT müssen die für die Bildgebung verwendeten Wichtungen unterschieden werden. Die vorliegenden Magnetresonanzdaten wurden größtenteils mit einer T1 Wichtung aufgenommen (vier Datensätze). Weiterhin steht ein Datensatz mit einer T2 Wichtung zur Verfügung. Die MRT-Daten bilden ausschließlich den Kopf ab (s. Abb. 2.9). Neben der monomodalen Visualisierung werden sie mit fünf CT-Datensätzen (vgl. Abschnitt 2.2.2) und einem PET-Datensatz (vgl. Abschnitt 2.3.2) registriert und visualisiert. Im Folgenden werden beispielhaft die Eigenschaften von drei MRT-Datensätzen genauer dargestellt.

Eigenschaften der Datensätze Die vorliegenden Datensätze unterscheiden sich ganz erheblich in ihren Datenmengen. Dies liegt an den Unterschieden in Auflösung und Anzahl der Bits mit denen ein Messwert gespeichert wird. Der Speicherbedarf des hochaufgelösten Datensatzes aus der ersten Spalte in Tabelle 2.3 ist aus diesem Grund mehr als 10 mal so hoch wie der des Datensatzes aus Spalte zwei. Die MRT-Datensätze wurden ebenso wie die CT-Daten mit einer anisotropen Voxelgröße aufgenommen.

Auch in den Einstellungen am Gerät unterscheiden sich die vorliegenden Datensätze stark. Wie aus den drei Beispielen in Tabelle 2.3 zu erkennen ist, ist keine Standardisierung der Aufnahmen erfolgt. Vor allem die Unterschiede in Repititionszeit (die Zeitspanne zwischen zwei Anregungen) und Echozeit (die Zeitspanne vom messbaren Signal bis zu seinem Echo, dass durch einen zweiten 180° Impuls erzeugt wurde) und die daraus resul-

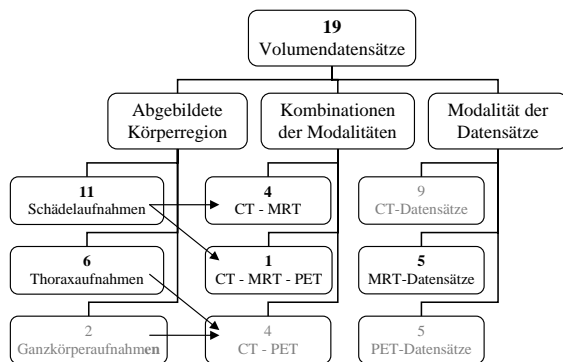


Abbildung 2.9.: Einteilung der verwendeten MRT-Datensätze - die für diesen Abschnitt nicht relevanten Datensätze sind grau unterlegt.

tierende Wichtung müssen bei der Definition von Transferfunktionen beachtet werden. Zwischen einer T1 und einer T2 Wichtung wird der Unterschied in stark wasserhaltigen Regionen am deutlichsten (T1 - Wasser dunkel, T2 - Wasser hell). Doch selbst zwischen den beiden T1 gewichteten Datensätzen „CT MRT Head 1“ und „CT MRT PET“ unterscheiden sich die Wichtungen, woraus sich Unterschiede in den abgebildeten Strukturen ergeben. Die Repititions- und Echozeit bieten die Möglichkeit bestimmte Gewebe gezielt stärker zu wichten. So wurde bei „CT MRT Head 1“ die weiße Substanz hervorgehoben, während in „CT MRT PET“ die graue Substanz das stärkste Signal gibt. Die Tabelle 2.3 zeigt anhand von drei Beispielen die Unterschiede in den Eigenschaften der vorliegenden MRT-Datensätze.

	CT MRT Head 1	CT MRT PET	CT MRT Head 2
Wichtung	T1	T1	T2
Magnetfeld	1,5 Tesla	1,5 Tesla	1,5 Tesla
Echozeit	9,0 ms	14,0 ms	90,0 ms
Repititionszeit	440,0 ms	735,0 ms	3800,0 ms
Flipwinkel	90 Grad	70 Grad	180 Grad
Anzahl Echos	1	1	2
Schichtdicke	3,0 mm	5,0 mm	5,0 mm
Schichtabstand	3,0 mm	1,0 mm	0,5 mm
Auflösung	512 x 512 x 56	256 x 256 x 21	256 x 256 x 22
Pixelgröße	0,59 x 0,59 mm	0,78 x 0,78 mm	0,97 x 0,97 mm
Bits pro Pixel insgesamt/ verwendet	16/16	16/12	16/12
Größe des Datensatzes brutto/ netto	28/28 MB	2,625/ 1,96 MB	2,75/ 2,06 MB
Kontrastmittel	ohne	ohne	ohne
Scanner	General Electrics - Genesis Signa	Siemens - Vision	Siemens - Vision

Tabelle 2.3.: Eigenschaften von drei verwendeten MRT-Schädeldatensätzen

Abgebildete Körperregionen Alle verwendeten MRT-Daten zeigen ausschließlich den Kopf verschiedener Patienten. Thoraxaufnahmen sind nur beschränkt möglich, da für gut aufgelöste Datensätze lange Aufnahmezeiten von mehreren Minuten notwendig sind. Dabei würden Bewegungsartefakte durch Atmung und Herzschlag entstehen. Diese können nur durch eine aufwendige Anpassung der Aufnahmezeitpunkte an den Herzschlag und Atemrhythmus reduziert werden. Anders als in CT-Daten sind die Strukturen in MRT-Schädelaufnahmen weniger kompakt, da MRT in der Lage ist Unterschiede in den Weichgeweben zu erfassen und so z.B. graue von weisser Substanz und das Gehirn vom Liquorraum zu trennen.

Interessierende Strukturen innerhalb der Daten Für die Visualisierung sind vor allem krankhaft veränderte Strukturen im Gehirn wie z.B. vergrößerte Liquorräume oder Tumore von Interesse. Die Größe der Strukturen variiert je nach Ausprägung der Krankheit. Auch die optischen Eigenschaften lassen sich nicht allgemein festlegen, da sie von der Wichtung des Datensatzes abhängig sind.

Artefakte Die in MRT-Aufnahmen auftretenden Artefakte können angelehnt an [Kau] in drei Gruppen eingeteilt werden. Die erste Gruppe von Artefakten entsteht bei der Datenmessung und basiert auf den Eigenschaften des Messgerätes. So erzeugt eine Inhomogenität des Dauermagnetfeldes Störungen innerhalb des Bildes. Diese Störungen können sowohl die Form von Strukturen im Bild, als auch die abgebildeten Intensitäten verändern. Eine weitere Ursache für solche Artefakte können Fehler bei der Gradientenschaltung sein. Eine Inhomogenität des elektromagnetischen Impulses bewirkt Veränderungen der Intensität innerhalb eines kleinen Bereiches im Bild. Die Hauptursache für dieses Artefakt sind nichtferromagnetische Materialien (z.B. Zahnersatz), welche den elektromagnetischen Impuls daran hindern, in das dahinterliegende Gewebe einzudringen.

Die zweite Gruppe von Artefakten entsteht ebenso bei der Datenerfassung, resultiert allerdings aus den Eigenschaften des abzubildenden Volumens. Der am häufigsten in der MR-Bildgebung auftretende Bildfehler ist das Bewegungsartefakt. Durch die langen Aufnahmezeiten bei hochaufgelösten anatomischen Datensätzen passiert es häufig, dass der Patient sich zwischen einer Anregung und dem Auslesen des Signals bewegt. In Thoraxaufnahmen basieren diese Artefakte häufig auf Bewegungen des Herzens oder des gesamten Brustkorbs durch die Atmung. Artefakte dieser Art können durch die Abstimmung der Messzeitpunkte auf die Herztätigkeit und Atmung (engl. *triggering*) unterdrückt werden. Bewegungsartefakte stellen sich als unscharfe Bildbereiche dar. Flussartefakte basieren auf der Bewegung von Flüssigkeiten (z.B. Blut) im Volumen. Sie treten auf, da der angeregte Teil der Flüssigkeit die selektierte Schicht zum Zeitpunkt der Datenmessung bereits wieder verlassen hat. Das Ergebnis sind Blutgefäße, die obwohl das Blut sehr viel Wasser enthält, kein Signal abgeben und sich daher im Bild schwarz darstellen. Das letzte Artefakt dieser Gruppe wird als chemische Verschiebung (engl. *Chemical Shift*) bezeichnet. Es resultiert aus den unterschiedlichen elektromagnetischen Eigenschaften von fett- und wasserhaltigem Gewebe und einer daraus entstehenden falschen Positionskodierung während der Aufnahme. Das Signal eines wasser- und fetthaltigen Voxels wird dabei nicht dem eigentlichen Voxel, sondern einem seiner Nachbarn zugeordnet.

Die dritte Gruppe von Artefakten basiert sowohl auf den Eigenschaften des Magnetresonanztomographen als auch auf der Bildrekonstruktion. Ein möglicher Bildfehler stellt sich als Einfaltung von Bereichen des Volumens, die nicht innerhalb des Bildes liegen, dar. Diese eingefalteten Bildbereiche wurden bei der Messung mit angeregt und anschließend wird ihr Signal gemessen. Dieses Signal wird dann bei der Rekonstruktion über die innerhalb des Bildes liegenden Signale auf der gegenüberliegenden Seite addiert. Das letzte Artefakt tritt an starken Kanten in schlecht aufgelösten Bildern auf. Dort bilden sich parallele Linien, die Wiederholungen des Kantensignals darstellen. Dieses Phänomen wird als Gibbs Ringing bezeichnet.

Standardisierung Aufgrund der großen Varianz der Einstellungen für die Datenaufnahme und der daraus resultierenden Unterschiede im Bildmaterial ist eine Standardisierung im Hinblick auf einheitliche Transferfunktionen für die vorliegenden MRT-Datensätze nicht möglich. Die Wahl der genauen Repititions- und Echozeit folgt zwar einigen standardisierten Vorgaben, liegt aber zum Teil auch im Ermessen des Radiologen. Aus diesem Grund ist es erforderlich, für jeden MRT-Datensatz eine eigene Transferfunktion für die Abbildung auf Farbe und Transparenz zu spezifizieren.

2.5. Vom Messsignal zum Volumendatensatz

2.5.1. Bildrekonstruktion

Die drei in den vorangegangenen Abschnitten vorgestellten Modalitäten sind Schnittbildverfahren, bei denen die Aufnahme eines 3D-Volumens in Form von mehreren parallelen Schichten erfolgt. Bei der Rekonstruktion von Schichtbildern aus den gemessenen Daten werden die kontinuierlichen Signale in eine diskrete Ortsbereichsfunktion umgewandelt. Die dafür notwendige Diskretisierung (Digitalisierung) ist ein zweistufiger Prozess (vgl. [LOPR97]).

1. Diskretisierung des Argumentbereiches (Ortsbereich) – **Abtastung**
2. Diskretisierung des Wertebereiches – **Quantisierung**

Die kontinuierlichen Signale werden meist im Frequenzraum betrachtet. Dort sind sie als Summe von Sinusfunktionen repräsentiert. Um das Signal in seine Bestandteile zu zerlegen, wird die diskrete Fourier-Transformation angewendet. Hierbei wird das Signal $f(x)$ nur an N diskreten Stellen abgetastet (Sampling).⁴

$$F(u) = \sum_{0 \leq x \leq N-1} f(x) [\cos(2\pi ux/N) - i \sin(2\pi ux/N)], \quad (2.2)$$

Um eine ausreichende Abtastung zu gewähren und eine Artefaktbildung zu verhindern, muss die Abtastrate r nach dem Abtasttheorem mindestens doppelt so hoch sein, wie die höchste Frequenz f_{max} (Grenzfrequenz), aus der sich das Signal zusammensetzt. Eine solche Abtastrate nennt sich Nyquist-Rate.

$$r = 1/N = 2 \cdot f_{max} \quad (2.3)$$

Für eine vollständige Rekonstruktion des Bildes aus dem abgetasteten Signal, müsste das Signal im Ortsraum mit einer *sinc*-Funktion (Abb. 2.10(a)) gefaltet werden.

$$\text{sinc}(x) = \sin(\pi x) / (\pi x) \quad (2.4)$$

Das Äquivalent der *sinc*-Funktion im Frequenzraum, ist ein Pulsfilter (Abb. 2.10(b)). Dieser entspricht einer Bandpassfilterung der Frequenzen, aus denen sich das Signal zusammensetzt.

$$S(u) = \begin{cases} 1, & \text{wenn } -k \leq u \leq k; \\ 0, & \text{sonst.} \end{cases} \quad (2.5)$$

Da die *sinc*-Funktion im Ortsraum unendlich weit ist, müssen für die Rekonstruktion eines Pixelwertes die Messwerte aller Abtastpunkte mit einbezogen werden. Diese Eigenschaft der Funktion wird „*infinite support*“ genannt. Da dies aber enorme Rechenzeit benötigt, werden im Frequenzraum häufig Dreieck- oder Gaußfilter für die Rekonstruktion des Signals im Frequenzraum verwendet (vgl. [FvDFH90, RS01]). Diese schränken

⁴Diese und die folgenden Gleichungen sind zum besseren Verständnis nur für den eindimensionalen Fall dargestellt.

den Bereich der für die Rekonstruktion verwendeten Frequenzen ein. Ein Dreieckfilter im Frequenzraum und sein Äquivalent im Ortsraum sind in Abbildung 2.10(d) und 2.10(c) dargestellt.

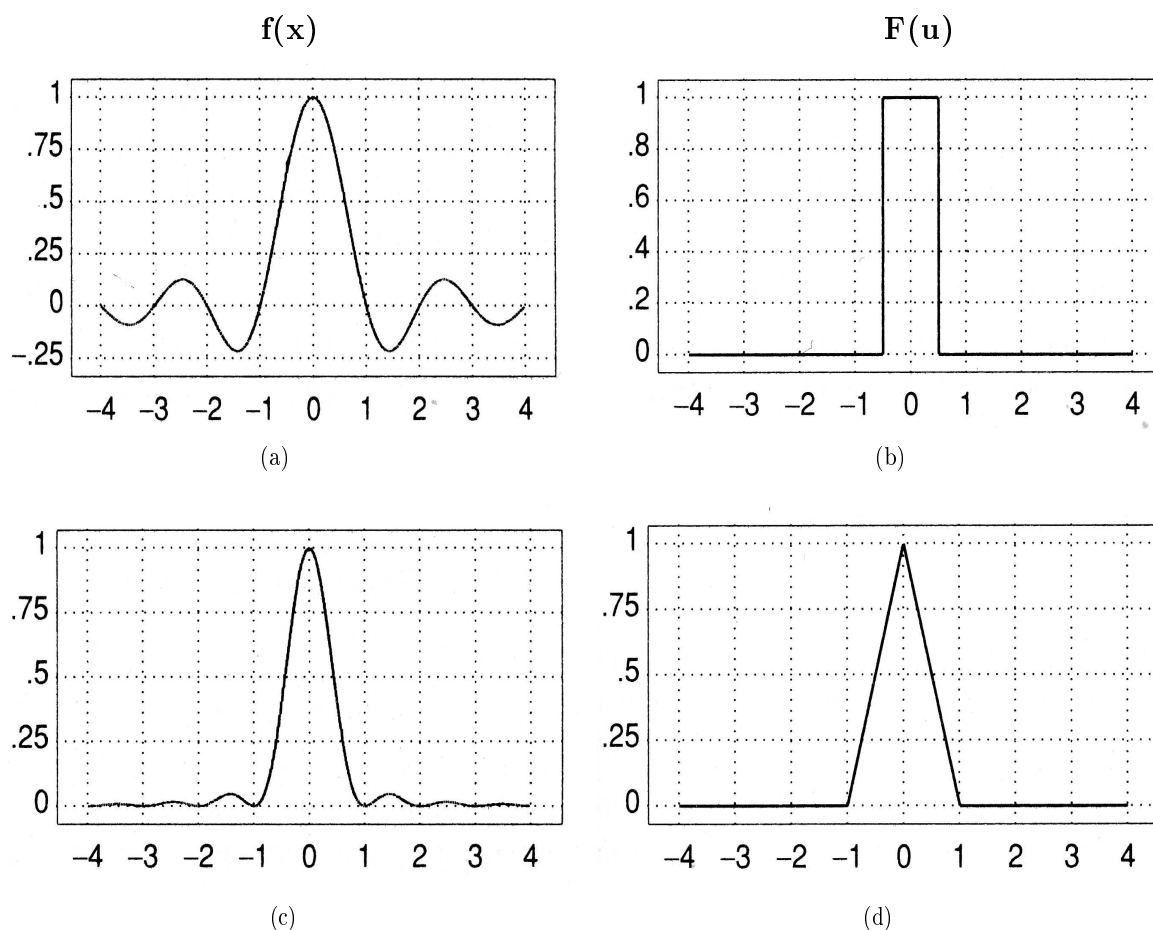


Abbildung 2.10.: Für die Rekonstruktion eines Bildes aus einem gemessenen Signal müsste das Signal im Ortsraum $f(x)$ mit einer *sinc*-Funktion (a) gefaltet werden. Dies entspricht einer Faltung des Signals im Frequenzraum $F(u)$ mit einem Pulsfilter (b). Um die Berechnung zu vereinfachen, werden in der Praxis häufig Dreieckfilter (d) verwendet. Das Äquivalent im Ortsraum ist in (c) gezeigt. Quelle: [FvDFH90].

Bei der Beschränkung der Rekonstruktion auf bestimmte Frequenzen des Fourier-Spektrums eines Signals, entstehen Artefakte wie Aliasing oder Moiré-Effekte. Die Parameter des Rekonstruktionsfilters können vom Radiologen gewählt werden und haben Einfluss auf die Qualität der Rekonstruktion. So ist es möglich die Daten „weich“ zu rekonstruieren, d.h. das Rauschen in den Daten wird geglättet. Mit dem Rauschen geht aber auch Kanteninformation verloren. Eine „harte“ Rekonstruktion hingegen behält die Kanteninformation bei und feinere Strukturen sind im resultierenden Bild erkennbar. Dies geht allerdings mit einem erhöhten Rauschanteil einher. Auch eine mittlere Rekonstruktion ist möglich. Diana Stölzel untersucht in [Stö04] sehr ausführlich die Auswirkung der Rekonstruktion auf die Qualität von Transferfunktionen, welche auf Kanteninformationen basieren.

Aufgrund von Rundung, Mittelwert- oder Schwellwertbildung entstehen auch bei der Quantisierung Fehler. Die reellen Zahlenwerte der Abtastung werden auf die Menge der vom Rechner speicherbaren Werte $W \subset \mathbb{N}$ abgebildet.

$$f(x) \in \mathbb{R} \rightarrow g(x) \in W \quad (2.6)$$

Die Kodierung der quantisierten Werte erfolgt binär, meist mit 8 oder 16 Bits. Dies muss bei der Visualisierung medizinischer Daten beachtet werden, da es nicht möglich ist, eine Transferfunktion, die für 8 Bit Daten definiert wurde, ohne weiteres auf 16 Bit Daten anzuwenden und umgekehrt.

Das Ergebnis von Abtastung und Quantisierung ist eine Funktion, die jedes Pixel eines rechteckigen, endlichen Gitters der Dimension $(M \times N)$ auf einen endlichen Wertebereich abbildet. In medizinischen Bildern ist dieser Wertebereich meist die Grauwertmenge \underline{G} :

$$f : \underline{M} \times \underline{N} \rightarrow \underline{G} \quad (2.7)$$

Die Elemente der Bereiche sind natürliche Zahlen, so dass für die Mengen \underline{M} , \underline{N} , \underline{G} und ihre Elemente m , n , g gilt: $m \in \underline{M} = \{0, 1, \dots, M - 1\}$, $n \in \underline{N} = \{0, 1, \dots, N - 1\}$ und $g \in \underline{G} = \{0, 1, \dots, G - 1\}$. Mit Hilfe dieser Definition aus [LOPR97] kann auf die Pixelwerte eines Bildes wie folgt zugegriffen werden:

$$f(m, n) = g \quad (2.8)$$

Der resultierende Wert entspricht einer Mittelung der reellen Werte in einer Umgebung um den dazugehörigen Messpunkt. Der nächste Abschnitt geht näher auf diese Besonderheit ein.

2.5.2. Partialvolumeneffekt

Der Partialvolumeneffekt ist in allen Volumendatensätzen vorhanden. Er tritt auf, wenn innerhalb der Umgebung, aus welcher der Wert für einen Voxel rekonstruiert wird, eine Grenze zwischen mehreren Geweben mit unterschiedlichen Eigenschaften verläuft. Der Wert des Voxels setzt sich in diesem Fall aus den Messwerten der Gewebe zusammen. Abbildung 2.11(a) zeigt ein Volumen im Querschnitt, in dem die beiden Gewebe A und B durch eine eindeutige Grenze voneinander getrennt sind. Somit entspricht jeder der drei Punkte P_1 , P_2 und P_3 genau einem Gewebe. Bei der Rekonstruktion des Bildes wird nun die Umgebung um jeden Abtastpunkt mit einbezogen. Dabei ergibt sich für Punkte, die nah an einer Grenze liegen, eine Mittelung der Messwerte und somit ein Mischgewebe A&B. Die anschließende Abtastung für die Darstellung des Bildes ist in Abbildung 2.11(b) dargestellt. Das Gitter soll eine mögliche Verteilung und Ausdehnung der Voxel im Bild darstellen. Der Punkt P_3 liegt innerhalb des Bereiches, in dem die Messwerte gemischt wurden. Würde der Wert dieses Punktes für das gesamte Voxel gelten, dann wäre im Bild an dieser Stelle die Grenze zwischen den Geweben unscharf [LFB98].

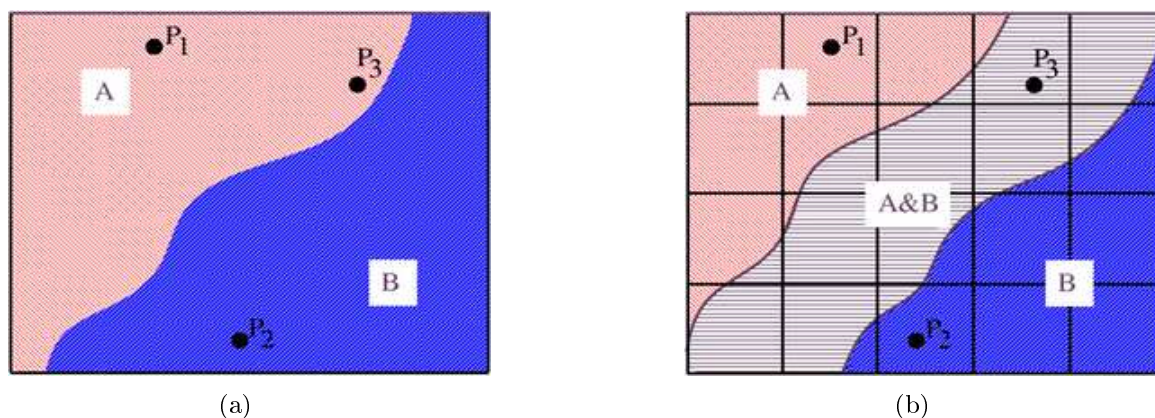


Abbildung 2.11.: Bei der Abbildung zweier Gewebe A und B (s. Abb. (a)) tritt durch die Diskretisierung der Signale an der Grenze zwischen den Geweben der Partialvolumeneffekt auf. Da bei der Berechnung des Wertes eines Abtastpunktes auch die Umgebung mit eingeht, vermischen sich die Gewebe in der Nähe der Grenze (Abb. (b)). Im rekonstruierten Bild würde ein Voxel, dessen Wert sich aus Punkt P_3 ergibt, nicht eindeutig zu einem Gewebe zugeordnet werden können und den Mischwert A&B erhalten. Quelle: [LFB98].

Die Auswirkungen des Partialvolumeneffekts müssen bei der Visualisierung mit Transferfunktionen beachtet werden. So treten zum einen Bereiche im Bild auf, die sich nicht eindeutig einem Gewebe zuordnen lassen. Zum anderen haben unscharfe Kanten Einfluss auf die Berechnung der Gradienten, welche neben anderen Eigenschaften des Bildes als Parameter für Transferfunktionen genutzt werden.

Die Ausprägung des Partialvolumeneffekts variiert mit der Größe der Volumenelemente. Daher sind Partialvolumenartefakte verstärkt in den Thorax- und Ganzkörperaufnahmen sowie PET-Daten mit einer geringen Ortsauflösung zu finden. Die Verwischung der exakten Kanteninformation durch den Partialvolumeneffekt wirkt sich in den verwendeten Aufnahmen vor allem zwischen den einzelnen Schichten des Volumens aus. Aus den Tabellen 2.1, 2.2 und 2.3 wird ersichtlich, dass die verwendeten Datensätze Schichtdicken von mehreren Millimetern aufweisen. Eine Auswirkung des Partialvolumeneffekts ist, dass der Rand von rundlichen Strukturen treppenartig dargestellt wird. Weiterhin werden z.B. in PET-Aufnahmen, durch die geringe Auflösung auch innerhalb der Schichten sowie die Ungenauigkeit der Messung, die Grenzen der interessierenden Strukturen verwischt, wodurch diese in der Visualisierung nicht genau dargestellt werden können.

2.6. Registrierung von medizinischen Datensätzen

Für die multimodale Visualisierung spielt die Registrierung von medizinischen Datensätzen eine große Rolle. Durch die Vor- und Nachteile der bildgebenden Verfahren bezüglich ihrer Abbildungseigenschaften kann es sinnvoll sein, Daten aus verschiedenen Quellen gleichzeitig darzustellen. Durch die multimodale Visualisierung wird es möglich, dass sich die Daten ergänzen und dem Betrachter die Informationsaufnahme erleichtert wird

bzw. neue Informationen erschlossen werden. Insbesondere die Kombination von anatomischen und funktionellen Daten ist sinnvoll, um krankhafte Veränderungen genau zu lokalisieren. Die Visualisierung registrierter Daten kann somit theoretisch in fast allen klinischen Bereichen Anwendung finden. Neben der Diagnose können registrierte Daten auch für die Operations-/ Bestrahlungsplanung, intraoperative Überwachung sowie die Verlaufskontrolle von Therapien und zur Auswertung der Behandlungsergebnisse verwendet werden. Für letztere bietet sich eine Multivolumen-Visualisierung der gleichen Modalität an, wobei die Aufnahmen zu unterschiedlichen Zeitpunkten angefertigt wurden. Die Entwicklung von robusten, leicht zu parametrisierenden und schnellen Registrierungsverfahren sind die Schwerpunkte aktueller Forschung auf diesem Gebiet. Das Ziel ist es, die Registrierung mit in die klinische Diagnostik zu integrieren. Bisher wird die Registrierung von Datensätzen nur vorgenommen, wenn ein großer Nutzen daraus gezogen werden kann.

Dieser Abschnitt soll nur einen kurzen Überblick über mögliche Formen der Registrierung geben. Alle für die in dieser Arbeit vorgestellte multimodale Visualisierung verwendeten Daten wurden bereits im Vorfeld automatisch, basierend auf der von Maes et al. in [MCV+97] vorgeschlagenen *Mutual Information Registration* (MIR), registriert. Diese optimiert die zur Registrierung verwendete Transformation anhand der *Mutual Information* (MI) $I(A, B)$, einer statistischen Größe über zwei Zufallsvariablen (Datensätze) A und B . Die Registrierung setzt sich aus zwei Teilschritten zusammen. Zunächst muss die Transformation bestimmt werden, welche die beiden Datensätze in das gleiche Koordinatensystem überführt. Anschließend müssen die Auflösungen der Datensätze angeglichen und die berechnete Transformation angewendet werden. Dies wird auch als Ausrichtung der Datensätze (*Alignement*) bezeichnet (vgl. [FPT04]). Für die Bestimmung der Transformation wird einer der beiden Datensätze, das sogenannte *Fixed Image*, nicht verändert und der andere Datensatz (*Floating Image*) wird je nach den verwendeten Transformationen modifiziert. Somit muss die berechnete Transformation für die spätere Visualisierung auch nur auf einen der beiden Datensätze (welcher während der Registrierung das *Floating Image* war, bzw. die Inverse-Transformation auf das *Fixed Image*) angewendet werden. Die Transformationen, welche für die Registrierung in dieser Arbeit berechnet werden, zählen zu den rigiden Transformationen und setzen sich aus Rotationen und Translationen zusammen. Die zu maximierende MI beschreibt die statistische Abhängigkeit der beiden Zufallsvariablen bzw. die Redundanz der Information der Datenwerte zusammengehöriger Voxel aus den beiden Datensätzen. Die MI wird dann maximal, wenn die beiden Datensätze exakt zueinander ausgerichtet sind. Bei der MIR werden keinerlei Annahmen über die Beziehung zwischen den Datenwerten (wie z.B. der Intensitätsunterschied) gemacht. MIR ist auch nicht auf bestimmte Bildinhalte oder verwendete Modalitäten beschränkt. Damit eignet sich die MIR sehr gut für die Registrierung von medizinischen Volumendatensätzen, wie sie für diese Arbeit benötigt wurde (vgl. [MCV+97]).

Die Klassifikation von Registrierungsverfahren erfolgt nach [MV98] anhand von 9 Punkten:

- | | | |
|--------------------------------------|--------------------------|---------------------------|
| 1. Dimension der registrierten Daten | 2. Registrierungsbasis | 3. Art der Transformation |
| 4. Domäne der Transformation | 5. Benötigte Interaktion | 6. Optimierungsstrategien |
| 7. Beteiligte Modalitäten | 8. Beteiligte Subjekte | 9. Beteiligte Objekte |

Anhand der Dimensionalität (1.) werden zwei- und dreidimensionale Daten sowie sogenannte 4D-Datensätze, bei denen ein 3D-Volumen zu unterschiedlichen Zeitpunkten aufgenommen wird, unterschieden. Für die in dieser Arbeit erstellten Visualisierungen wurden dreidimensionale Datensätze miteinander registriert.

Die Registrierungsbasis (2.) gibt an, welche Informationen für die Registrierung verwendet werden. Unterschieden wird in extrinsische Information, die künstlich von außen an das Objekt angefügt wird und intrinsische Information, die sich aus den aufgenommenen Bildern ergibt. Beispiele für extrinsische Informationen sind Stereotaktische Rahmen und Markierungen auf der Hautoberfläche. Als intrinsische Information können z.B. segmentierte Oberflächen, oder anatomische Landmarken genutzt werden. Die für die Registrierung in dieser Arbeit genutzte MIR verarbeitet die Grauwerte der Bilder und nutzt somit intrinsische Informationen.

Um die Daten in Übereinstimmung zu bringen sind Transformationen (3.) der Voxel bzw. der segmentierten Oberflächen notwendig. Eine Auswahl von Transformationen ist in Abbildung 2.12 dargestellt. Die Registrierung, welche dieser Arbeit zugrunde liegt, basiert auf rigiden Transformationen.

Transformationen können lokal auf einzelne Voxel bzw. Oberflächenpunkte angewendet werden oder global den gesamten Datensatz verändern (4.). Für eine Grobregistrierung werden meist globale Transformationen angewendet, um anschließend durch lokale Transformationen feinere Strukturen übereinander zu bringen. Die Registrierungen wurden für diese Arbeit global ausgeführt.

Weiterhin wird nach der Art und Menge der nötigen Interaktion (5.) unterschieden. Automatische Verfahren registrieren Datensätze ohne menschliche Unterstützung, während halb-automatische Registrierungsansätze meist eine initiale Registrierung durch den Benutzer erwarten. Weiterhin besteht die Möglichkeit, dass der Nutzer Ergebnisse des Algorithmus korrigiert. Bei der interaktiven Registrierung gibt der Algorithmus eine initiale Transformation vor und der Nutzer registriert die Datensätze selbst. Die verwendete MIR zählt zu den automatischen Verfahren.

Bei automatischen oder halb-automatischen Verfahren finden verschiedene Arten von Optimierungsstrategien Anwendung (6.). Die Parameter für eine Transformation können zum einen für feste Punkte in den Datensätzen über eine Optimierung berechnet werden. Dies ist allerdings nur bei kleinen Informationsmengen, wie bei der markerbasierten Registrierung möglich. Die zweite Art der Optimierung ist eine Suche nach den Transformations-Parametern. Hierbei wird eine Funktion über die Parameter erstellt. Diese beschreibt das Registrierungs-Paradigma auf mathematische Weise. Durch eine

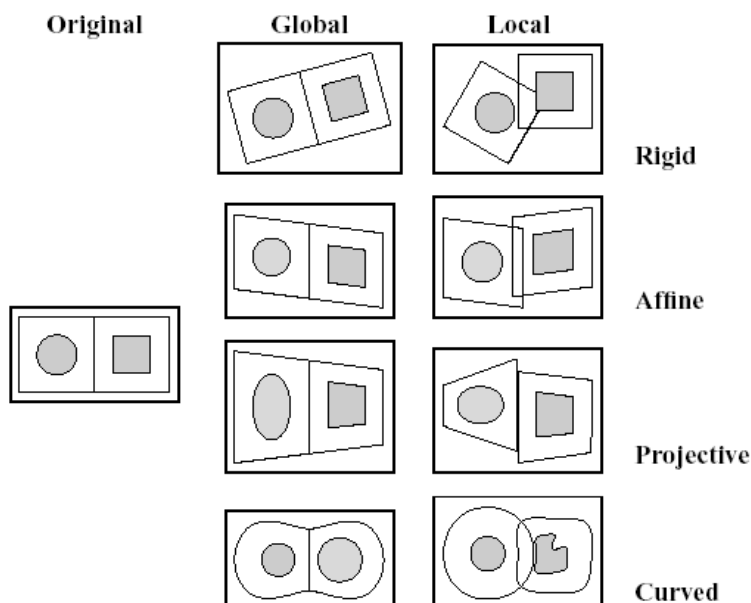


Abbildung 2.12.: In dieser Abbildung sind beispielhaft vier in der medizinischen Registrierung verwendete Transformationen dargestellt. Die in dieser Arbeit verwendeten rigiden Transformationen beinhalten Translation und Rotation. Affine Transformationen bewahren die Parallelität von Geraden, während projektive Transformationen nur noch Geraden auf Geraden abbilden. Curved Transformationen sind am mächtigsten, da sie Objekte beliebig verformen können. Quelle: [MV96].

Optimierung dieser Funktion werden dann die Datensätze registriert. Für die MIR wurde eine Optimierung mit dem SIMPLEX-Algorithmus durchgeführt.

Besonders häufig werden die in diesem Kapitel beschriebenen CT-, MRT- und PET-Daten miteinander registriert. In klinischen Anwendungen vor allem in der Kardiologie treten aber auch Ultraschalldaten auf (vgl. [PMV03], S.996). Der (7.) Punkt Modalitäten unterteilt sich in die Bereiche monomodale und multimodale Registrierung. Während die monomodale Registrierung häufig für Verlaufskontrollen genutzt wird, können mit der multimodalen Registrierung die Vorteile der Bildgebungsverfahren kombiniert werden. Die MIR ermöglicht die Registrierung unterschiedlicher Modalitäten.

Punkt (8.) klassifiziert mögliche Anwendungen in intra- und inter-Subjekt Registrierung. Erstere verwendet ausschließlich Daten eines Patienten, wohingegen die inter-Subjekt Registrierung unterschiedliche Patienten miteinander registriert, um zum Beispiel statistische Informationen über die Ausprägung von krankhaften Veränderungen zu erhalten. Für die Visualisierung in dieser Arbeit wurden Datensätze des gleichen Patienten (intra-Subjekt) miteinander registriert.

Es werden Datensätze von fast allen Bereichen des menschlichen Körpers miteinander registriert. Angefangen beim Kopf mit Schädelknochen und Gehirn, über Thorax und Abdomen, bis hin zu den Extremitäten. Einen sehr ausführlichen Überblick über eine Klassifikation der medizinischen Registrierung geben die Artikel von J. Maintz, M. Viergever et al. [PMV03], [MV98], [MV96].

3. Multimodale Visualisierung medizinischer Volumendatensätze

In the context of biomedical imaging, the term *fusion* refers to the combination of complementary information obtained from multiple data sets into a final image.

CHRISTOF REZK-SALAMA,
AUS [RS01]

Eine Visualisierung ist immer von Zielen geprägt, die es zu erreichen gilt. Die Eignung einer Visualisierung hängt von diesen Zielen und von den darzustellenden Daten ab. Eine geeignete Visualisierung ermöglicht es dem Betrachter die gesamte Information, die in den dargestellten Daten beinhaltet ist, wahrzunehmen und als Grundlage für Entscheidungen zu nutzen (vgl. [SM00], S. 7). Eine solche Visualisierung für eine Menge von Daten zu finden ist keine triviale Aufgabe. Möglichst viel Information muss verständlich, ohne unnötigen Interpretationsaufwand durch den Betrachter und mit minimalem Gestaltungsaufwand vermittelt werden. In den folgenden Abschnitten werden verschiedene Visualisierungsmöglichkeiten vorgestellt. Neben der in dieser Arbeit verwendeten direkten Volumenvisualisierung anhand von Transferfunktionen werden auch die beiden alternativen Formen kurz erläutert. Die wichtigsten Eigenschaften der drei analysierten Methoden sind in Tabelle A.1 im Anhang übersichtlich zusammengefasst.

3.1. Anforderungen an die zu erzeugende Visualisierung

Die bestehende Software MIP ist um eine 3D-Visualisierung zu erweitern, welche zwei Datensätze unabhängig vom verwendeten bildgebenden Verfahren gemeinsam darstellen kann. Dabei sollen die in den Datensätzen enthaltenen Strukturen vollständig abgebildet und ihre Grenzen betont werden. Neben der Darstellung einzelner Strukturen soll Kontextinformation anhand weiterer Strukturen aus dem gleichen Datensatz und komplementäre Information aus anderen Modalitäten zur Visualisierung hinzugefügt werden können. Die Visualisierung ist ohne eine aufwändige und fehleranfällige Vorbereitung

der darzustellenden Daten, wie z.B. eine Oberflächenextraktion, zu erzeugen. Falls möglich ist für das Rendering der Daten auf Hardwareunterstützung zurückzugreifen. Die notwendigen Einstellungen für das Rendering sollen für den Nutzer intuitiv und leicht veränderbar sein. Der Nutzer muss die Visualisierung mit möglichst geringem Aufwand erzeugen können. Dazu zählt auch, dass die Fusion der Daten automatisch erfolgt und nicht durch den Nutzer über eine Fusionsfunktion beschrieben werden muss. Änderungen der Visualisierungsparameter Beleuchtung und Kameraposition sollen in Echtzeit umgesetzt werden. Auch eine nachträgliche Veränderung der Registrierung der Datensätze soll interaktiv durchführbar sein und ist somit möglichst schnell zu aktualisieren. Nachdem die Visualisierung erstellt wurde, sollen interaktive Betrachtungsmöglichkeiten, wie z.B. zoomen und rotieren, zur Verfügung stehen. Weiterhin muss das Visualisierungssystem mit den Programmbibliotheken ITK (Insight Toolkit), VTK (Visualization Toolkit) und wxWidgets implementiert werden.

3.2. Schichtbilder und deren Eignung für das Visualisierungssystem

Die in der klinischen Routine am weitesten verbreitete Methode medizinische Volumendaten zu analysieren, ist anhand der schichtweisen Darstellung. Dabei werden die Schichtbilder entweder wie Röntgenbilder ausbelichtet oder direkt an einem Monitor betrachtet. Die Darstellung der Bilddaten erfolgt am Rechner in den drei zueinander orthogonalen Orientierungen axial, coronal und sagittal. Die Schichtbilddarstellung der Datensätze ist in der zu erweiternden Software MIP bereits implementiert und die vorhandene Möglichkeit zur multimodalen Darstellung von registrierten Datensätzen wird im Folgenden vorgestellt.

Die Darstellung der Daten erfolgt auf Standardmonitoren meist in 256 Graustufen, während spezielle Monitore für die digitale Radiologie auch bis zu 1024 Grauwerte darstellen können. Da z.B. Computertomographiedaten, wie in Abschnitt 2.2.1 beschrieben, mit 4096 Hounsfield Einheiten aufgenommen werden, müssen die gemessenen Werte häufig auf 256 Graustufen abgebildet werden. Um einen Informationsverlust zu vermeiden, wird eine spezielle Technik für die Einschränkung des Grauwertbereichs eingesetzt. Beim sogenannten *Window & Level* Verfahren werden anhand von zwei Parametern die darzustellenden Grauwerte aus dem Grauwertbereich $G = \{0, \dots, 4095\}$ ausgewählt. Damit erhöht sich der Kontrast im Bild gegenüber der Abbildung aller 4096 Messwerte. Der Parameter *Window Width* gibt an, wie viele Messwerte auf die 256 Graustufen abgebildet werden und damit wie stark der Kontrast verstärkt wird. Durch den *Window Level* Parameter kann die Lage dieses Bereiches verändert werden. Die beiden Parameter bestimmen demnach, welche Gewebe im Bild sichtbar sind. Häufig werden die sogenannten „Lungenfenster“, „Weichteilfenster“ und „Knochenfenster“ verwendet. Abbildung 3.1 zeigt die Fensterung beispielhaft an einem Schädel-CT.

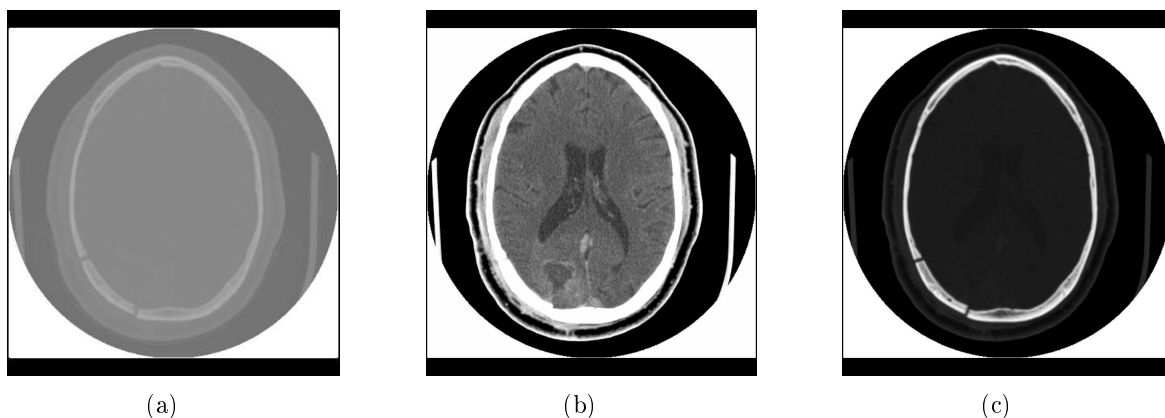


Abbildung 3.1.: Die Beschränkung des Grauwertbereichs, welcher auf die 256 darstellbaren Grauwerte abgebildet wird, kann den Kontrast im Bild erheblich verbessern. In (a) wurde keine Fensterung vorgenommen. Es sind weder die Hirnstrukturen noch die Knochen gut erkennbar. Auf Abb. (b) wurde eine Weichteilfensterung angewendet. Der Kontrast im Weichgewebe (Gehirn) konnte so erheblich verbessert werden. Allerdings werden Knochen so völlig überblendet dargestellt. Um den Kontrast im Bereich der Knochen anzuheben, wird das Knochenfenster (c) verwendet.

Multimodale Visualisierung mit Schichtbildern

Die multimodale Visualisierung wird bei der Schichtbilddarstellung z.B. über ein Schachbrettbild möglich mit dem zwei Schichtbilder parallel angezeigt werden. Das Bild wird dazu in eine vorgegebene Anzahl von Feldern unterteilt. Jedes Feld zeigt den Ausschnitt aus einem der beiden Schichtbilder, der sich an der Position des Feldes befindet (s. Abb. 3.2).

Durch das Schachbrettmuster ist es möglich, manuelle Registrierungen zu überwachen oder automatische Registrierungen zu überprüfen. Die Schichtbilddarstellung verbindet eine schnelle Visualisierung beider Volumen mit der Möglichkeit einen genauen Blick auf den Bildinhalt zu werfen. Dadurch kann die Registrierung anhand der Lage von Knochenstrukturen oder anderen anatomischen Landmarken überprüft werden. Die Lokalisierung von krankhaften Veränderungen erfordert eine interaktive Verschiebung des Schnittpunktes bei nur einer Unterteilung oder die sukzessive Unterteilung des Schachbrettmusters, um zu erreichen, dass an der Position der Veränderung die richtige Modalität angezeigt wird. Da die Lage der Felder bei mehr als einer Unterteilung nicht frei wählbar ist, ist es allerdings möglich, dass eine gewünschte Unterteilung für eine bestimmte Blickrichtung nicht definiert werden kann. In diesem Fall muss die Reihenfolge der Datensätze getauscht werden um an der Position der interessierenden Struktur den richtigen Datensatz anzuzeigen. In bestimmten Fällen wird der Effekt allerdings nur auf eine oder beide der anderen Blickrichtungen verschoben, so dass dann in diesen die gewünschte Unterteilung nicht erreicht werden kann (vgl. Abschnitt 6.2). Analog zur Schachbrettdarstellung könnte auch ein Interlacing, d.h. ein abwechselndes Darstellen der Bildzeilen beider Schichtbilder implementiert werden. Dies entspricht einem sehr feinen Schachmuster, das allerdings nur in der y-Dimension unterteilt wurde.

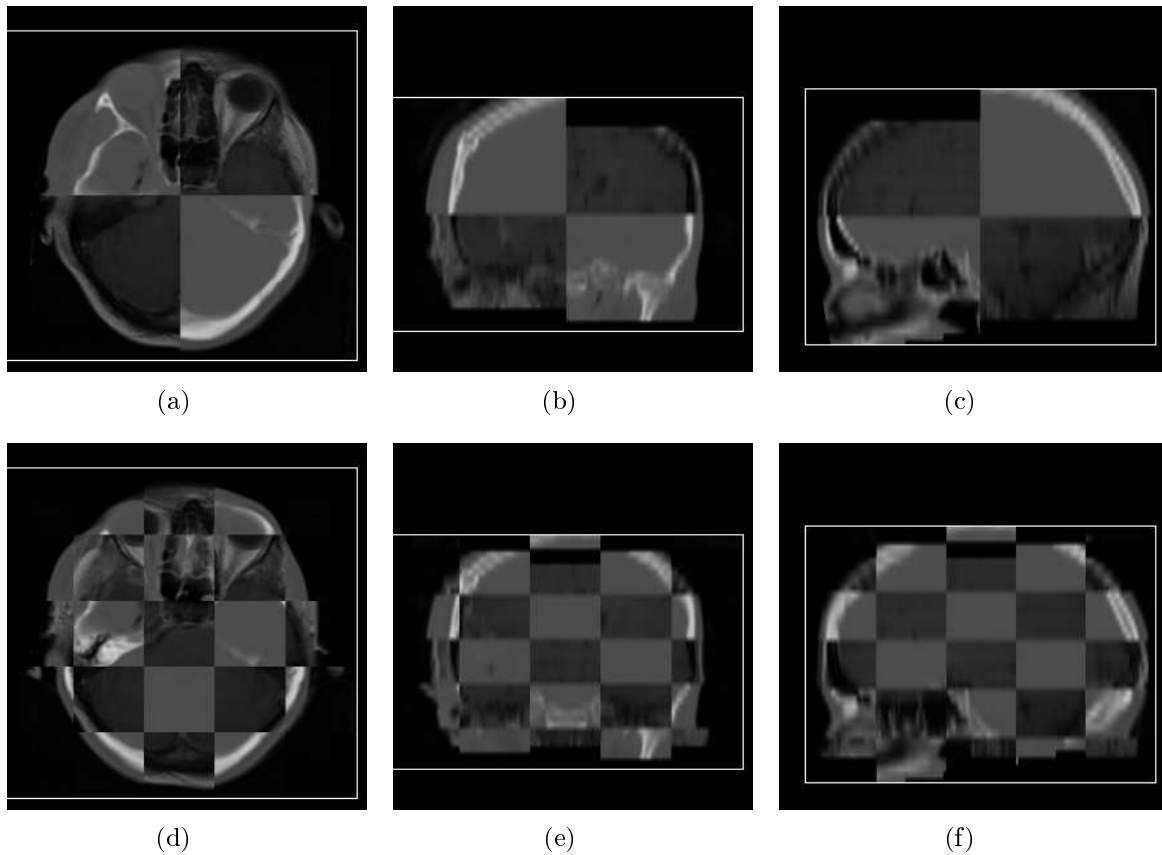


Abbildung 3.2.: Die Abbildungen zeigen gleichzeitig eine CT- und eine MRT-Aufnahme nachdem diese miteinander registriert wurden. Um beide Volumenaufnahmen darzustellen, wird eine Art Schachbrettbild verwendet. Die Felder zeigen abwechselnd die Bereiche aus den beiden Originalbildern, welche sich an der Position des Feldes befinden. Die Anzahl der Felder ist dabei wählbar. In den Abbildungen (a)-(c) wurde je eine Unterteilung in x- und y-Richtung des Bildes gewählt, während die Abbildungen (d)-(f) die gleichen Ansichten bei vier Unterteilungen zeigen.

Eine denkbare Alternative ist die Fusion der Grauwert- oder Farbdarstellungen der Schichtbilder. Im einfachsten Fall werden die Daten über eine Wichtung miteinander fusioniert, wobei die Summe der beiden Gewichte 1 ergibt (vgl. [Pen99]). Sollen die speziellen Eigenschaften der Datenwerte beachtet werden, muss eine Funktion definiert werden, die basierend auf den Intensitätswerten und ggf. den berechneten Gradientenwerten für jedes Pixel entscheidet, welche der möglichen Informationen aus den beiden Datensätzen angezeigt wird oder ob ein neuer Datenwert aus beiden Informationen berechnet wird. In [PTF02] werden solche Funktionen daher als *Decision Functions* bezeichnet. Eine solche Entscheidungsfunktion muss vor der Visualisierung durch den Betrachter in Abhängigkeit von der Fragestellung definiert werden. Diese Art der Schichtbilddarstellung hat den Vorteil, dass an allen Positionen innerhalb der Schicht die für den Nutzer interessante Bildinformation dargestellt wird. Allerdings kann durch eine falsche Entscheidung auch eine Visualisierung entstehen, die zu Fehlinterpretationen führt.

Eignung der multimodalen Schichtbilddarstellung für das Visualisierungssystem

Schichtbilddarstellungen lassen sich sehr schnell und einfach aus den Volumendaten erzeugen. Sie bieten eine gute Übersicht über den Inhalt einer Schicht, da keine Verdeckungen auftreten können. Gleichzeitig erlauben sie durch die hohe Ortsauflösung innerhalb einer Schicht einen genauen Einblick ins Innere des Volumens. Auch multimodale Visualisierungen lassen sich durch Schichtbilder erzeugen, indem die Daten z.B. über ein Schachbrettmuster dargestellt werden. Die Vorteile der Schichtbilddarstellung mehrerer Datensätze sind die Möglichkeit die Genauigkeit einer Registrierung sehr gut mit dem Auge abschätzen zu können sowie Strukturen in den Datensätzen parallel genau zu betrachten und basierend auf der neu gewonnenen Information Schlüsse zu ziehen oder die Daten weiter zu verarbeiten z.B. durch eine Segmentierung. Um einen Datensatz zu untersuchen, ist allerdings eine zeitaufwändige Betrachtung aller Schichten nötig. Dies wird durch die Verbesserung der Auflösung zwischen den einzelnen Schichten und der daraus resultierenden hohen Schichtanzahl immer schwieriger. Vor allem für die Kommunikation von Diagnoseergebnissen an Nichtmediziner sind Schichtbilddarstellungen wenig geeignet. Ein weiterer Nachteil ist die fehlende Volumeninformation, welche dem Arzt als Kontext für eine Diagnose sehr hilfreich sein kann. Multimodale Darstellungen anhand von Schichtbildern erfüllen somit nicht die Anforderung einer 3D-Visualisierung der Daten. Aus diesem Grund werden in den folgenden Abschnitten die beiden Arten der Volumenvisualisierung untersucht.

3.3. Indirekte Volumenvisualisierung und deren Eignung für das Visualisierungssystem

Dieser Abschnitt beschäftigt sich mit der indirekten Volumenvisualisierung. Anders als bei der rein zweidimensionalen Schichtbilddarstellung, wird bei der indirekten Volumenvisualisierung eine dreidimensionale Repräsentation der Datensätze generiert. So werden z.B. Oberflächen extrahiert und dann gerendert.

Indirekte Visualisierungspipeline

Die für die Generierung einer Visualisierung nötigen Schritte werden in der allgemeinen Visualisierungspipeline zusammengefasst. Diese Pipeline besteht aus dem *Filtering*, *Mapping*, *Shading* und *Rendering* (vgl. Abb. 3.3).

Das *Filtering* ist ein Vorverarbeitungsschritt, in dem die Daten z.B. geglättet werden. Beim *Mapping* wird die später zu visualisierende geometrische Repräsentation der Daten generiert. Im Schritt *Shading* wird anhand der verwendeten Lichtquellen eine Beleuchtungsberechnung und Schattierung des Datensatzes vorgenommen. Das *Rendering* bereitet die Daten auf die Ausgabe auf einem Anzeigegerät vor und veranlasst den Zeichenprozess durch die Grafikkarte (vgl. [SM00], [SML02] und [RS01]). Abbildung 3.4 zeigt die Visualisierungspipeline für die indirekte Volumenvisualisierung.

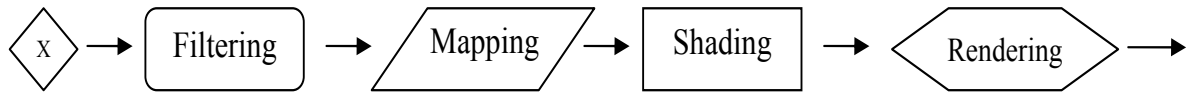


Abbildung 3.3.: In der allgemeinen Visualisierungspipeline werden die Daten nach dem Vorverarbeitungsschritt (*Filtering*) im Schritt *Mapping* in eine andere Repräsentation überführt. Die modifizierten Volumendaten werden anschließend ggf. beleuchtet (*Shading*) und gerendert (*Rendering*).

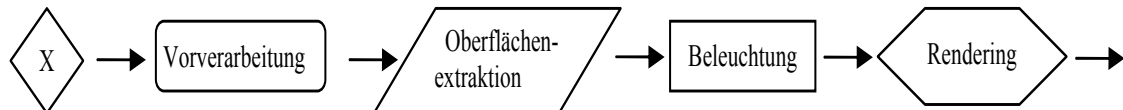


Abbildung 3.4.: Nach dem Vorverarbeitungsschritt (*Filtering*) werden die Daten im Schritt *Mapping* in eine andere Repräsentation überführt. Dies ist hier durch eine Oberflächenextraktion dargestellt. Die so erzeugten Geometriedaten werden anschließend beleuchtet (*Shading*) und gerendert (*Rendering*).

Die Besonderheit der indirekten Volumenvisualisierung liegt in den Daten, welche im Schritt *Rendering* auf dem Bildschirm angezeigt werden. Aus den skalaren Voxeldaten werden beim *Mapping* Geometriedaten (z.B. Dreiecksnetze) erzeugt, die dann angezeigt werden. Diese können sehr effektiv von modernen Grafikkarten verarbeitet werden. Der räumliche Eindruck einer indirekten Volumenvisualisierung wird durch Beleuchtungsrechnungen (*Shading*) verstärkt.

Oberflächenextraktion

Die Oberfläche, welche die Position, Größe und Form einer gesuchten Struktur im Datensatz repräsentiert, wird anhand eines gewählten Schwellwertes aus dem Volumen extrahiert. Da alle Punkte auf dieser Oberfläche den gleichen Datenwert besitzen, wird sie als Isooberfläche bezeichnet. Für die Extraktion der Isooberflächen aus Volumendatensätzen hat sich der *Marching Cube* Algorithmus als Standardverfahren etabliert [LC87]. Dabei wird mit Hilfe eines Würfels direkt in den Voxeln des Volumendatensatzes nach dem Verlauf der Isooberfläche gesucht und diese gegebenenfalls durch Dreiecke approximiert. Dafür werden die Punkte auf der Oberfläche durch trilineare Interpolation der Voxelwerte ermittelt.

Der *Marching Cube* Algorithmus erzeugt einzelne Dreiecke, die in ihrer Gesamtheit die Isooberfläche repräsentieren. Die Anzahl der Dreiecke ist für eine effektive Speicherung und Visualisierung meist zu hoch. Um den benötigten Speicherplatz zu reduzieren, erfolgt die Speicherung der Dreiecke als Dreiecksnetz. Darin muss jeder Eckpunkt eines Dreiecks nur einmal mit der Information, zu welchen Dreiecken er gehört, gespeichert

werden. Ein solches Dreiecksnetz kann anschließend durch die Vereinigung benachbarter Dreiecke zusammengefasst werden. Dieses Vorgehen wird auch als Dezimierung bezeichnet (vgl. [SZL92]). Als Entscheidungskriterium für die Vereinigung werden z.B. die Größe des entstehenden Dreiecks und die Differenz zwischen den Oberflächennormalen der zu vereinigenden Dreiecke herangezogen (vgl. [SZL92]).

Da die Extraktion der gesuchten Oberflächen aus großen Datensätzen sehr zeitaufwändig ist, werden in klinischen Workstations verbesserte Versionen des *Marching Cube* Algorithmus verwendet. Lewiner et al. [LLVT03] geben ein Beispiel für die Beschleunigung des Verfahrens und die Möglichkeit der Hardwareunterstützung. Eine Weiterentwicklung des *Marching Cube* Algorithmus ist der *Marching Tetrahedra* Algorithmus, bei dem anstelle eines Würfels ein Tetraeder durch den Datensatz wandert. Der Vorteil ist die Eliminierung der Mehrdeutigkeiten des Oberflächenverlaufs, die beim *Marching Cube* Algorithmus auftreten können. Allerdings verdreifacht sich die Anzahl der extrahierten Dreiecke (vgl. [ST90]).

Oberflächenrepräsentationen haben neben dem schnellen Rendering den Vorteil, dass Flächeninhalt der Oberfläche und Volumen des extrahierten Objektes schnell berechnet werden können. Die extrahierte Oberfläche z.B. eines Organs ist durch die notwendigen Interpolationen immer nur eine Approximation der eigentlichen Oberfläche. Die Genauigkeit ist hierbei von dem gewählten Schwellwert und der Qualität der echten Isooberfläche in den Daten abhängig. Mögliche Ungenauigkeiten können dem Betrachter z.B. durch Unschärfe vermittelt werden.

Ein geeigneter Schwellwert, mit dem allein die interessierende Oberfläche vollständig extrahiert wird, kann häufig nicht gefunden werden. Datenwerte, die nicht dem Schwellwert entsprechen, werden nicht dargestellt. Somit kann durch die Wahl eines falschen Schwellwertes eine fehlerhafte Visualisierung erzeugt werden. Zum einen geht Information verloren, indem Teile der interessierenden Oberfläche nicht gerendert werden. Zum anderen kann es zu einer verfälschten Darstellung kommen, indem Oberflächen miteinander verbunden werden, die im Volumen nicht zusammenhängen. Da grundsätzlich alle Isooberflächen mit dem gewählten Schwellwert extrahiert werden, kann es zu einer hohen Anzahl an Oberflächen kommen. Artefakte, die sich z.B. aus Inhomogenitäten in den Messdaten ergeben, müssen von der gewünschten Information getrennt werden.

Neben der Oberflächenextraktion direkt aus dem Volumendatensatz, wird häufig mit Segmentierungsergebnissen gearbeitet. Dies bietet sich an, wenn für die Volumetrie ohnehin eine Segmentierung von wichtigen Regionen durchgeführt werden muss. Die Segmentierung der Daten fällt in den Schritt *Filtering* der Visualisierungspipeline. Aus den gewonnenen Segmentierungsergebnissen lässt sich eine Oberfläche extrahieren, indem die äußeren Voxel miteinander zu einem Dreiecksnetz verbunden werden. Dies hat den Vorteil, dass die Anzahl der Voxel, die für die Oberflächengenerierung betrachtet werden müssen, sehr gering ist. Allerdings hängt die Qualität der Oberfläche stark von der Qualität des Segmentierungsverfahrens ab.

Indirekte Volumenvisualisierung für multimodale Daten

Die indirekte Visualisierung mehrerer Volumendatensätze erfordert die Extraktion aller relevanten Oberflächen aus den Daten. Die erzeugten geometrischen Repräsentationen werden dann an die Grafikkhardware für das Rendering übergeben. Bei der Multivolumen-Visualisierung machen sich zwei Nachteile dieses indirekten Verfahrens besonders bemerkbar. Zum einen muss die Oberflächengenerierung für alle anzuzeigenden Oberflächen durchgeführt werden. Dies erhöht die Rechenzeit vor der eigentlichen Visualisierung und erfordert Interaktion des Nutzers, der die Schwellwerte angeben muss. Zum anderen gehen Kontextinformationen, wie die Lage umgebender Organe und Gewebe, verloren. Für die Registrierung von Datensätzen ist Kontextinformation oft hilfreich, da mehr Anhaltspunkte für die Beurteilung der Qualität einer Registrierung vorhanden sind. Abbildung 3.5(a) zeigt ein indirektes Volumenrendering eines CT- und eines PET-Datensatzes. Einen Einblick auf innere Strukturen bieten Isooberflächendarstellungen bei denen mit Transparenz gearbeitet wird. Eine Visualisierung mehrerer Isooberflächen des Zahn CT-Datensatzes aus [PLB⁺01] ist in 3.5(b) dargestellt. Hierbei wurde den Oberflächen eine Transparenz zugewiesen, um die extrahierten Strukturen im Inneren des Zahnes sichtbar zu machen.



Abbildung 3.5.: (a) Darstellung zweier registrierter Volumendatensätze mittels indirekter Volumenvisualisierung. Die Abbildung zeigt die Kombination eines CT- und eines PET-Datensatzes. Die Knochen aus dem CT dienen dazu, die Bereiche des PET-Datensatzes, in denen verstärkte Stoffwechselaktivität vorliegt, in den anatomischen Kontext einordnen zu können. (b) Mehrere Isooberflächen wurden aus einem synthetischen Zahndatensatz extrahiert. Um auch die Oberflächen im Inneren des Volumens sehen zu können, wurde den Oberflächen eine Transparenz zugewiesen. Quelle: (b) [MC].

Eignung der indirekten multimodalen Volumenvisualisierung für das Visualisierungssystem

Die Extraktion der Isooberflächen ist durch ihren hohen Aufwand und ihre Fehleranfälligkeit der Hauptnachteil der indirekten Volumenvisualisierung. Sämtliche darzustellenden Oberflächen müssen durch den Nutzer aus den Datensätzen extrahiert werden. Die Extraktion muss bei Änderungen an den Daten ggf. wiederholt werden. Durch die Extraktion der Isooberfläche geht Information über die Daten unterhalb der Oberfläche, wie z.B. die Strukturiertheit des Gewebes, verloren. Die Anforderungen an die zu erstellende Visualisierungskomponente verlangen jedoch, dass die Strukturen nicht auf ihre Oberfläche reduziert werden. Somit erfüllt die indirekte Volumenvisualisierung zwei wichtige Anforderungen an das zu entwickelnde Visualisierungssystem nicht. Eine Alternative stellt die direkte Volumenvisualisierung dar, die im folgenden Abschnitt analysiert wird.

3.4. Direkte Volumenvisualisierung

Die direkte Volumenvisualisierung stellt die Originaldaten eines 3D-Datensatzes direkt und möglichst detailgetreu dar. Die Daten werden nicht, wie bei der indirekten Volumenvisualisierung, in eine geometrische Repräsentation umgewandelt. Für das Rendering der Daten wird falls möglich auf die Unterstützung der Grafikkhardware zurückgegriffen. Dieser Abschnitt erläutert die grundlegenden Eigenschaften der direkten Volumenvisualisierung und analysiert mögliche Formen der Visualisierungspipeline für die multimodale Visualisierung. Weiterhin wird die Verwendbarkeit für die Erzeugung multimodaler Visualisierungen untersucht.

3.4.1. Direkte monomodale Volumenvisualisierung

Die direkte Visualisierungspipeline stellt die Verarbeitungsschritte dar, die für eine direkte Visualisierung der Volumendaten nötig sind (s. Abb. 3.6).

Im Schritt *Filtering* werden die Volumendatensätze häufig durch eine Medianfilterung geglättet. Sehr häufig wird außerdem die Berechnung der Gradienteninformation (Stärke und Richtung des Gradienten) durchgeführt. Die Gradientenstärke (Betrag des Gradienten oder auch *Gradient Magnitude*) beschreibt die Stärke einer Kante und wie nah sich das jeweilige Voxel an dieser Kante befindet. Die Gradientenrichtung gibt an, in welcher Richtung sich die Kante vom jeweiligen Voxel aus gesehen befindet. Diese Information wird für die Erstellung der direkten Volumenvisualisierung mit Hilfe von mehrdimensionalen Transferfunktionen benötigt (vgl. Abschnitt 4.3).

Im Schritt *Mapping* werden die gemessenen Skalarwerte und die berechneten Gradientenwerte eines Voxels durch Transferfunktionen auf Farb- und Durchsichtigkeitswerte abgebildet. Die definierten Transferfunktionen werden im Schritt *Rendering* verwendet, um eine 3D-Darstellung der Volumendaten zu erzeugen. Ziel der Abbildung durch Transferfunktionen ist es, die relevanten Bereiche innerhalb des Volumens hervorzuheben, ohne dazu eine aufwändige Segmentierung durchführen zu müssen.

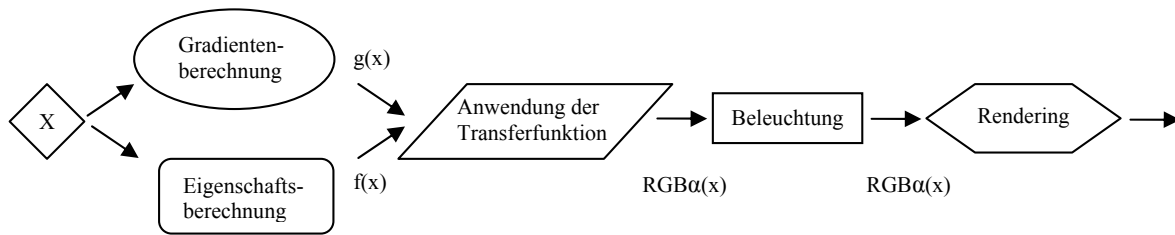


Abbildung 3.6.: Nach dem Vorverarbeitungsschritt (*Filtering*) wird den Voxeln des Datensatzes im Schritt *Mapping* anhand ihrer Eigenschaften eine Farbe und Transparenz zugewiesen. Dies wird mittels Transferfunktionen erreicht. Der räumliche Eindruck bei der Volumendarstellung kann durch Beleuchtungseffekte (*Shading*) verstärkt werden. Die Voxel werden anschließend gerendert.

3.4.2. Direkte multimodale Volumenvisualisierung

Die multimodale Visualisierung erlaubt die kombinierte Darstellung mehrerer miteinander registrierter Volumendatensätze (vgl. 2.6) auch unterschiedlicher Aufnahmemodalitäten. So können beispielsweise CT-Aufnahmen den anatomischen Kontext für funktionelle Information aus PET-Aufnahmen bilden. Dies kann die Kommunikation von diagnostizierten Funktionsstörungen erleichtern. Vier Eigenschaften charakterisieren die unterschiedlichen Möglichkeiten mehrere Volumen gleichzeitig darzustellen. In [FPT02] werden diese Visualisierungsmöglichkeiten ausführlich auf ihre Eignung untersucht.

Art der Fusion Zunächst muss zwischen zwei Arten der Datenfusion unterschieden werden. Bei der Darstellung durch den Renderer kann entweder pro Voxel nur eine Eigenschaft (*One Property per Point*, OPP) oder verschiedene Eigenschaften (*Multiple Properties per Point*, MPP) dargestellt werden. Die OPP-Variante bildet z.B. die Skalarwerte zweier Volumen gemeinsam auf Farbe ab. Zum einen ist dabei eine komplementäre Darstellung der Daten möglich, bei der beide Volumen gleichermaßen zum Wert eines Punktes im gerenderten Bild beitragen (z.B. MRT und CT). Zum anderen wird bei der supplementären Methode für jedes Voxel entschieden, welcher Datensatz zu rendernden Wert beisteuert. Hierbei ist es auch möglich, dass aus den beiden Datensätzen unterschiedliche Eigenschaften für die Visualisierung herangezogen werden (z.B. MRT und PET). Diese Fusion der Daten ist von den Vorüberlegungen her ein aufwändiger Prozess. Die Auswahl von Eigenschaften ist mit einer Segmentierung zu vergleichen. Es müssen für alle Kombinationen von Modalitäten eigene Fusionsfunktionen angegeben werden. Da die Visualisierung möglichst automatisch und ohne die Definition einer Fusionsfunktion erzeugt werden soll, werden in dieser Arbeit nur komplementäre OPP-Visualisierungen generiert. Bei der MPP-Visualisierung werden die Eigenschaften zweier Datensätze auf unterschiedliche Visualisierungsparameter abgebildet. So kann die eine Modalität z.B. auf Grauwerte im gerenderten Bild abgebildet werden, während die Voxelwerte des anderen Datensatzes Farbwerten entsprechen. Die Art der Fusion beschreibt somit, nach welchem Kriterium die darzustellenden Eigenschaften der Datensätze für das spätere Rendering ausgewählt werden.

Zeitpunkt der Datenfusion Das zweite Kriterium für die Wahl einer Visualisierungsmethode ist der Zeitpunkt der Datenfusion. Der Fusionszeitpunkt bestimmt, ob für jedes Volumen eine eigene Transferfunktion definiert werden muss, oder ob für die Visualisierung eine gemeinsame Transferfunktion verwendet werden kann. Nach dem Zeitpunkt werden die in Abschnitt 3.4.3 vorgestellten Visualisierungspipelines unterschieden.

Zeitpunkt der Registrierung Der Zeitpunkt der Registrierung der Datensätze fließt als drittes Kriterium mit in den Entwurf des Visualisierungssystems ein. Für die Erstellung der Visualisierung ist entscheidend, ob die Daten bereits vor der Eingabe in die Visualisierungspipeline transformiert wurden, oder ob die im Registrierungsschritt ermittelte Transformation während des Renderings auf die Volumenelemente angewendet werden muss. Letzteres würde die benötigte Rechenzeit für ein erneutes Rendering der Visualisierung (z.B. nach Rotation der Kamera) erhöhen, auch wenn die Registrierungsparameter nicht geändert wurden. Der Nachteil der Transformation vor der Visualisierungspipeline ist, dass durch die für Transformation und Angleichung der Voxelgröße benötigte Interpolation ein Fehler in die Daten eingebracht wird. Allerdings ist es ohne eine Vorabregistrierung nicht möglich einem Voxel eines Datensatzes den dazugehörigen Voxel aus dem anderen Datensatz zuzuordnen.

Veränderung der Visualisierungsparameter Die Aktualisierungszeit ist auch bei der Veränderung der Visualisierungsparameter Licht, Transferfunktion und Kameraposition das Kriterium für den Entwurf des Visualisierungssystems. Je nach verwendetem Fusionszeitpunkt muss nach der Veränderung der Visualisierungsparameter die gesamte Pipeline ggf. sogar für alle darzustellenden Datensätze erneut durchlaufen werden.

3.4.3. Pipelines der direkten multimodalen Volumenvisualisierung

Der Zeitpunkt der Datenfusion beeinflusst maßgeblich den Aufwand der nötig ist, um auf Veränderungen der Parameter der Visualisierung zu reagieren. In diesem Abschnitt sollen die verschiedenen Fusionszeitpunkte und die dazugehörigen Visualisierungspipelines vorgestellt werden. In Anlehnung an [FPT04] werden die Messwerte der Datensätze als *Property* und die optischen Parameter eines Voxels als *Material* bezeichnet.

Property Fusion Die *Property Fusion* beschreibt die Fusion der Messwerte der Datensätze zu Beginn der Visualisierungspipeline. Anschließend werden die fusionierten Daten mit den daraus ermittelten Gradienteninformationen an die Visualisierungspipeline weitergegeben. Für die Fusion werden die darzustellenden Eigenschaften aus den Volumen ausgewählt bzw. neue Eigenschaften berechnet. Dies hängt davon ab, wieviele Eigenschaften in der Visualisierung dargestellt werden sollen (OPP oder MPP) und ob die Daten die gleichen Eigenschaften darstellen (OPP-komplementär) oder sich gegenseitig ergänzen (OPP-supplementär). Abbildung 3.7 verdeutlicht dies.

In dieser Arbeit wurde die *Property Fusion* verwendet, um OPP-Visualisierungen komplementärer Daten zu erzeugen. Somit wird aus den Eigenschaften beider Datensätze auf den darzustellenden optischen Parameter geschlossen. Für die Fusion müssen

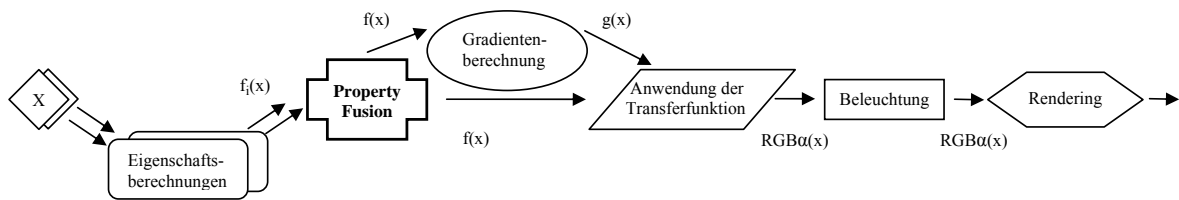


Abbildung 3.7.: Bei der *Property Fusion* werden die Daten der Volumen bereits im ersten Schritt der Pipeline miteinander fusioniert. Aus den Eingangsdaten $f_i(x)$ werden dabei die darzustellenden Eigenschaften ausgewählt (OPP-supplementär) bzw. eine gemeinsame neue Eigenschaft berechnet (OPP-komplementär). Dadurch entsteht ein gemeinsamer Volumendatensatz mit den gewählten Voxel-eigenschaften $f(x)$, der anschließend die Visualisierungspipeline durchläuft. Damit wird in der *Property Fusion* Pipeline nur eine Transferfunktion für die beiden darzustellenden Datensätze verwendet.

demnach die korrespondierenden Intensitätswerte für alle Positionen im Volumen nur zusammengetragen und nicht, wie im supplementären Fall, über eine Fusionsfunktion ausgewählt werden. Der Hauptvorteil die Daten bereits sehr früh zu fusionieren liegt darin, dass die Pipeline sehr schnell auf Änderungen der Visualisierungsparameter reagiert und somit die Darstellung der Daten in Echtzeit aktualisieren kann. Sollen allerdings die Registrierungsparameter geändert werden, so muss die gesamte Visualisierungspipeline erneut durchlaufen werden. Da die Fusion vor der Berechnung der Materialeigenschaften liegt, wird hier für beide Volumen eine gemeinsame Transferfunktion verwendet.

Property & Gradient Fusion Die *Property & Gradient Fusion* berechnet zunächst die Gradienteninformation unabhängig für jeden Datensatz. Die Gradienteninformation wird dazu benutzt, Kanten in der resultierenden Visualisierung hervorzuheben. Die Kanteninformation ist hilfreich bei der Beurteilung der Registrierungsqualität. Die Messwerte und Gradienten aller Datensätze werden anschließend dazu herangezogen, eine gemeinsame Materialeigenschaft wie z.B. Transparenz und Farbe über eine Transferfunktion zu bestimmen (s. Abbildung 3.8). Die Eigenschaften der einzelnen Datensätze und ihre Gradienteninformationen werden analog zur *Property Fusion* zusammengefasst bzw. die darzustellenden Werte ausgewählt. Die fusionierten Werte werden gemeinsam mit der Materialeigenschaft an die nächsten Schritte der Visualisierungspipeline übergeben. Änderungen an der gemeinsamen Transferfunktion, welche für die Bestimmung der Materialeigenschaft genutzt wird, beeinflussen die Fusion. Somit müssen nach einer Änderung der Transferfunktion alle Schritte der Visualisierungspipeline erneut ausgeführt werden. Dies ist auch bei einer Modifikation der Registrierung der Fall. Änderungen der Beleuchtung und Kameraposition können, wie bei der *Property Fusion*, sehr schnell aktualisiert werden.

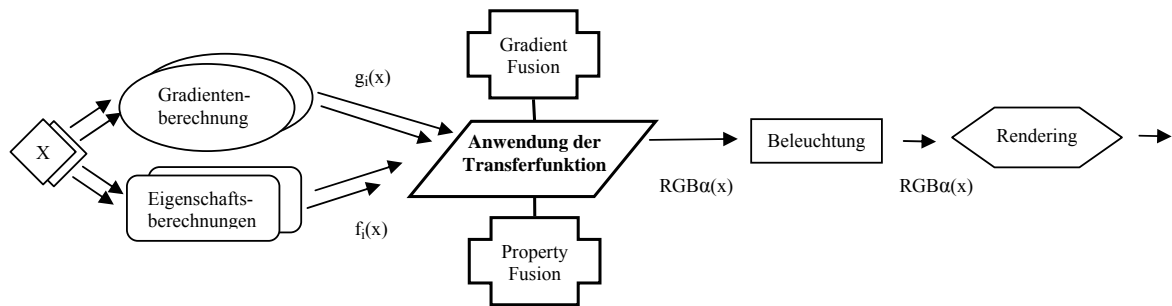


Abbildung 3.8.: Bei der *Property & Gradient Fusion* werden für beide Volumen zunächst die Gradientenwerte $g_i(x)$ berechnet. Über diese und die Eingangsdaten $f_i(x)$ wird anschließend eine Transferfunktion definiert. Durch Anwendung dieser Transferfunktion entsteht ein Volumendatensatz, mit den optischen Eigenschaften $RGB\alpha(x)$, der den Rest der Visualisierungspipeline durchläuft. Damit wird auch in der *Property & Gradient Fusion* Pipeline nur eine Transferfunktion für die beiden darzustellenden Datensätze verwendet.

Material Fusion Die *Material Fusion* ist eine Fusion der optischen Parameter der Datensätze. Die Gradientenwerte werden parallel für alle Volumen berechnet. Anschließend werden die Materialeigenschaften (Farbe und Transparenz) für jedes Volumen durch eine eigene Transferfunktion bestimmt (s. Abbildung 3.9). Die Fusion erfolgt über eine gewichtete Kombination dieser Materialeigenschaften, die durch eine sogenannte Fusionsfunktion beschrieben wird. Je nach Art der Gewichtung können so auch mehrere Eigenschaften zur Visualisierung beitragen (OPP = binär; MPP = variabel). Änderungen an dieser Fusionsfunktion erfordern ebenso wie Modifikationen der Registrierung nur eine erneute Fusion und eine anschließende Beleuchtungsberechnung. Ein weiterer Vorteil dieses Verfahrens liegt darin, dass die Visualisierungsparameter Kameraposition und Beleuchtung geändert werden können, ohne die gesamte Pipeline erneut auszuführen. Sollen allerdings die Transferfunktionen für die Abbildung der Skalarwert-/ Gradient-Kombination auf die Materialeigenschaften modifiziert werden, so muss die gesamte Pipeline erneut durchlaufen und die Fusionsfunktion evtl. aufwändig angepasst werden.

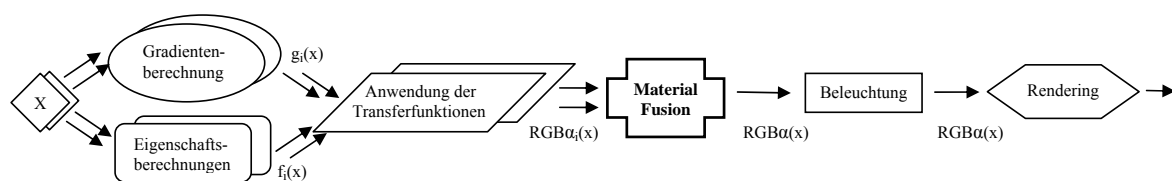


Abbildung 3.9.: Bei der *Material Fusion* werden für beide Volumen sowohl die Gradientenwerte $g_i(x)$, als auch die Materialeigenschaften $RGB\alpha_i(x)$ separat berechnet. Somit müssen zwei Transferfunktionen definiert werden. Die Materialeigenschaften werden anschließend über eine zu definierende Fusionsfunktion zu einer optischen Eigenschaft $RGB\alpha(x)$ je Voxel vereint, welche den Rest der Pipeline durchläuft.

Shading Fusion Grundsätzlich gleicht die *Shading Fusion* Pipeline der *Material Fusion*. Allerdings wird die Beleuchtung (engl. *Shading*) der Volumen noch separat vorgenommen (s. Abbildung 3.10). Dies erhöht die Unterscheidbarkeit der einzelnen Oberflächen, da die Reflektionen separat berechnet und anschließend gewichtet fusioniert werden. Da die Beleuchtungsberechnungen allerdings für jedes Volumen einzeln durchgeführt werden, erhöht sich die Rechenzeit für einen Durchlauf der Pipeline. Modifikationen der Registrierungsparameter erfordern eine erneute Beleuchtungsberechnung für das betreffende Volumen. Nach Änderungen an den Lichtquellen oder den Kameraparametern muss die Beleuchtung für jedes Volumen wiederholt werden. Modifikationen der Transferfunktionen der Volumeneigenschaften erfordern einen neuen Durchlauf der Pipeline. Die Fusion kann effektiv geändert werden.

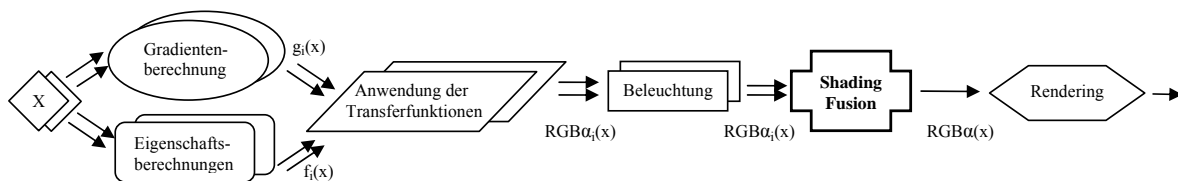


Abbildung 3.10.: Bei der *Shading Fusion* wird die Pipeline für beide Volumen bis zur Beleuchtungsberechnung separat ausgeführt. Anschließend werden die beleuchteten optischen Eigenschaften $RGB\alpha_i(x)$ über eine Fusionsfunktion vereint und gerendert.

Color Fusion Die Visualisierungspipeline wird im Falle der *Color Fusion* für alle Volumen separat ausgeführt. Somit wird für jedes Volumen eine eigene Transferfunktion verwendet. Die resultierenden Farb- und Transparenzwerte eines Voxels werden im *Rendering* gewichtet fusioniert (s. Abbildung 3.11). Für die Fusion muss keine explizite Fusionsfunktion angegeben werden, wenn beide Datensätze gleich gewichtet werden. Die *Color Fusion* eignet sich sehr gut für MPP-Visualisierungen, bei denen mehrere Eigenschaften pro Voxel gleichzeitig dargestellt werden. Die *Color Fusion* wird in dieser Arbeit verwendet, um zwei Volumendatensätze, aus denen interessierende Strukturen auf Farben abgebildet wurden, gemeinsam darzustellen. Da nur die optischen Parameter fusioniert werden und keine Unterscheidung anhand der Voxel-eigenschaften mehr möglich ist, können OPP-Visualisierungen mit diesem Verfahren nicht erstellt werden. Änderungen an den Transferfunktionen erfordern einen kompletten Pipeline Durchlauf. Nach einer Verschiebung der Kameraposition oder einer Modifikation der Lichtquellen muss die Beleuchtungsberechnung für beide Volumen erneut ausgeführt werden. Die Modifikation der Registrierung erfordert lediglich eine erneute Beleuchtungsberechnung für den betreffenden Datensatz und die Fusion der Daten. Die Fusion kann sehr effektiv modifiziert werden, da sie den letzten Schritt in der Pipeline darstellt.

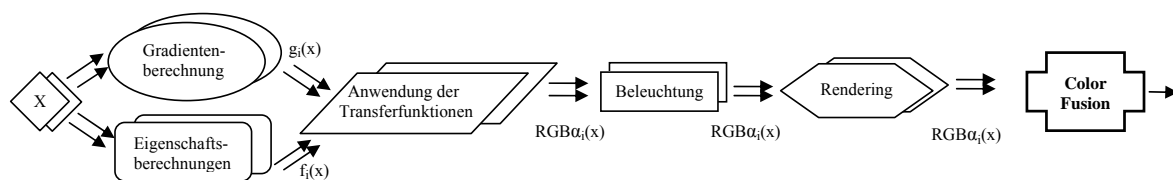


Abbildung 3.11.: Bei der *Color Fusion* wird die gesamte Pipeline für beide Volumina separat ausgeführt. Über zwei Renderer werden die beiden Volumendatensätze dargestellt und zu einer Visualisierung kombiniert.

Bewertung der multimodalen Pipelines Je später die Fusion der Daten in der Visualisierungspipeline durchgeführt wird, umso schneller kann die Visualisierung nach einer Veränderung der Fusionsparameter aktualisiert werden. Auch die Berechnung der Fusion vereinfacht sich meist. Allerdings muss die Pipeline teilweise oder vollständig für mehrere Datensätze parallel ausgeführt werden. Dies erhöht wiederum den Rechenaufwand. Modifikationen an den Visualisierungsparametern sind bei einer späteren Fusion mit einem höheren Zeitaufwand für die Aktualisierung des Renderings verbunden. Der Vorteil einer späteren Fusion liegt darin, dass für die Volumina separate Transferfunktionen definiert werden. Dies kann die Qualität der Visualisierung verbessern (vgl. Abschnitt 6.5). Allerdings erhöht sich so auch der Aufwand für den Nutzer, der diese Transferfunktionen definieren muss. Weiterhin verringert eine späte Fusion in der Visualisierungspipeline den Aufwand für die Aktualisierung nach der Änderung der Registrierungsparameter.

Für die zu entwickelnde Visualisierungskomponente wurden zwei verschiedene Fusionszeitpunkte gewählt. Dabei wurde darauf geachtet, dass in beiden Fällen keine Fusionsfunktion durch den Nutzer definiert werden muss. Weiterhin sollten die Vorteile einer frühen *Property Fusion* und einer späten *Color Fusion* ausgenutzt werden. Je nach Bedarf ist es so möglich, die Visualisierung durch eine *Color Fusion* und der Verwendung separater Transferfunktionen stärker an die Datensätze anzupassen und eine nachträgliche Registrierung zu erleichtern oder durch eine *Property Fusion* die Erzeugung der Visualisierung zu beschleunigen und schneller auf Veränderungen der Visualisierungsparameter zu reagieren.

3.4.4. Eignung der direkten multimodalen Volumenvisualisierung für das Visualisierungssystem

Bei der direkten Volumenvisualisierung werden die Daten direkt an den Renderer übergeben und die gesamte Datenmenge dargestellt. Aus diesem Grund geht keine Information verloren. Voraussetzung dafür ist allerdings, dass die Abbildung der Daten über geeignete Transferfunktionen angepasst an die Fragestellung geschieht. Änderungen der Visualisierungsparameter erfordern keine Veränderungen am Volumen, wodurch die Visualisierung schnell aktualisiert werden kann. Durch die Verwendung von Transparenz ist ein Blick ins Innere des Volumens möglich. Weiterhin können auf diese Weise die Grenzen zwischen Geweben betont werden. Die Verwendung von Farbe für die Klassifikation unterschiedlicher Gewebe erleichtert die Untersuchung des Volumendatensatzes. Um die Voxelinformation

auf Grafikparameter wie Farbe und Transparenz abzubilden, müssen Transferfunktionen definiert werden. Die in dieser Arbeit verwendeten *Spatialized Transfer Functions* sollen den Aufwand dieser Funktionsdefinition verringern (vgl. Abschnitt 4.4 und [RBS05]). Direkte multimodale Volumenvisualisierungen erfordern genaue Überlegungen für die Wahl des Fusionszeitpunktes. Dieser bestimmt maßgeblich den Aufwand für eine Aktualisierung der Visualisierung. Mit der *Property Fusion* und der *Color Fusion* wurden zwei Fusionszeitpunkte gewählt, deren Vorteile der Nutzer im zu entwickelnden Visualisierungssystem ausnutzen kann. Für die multimodale Visualisierung müssen unter Verwendung dieser beiden Fusionszeitpunkte auch keine Fusionsfunktionen definiert werden. Im Falle der *Color Fusion* ist es möglich die Registrierungsparameter auch nach Erzeugung der Visualisierung zu ändern und diese in Echtzeit zu aktualisieren. Nachteilig auf die Verarbeitungs- und Aktualisierungszeit der Visualisierung wirkt sich die große Menge an Daten aus, die an die Grafikhardware übergeben werden. Diese erhöht sich mit jedem Volumen erheblich. Auf die Darstellung der Daten durch den Renderer wird im folgenden Abschnitt eingegangen. Die für die direkte Volumenvisualisierung benötigten Transferfunktionen werden in Kapitel 4 untersucht.

3.5. Rendering

3.5.1. Renderingverfahren

Der Schritt *Rendering* der direkten Visualisierungspipeline umfasst die Darstellung der Volumendatensätze auf dem Betrachtungsmedium (z.B. Monitor) unter Verwendung der, durch die Transferfunktion vergebenen, optischen Eigenschaften wie Farbe und Transparenz. Dies entspricht einer Projektion der 3D-Daten auf eine 2D-Oberfläche. Die optischen Eigenschaften von Volumenelementen, die auf den gleichen Bildpunkt abgebildet werden, müssen dafür geeignet miteinander kombiniert werden. Die Überlagerung transparenter Voxel wird als Komposition oder Blending bezeichnet (vgl. [Ang00] und [RS01]). Für diese Arbeit wurde die in der medizinischen Visualisierung am häufigsten verwendete parallele Projektion der Datensätze einer perspektivischen Darstellung vorgezogen. Der Vorteil der parallelen Projektion ist, dass parallele Projektionsstrahlen mit gleicher Abtastdichte über die Schichten des Datensatzes verwendet werden können.

Eine grundlegende Einteilung der Renderingalgorithmen erfolgt in bild-, objekt- und texturbasiert (vgl. [Ang00], [FvDFH90] und [RS01]). In bildbasierten Ansätzen wird für jeden Bildpunkt der darzustellende Farbwert anhand der optischen Eigenschaften der betreffenden Voxel ermittelt. Objektbasierte Algorithmen gehen von den darzustellenden Volumenelementen aus und bestimmen deren Anteil am entgültigen Farbwert. Im texturbasierten Ansatz werden die Volumendaten als 3D-Texturen von der Grafikhardware dargestellt. Weiterhin existieren zwei Möglichkeiten die Daten zu verarbeiten. In *front to back* Ansätzen werden die Volumenelemente, die dem Betrachter oder auch der Kamera am nächsten sind, zuerst verarbeitet. *Back to front* Ansätze beginnen hingegen mit den am weitesten entfernten Voxeln.

Ray Casting

Der am häufigsten verwendete Renderingalgorithmus ist das *Ray Casting*. In diesem bildbasierten Ansatz werden Strahlen ausgehend von der Kamera durch die einzelnen Bildpunkte des Monitors gesendet und durch das Volumen verfolgt. Die Intensitäts- und Gradientenwerte der Volumenelemente, die auf dem Strahl liegen, werden durch die TFs auf optische Eigenschaften abgebildet, welche zur Berechnung des Pixelfarbwertes herangezogen werden (s. Abbildung 3.12).

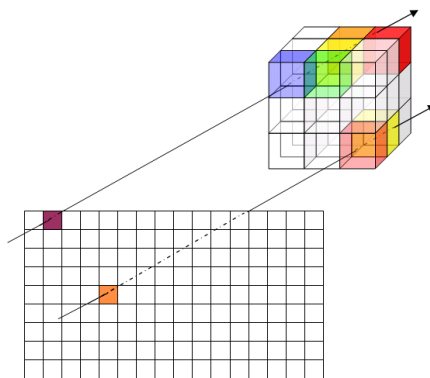


Abbildung 3.12.: Beim *Ray Casting* wird für jeden Bildpunkt ein Strahl durch das Volumen gesendet und die Farb- und Transparenzwerte der geschnittenen Voxel werden zum Farbwert des Bildpixels kumuliert.

In der Theorie entspricht dies einer Integration der Absorptions- und Emissionswerte der Volumenelemente auf dem Strahl (vgl. [RS01]). In der praktischen Umsetzung wird eine Abtastrate entlang des Strahls festgelegt und die optischen Eigenschaften an den Abtastpunkten bestimmt. Die Abtastrate sollte für die Reduktion von *Aliasing*-Artefakten unterhalb der Voxelgröße liegen. Für die Berechnung der Abtastwerte können zum einen die optischen Eigenschaften des nächstgelegenen Voxels verwendet werden. Eine andere Möglichkeit ist durch trilineare Interpolation der acht umgebenden Voxel-eigenschaften eine neue Eigenschaft zu berechnen. Die Diskretisierung und die daraus resultierende Berechnung des Farb- und Transparenzwertes über Summen und Produkte ist eine ausreichende Annäherung an das Integral über alle Punkte auf dem Strahl. Die Berechnung des Transparenzwertes α und des Farbwertes $C \in \{R, G, B\}$ eines Bildpunktes lässt sich für den *front to back* Ansatz wie folgt rekursiv umsetzen:

$$\alpha_i = \alpha_{i-1} + (1 - \alpha_{i-1}) * \alpha_i \quad (3.1)$$

$$C_i = C_{i-1} + (1 - \alpha_{i-1}) * \alpha_i * C_i \quad (3.2)$$

Die Komposition der Voxel erfolgt meist *front to back*, da dieser Ansatz die Möglichkeit bietet den Strahl nicht durch das gesamte Volumen verfolgen zu müssen. Als Abbruchkriterium wird die kumulierte Transparenz (auch Opazität) herangezogen. Die Begriffe Opazität und Transparenz sind Antonyme. Der Transparenzwert α ist damit gleich $(1 - \text{Opazitätswert})$. So kann die Strahlverfolgung abgebrochen werden, wenn die Opazität 95% erreicht hat, da Voxel die anschließend gerendert werden kaum noch zum Farbwert des Bildpunktes beitragen. Diese Optimierung des *Ray Casting* wird als *Early*

Ray Termination bezeichnet und kann die für das Rendering benötigte Zeit halbieren (vgl. [Lev90]).

Das Rendering mit Hilfe des *Ray Casting* Algorithmus muss nicht durch den Nutzer parametrisiert werden, da die Abtastrate fest vorgegeben werden kann. Nach der Modifikation der Kameraparameter ist ein erneutes Rendering des Datensatzes nötig. Um die Darstellung zu beschleunigen können die Abtastrate und die Strahlendichte stark reduziert werden. Die Datensätze müssen für die Darstellung mit dem *Ray Casting* Algorithmus nicht speziell vorbereitet werden. Aus diesem Grund wurde der *Ray Casting* Ansatz in das Visualisierungssystem integriert. *Ray Casting* bietet sehr gute Visualisierungsergebnisse, ist aber auch das langsamste Renderingverfahren. Aus diesem Grund sollen im Folgenden einige Alternativen betrachtet werden.

3D-Texture Mapping

3D-Texture Mapping Ansätze nutzen effektiv die Unterstützung moderner Grafikkarte für die Darstellung der Volumendatensätze aus. Hierzu wird das Volumen (gegebenenfalls mit Gradienteninformation) in einen 3D-Texturspeicher auf der Grafikkarte geladen. Dieser muss ausreichend groß sein oder die Textur muss in mehrere kleinere Teilstrukturen aufgeteilt werden (*Bricking*). Zu diesem Volumen kann auch eine *2D-Lookup Table* als 2D-Textur für die Abbildung auf Farb- und Transparenzwerte gespeichert werden. Die Darstellung erfolgt, indem in einem bestimmten Abstand Schnittebenen durch das 3D-Texturvolumen gelegt werden. Die Schnittebenen verlaufen parallel zur Bildebene, auf die das dreidimensionale Bild projiziert wird. Auf die Schnittebenen werden die betreffenden Teile der 3D-Textur durch die Hardware aufgetragen (*Mapping*), wodurch texturierte Polygone entstehen. Die Texturierung der Schnittebenen erfolgt durch trilineare Interpolation. Die Polygone werden anschließend von hinten nach vorn (*back to front*) zusammengesetzt (*Blending*). Während dieser Überblendung werden die Farb- und Transparenzwerte anhand der Eigenschaften der Textur Elemente (Texel) aus der *2D-Lookup Table* herausgelesen und für die Berechnung des Pixelwertes verwendet. Auch dies ist in modernen Grafikkarten effektiv in Hardware umgesetzt.

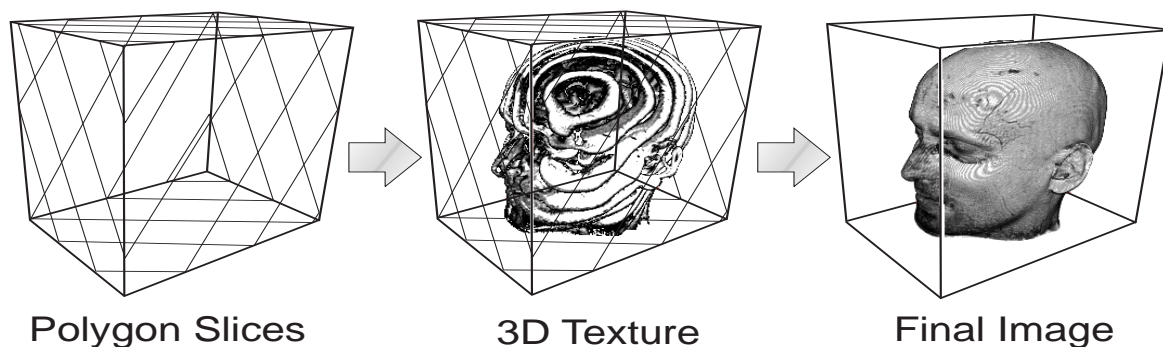


Abbildung 3.13.: Auf die Schnittebenen, welche parallel zur Bildebene liegen, werden beim *3D-Texture Mapping* die Datenwerte als Textur gelegt und diese Polygone anschließend durch ein *Blending* zur 3D-Visualisierung zusammgefügt. Quelle: [RS01].

Die Qualität des 3D-*Texture Mapping* reicht an die des *Ray Casting* heran, wenn die Schnittebenen dicht genug platziert werden. Die Geschwindigkeit des Rendering durch 3D-*Texture Mapping* ist durch die Hardwareunterstützung schneller als beim *Ray Casting* wodurch Aktualisierungen in Echtzeit möglich sind (vgl. [GK96]). Der große Nachteil des 3D-*Texture Mapping* sind die hohen Hardwareanforderungen und der enorme Speicherbedarf. Gegebenenfalls müssen große Datensätze in sogenannte *Bricks* unterteilt und diese getrennt verarbeitet werden. Das 3D-*Texture Mapping* konnte nicht in das Visualisierungssystem integriert werden, da es in der zur Verfügung stehenden VTK Version 4.4 noch nicht implementiert war.

Weitere Rendering Ansätze

Neben dem bildbasierten *Ray Casting* und dem texturbasierten *Texture Mapping* werden in der Literatur objektbasierte Ansätze für das Rendering verwendet. Beim *Splattting* z.B. wird für jedes Voxel des Datensatzes dessen Abdruck (engl. *Footprint*) auf dem Bild berechnet. Der Abdruck ergibt sich aus der Faltung des Voxelwertes mit einem 2D-Gaußfilter, dessen Filtermaske an die Größe des Datensatzes und an das zu erzeugende Bild angepasst werden muss. Die Voxel des Datensatzes werden senkrecht zur Sichtrichtung durchlaufen und der Datensatz auf diese Art schichtweise verarbeitet. Dabei kann entweder *front to back* oder *back to front* gezeichnet werden. Die Abdrücke der Voxel werden für die Berechnung des Datenwertes eines Bildpixels unter Beachtung der vergebenen Transparenz- und Farbwerte aufsummiert (vgl. [Wes90]). Der *Splattting* Ansatz wird in dieser Arbeit nicht verfolgt, da er nicht im VTK implementiert ist.

Der *Shear Warp* Ansatz stellt eine hybride Form des Rendering dar. Zunächst wird das 3D-Objekt geschert, indem die Schichten zueinander verschoben werden. Anschließend werden die Daten wie beim *Ray Casting* auf eine Bildebene im neu entstandenen Objektraum gerendert, wobei bilinear interpoliert wird (s. Abbildung 3.14). Es werden demnach keine Zwischenschichten interpoliert. Beim *Shear Warp* Ansatz muss das Volumen daher dreimal im Speicher abgelegt werden, um für jede der drei Blickrichtungen eine Repräsentation vorliegen zu haben. Je nach Blickwinkel auf den Datensatz, wird die Repräsentation mit der geringsten Abweichung gezeichnet. Im Gegensatz zum *Ray Casting* verlaufen die Strahlen beim *Shear Warp* durch die Scherung immer senkrecht zu den Schichten des Datensatzes. Dies vereinfacht die Komposition der Volumenelemente (vgl. [LL94]). Um die Abtastung des Datensatzes zu beschleunigen, wird eine Lauflängenkodierung der Daten vorgenommen, die es ermöglicht transparente Voxel zu überspringen. Das verzerrte Zwischenbild im Objektraum muss anschließend in den Bildraum überführt werden (*Warping*), wobei bilinear interpoliert wird. Die zweifache Interpolation ist Ursache für die Fehleranfälligkeit des Ansatzes. Weiterhin ist der *Shear Warp* Ansatz nicht im VTK umgesetzt und wurde daher nicht in das Visualisierungssystem integriert.

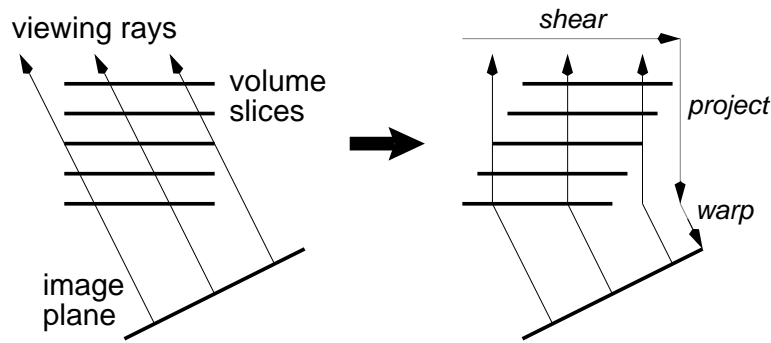


Abbildung 3.14.: Durch die Scherung der Schichten wird es möglich beim *Shear Warp* mit parallelen Strahlen zu arbeiten, welche senkrecht zu den Schichten des Datensatzes stehen. Das in den gescherten Objektraum projizierte Zwischenbild wird anschließend durch ein sogenanntes *Warping* in den Bildraum überführt. Quelle: [LL94]

3.5.2. Beleuchtungs- und Schattenberechnung

Um die Volumenvisualisierung dreidimensional erscheinen zu lassen, müssen während des Renderings eine Beleuchtungs- und eine Schattenberechnung (engl. *Shading*) durchgeführt werden. Die Beleuchtungsberechnung ermittelt die Intensität des Lichts, welches am Abtastpunkt reflektiert wird, unter Beachtung von Position, Anzahl und den Eigenschaften der verwendeten Lichtquellen (vgl. [RS01], [FvDFH90]). Das am häufigsten verwendete Beleuchtungsmodell wurde von Phong [Pho75] vorgestellt. Phongs Beleuchtungsmodell verwendet vier Vektoren um die Beleuchtung eines Punktes p zu bestimmen. Neben den Vektoren vom Punkt zur Kamera v , vom Punkt zur Lichtquelle l und der Reflektionsrichtung r muss noch die Normalenrichtung n des Punktes bekannt sein. Die Normalen der Punkte können über die Berechnung eines Grauwertgradienten angenähert werden (vgl. [HB]). Die Berechnung der Grauwertgradienten erfolgt über zentrale Differenzen (vgl. Abschnitt 4.3.3 und Gleichung 4.6). Weiterhin unterscheidet Phong zwischen drei Arten des ausfallenden Lichts. Die *ambient* Beleuchtung repräsentiert das Licht, welches nicht direkt von einer Lichtquelle ausgeht sondern z.B. aus der Lichtreflektion umgebender Objekte resultiert. Die *diffuse* Beleuchtung beschreibt das Licht, welches von matten Oberflächen gestreut wird. Die *spekulare* Beleuchtung beschreibt die Eigenschaft des Objektes das Licht fast vollständig in eine Richtung zu reflektieren.

Die Berechnung der Schattierung bestimmt, wie das Beleuchtungsmodell auf das Objekt angewendet wird (vgl. [RS01], [FvDFH90]) und ist vor allem von der Art der verwendeten Interpolation abhängig (vgl. [Ang00]). Anhand der Information aus der Beleuchtungsberechnung und den Eigenschaften des Punktes, wie Transparenz und Reflexion des einfallenden Lichts wird ein Schatten berechnet und der Farbwert des Punktes mit diesem modifiziert. Der schattierte Farbwert wird anschließend für die Komposition im Rendering verwendet.

3.5.3. Eignung der Renderingverfahren für multimodale Daten

Angelehnt an [FPT04] werden in diesem Abschnitt die Renderingverfahren auf ihre Eignung für die Erstellung multimodaler Volumenvisualisierungen untersucht.

Ausrichtung der Datensätze

Ray Casting bietet als einziges Renderingverfahren die Möglichkeit die Transformation der Registrierung erst im *Rendering* Schritt der Visualisierungspipeline anzuwenden und Datensätze mit unterschiedlicher Orientierung und Auflösung zu verarbeiten. Jeder darzustellende Datensatz besitzt in diesem Fall sein eigenes lokales Koordinatensystem. Die Abtastpunkte auf dem Strahl werden in das jeweilige lokale Koordinatensystem überführt und der Wert aus den umgebenden Voxeln interpoliert. Hierbei muss die Voxelgröße des Datensatzes beachtet werden, da ansonsten die berechneten Messwerte über unterschiedliche große Regionen in den Datensätzen interpoliert werden. Weiterhin kann der *Ray Casting* Ansatz auch Datensätze verarbeiten, die bereits zueinander ausgerichtet sind.

Für den objektbasierten Ansatz *Splatting* und das hybride *Shear Warp* ist eine Ausrichtung der Datensätze und die Angleichung der Auflösung notwendig, da aus den unterschiedlichen Schichtdicken ein Fehler bei der Tiefenkomposition der Voxelwerte resultiert. Dies verursacht eine unterschiedliche Tiefendarstellung der Datensätze. *Texture Mapping* Ansätze vermeiden diesen Fehler, da neue Schnittebenen durch den Datensatz gelegt werden. Neben der Tiefenkomposition tritt auch hier ein Fehler bei der Komposition der Datenwerte innerhalb einer Schicht bzw. Schnittebene auf, den alle drei Ansätze teilen. Aus diesem Grund können die drei Ansätze nicht direkt auf nicht ausgerichtete Datensätze angewendet werden.

Eignung für mehrere Volumendatensätze

Durch den enormen Speicherbedarf auf der Grafikhardware müssen die Datensätze bei der Verwendung des *Texture Mapping* Ansatzes für Multivolumen-Visualisierungen in kleinere und damit mehr *Bricks* unterteilt werden, damit sich die Daten aller darzustellenden Volumen gleichzeitig im Speicher befinden. Dies verlangsamt das *Texture Mapping*, da häufiger Daten an die Grafikkarte übergeben werden müssen.

Der *Splatting* Ansatz kann ohne weiteres für multimodale Visualisierungen ausgerichtet Datensätze verwendet werden. Die Daten müssen dabei für alle Fusions Pipelines bis auf die *Color Fusion* vor dem Splatting auf Voxel Ebene fusioniert werden. Die *Color Fusion* kann nach dem Splatting durch eine Komposition der berechneten Projektionen umgesetzt werden.

Die Anwendung des *Shear Warp* Algorithmus auf Multivolumendaten ist am kompliziertesten, da sich der benötigte Speicherbedarf für die drei Sichten auf das Volumen bei zwei darzustellenden Datensätzen verdoppelt. Weiterhin muss die Lauflängenkodierung an beide Datensätze angepasst werden, wodurch die Effizienz des Ansatzes verringert wird, da weniger Voxel übersprungen werden können.

Eignung für die gewählten Fusionszeitpunkte

Der *Ray Casting* Ansatz eignet sich sowohl für *Property Fusion* als auch *Color Fusion*. Allerdings können mit der *Property Fusion* bessere Ergebnisse erzielt werden, da in diesem Fall die trilineare Interpolation der Datenwerte für den einen Datensatz ausgeführt werden kann und nicht die beleuchteten Farbwerte zweier Datensätze durch *Nearest Neighbour* Interpolation komponiert werden müssen.

Für die *Color Fusion* bietet sich der *Texture Mapping* Ansatz an, da die Daten bereits beleuchtet vorliegen. Es existieren zwar Ansätze um Beleuchtungsberechnungen in monomodale Visualisierungen mittels *Texture Mapping* einzubauen (vgl. [MHS99]), allerdings wurden diese bisher noch nicht auf die Anforderungen multimodaler Visualisierungen angepasst.

Splatting und *Shear Warp* eignen sich am ehesten für *Color Fusion* mit Datensätzen, die zuvor bereits beleuchtet wurden, da sie nur bilinear und somit ausschließlich innerhalb der Schichten des Datensatzes interpolieren.

4. Transferfunktionen

Methods for Transfer Function construction are too complex and the tissue separation abilities of the Transfer Function need to be extended.

LUNDSTRÖM ET AL., AUS [LLY05]

Transferfunktionen (TFs) sind das wichtigste Mittel für die Erstellung einer aussagekräftigen direkten Volumenvisualisierung. Sie bilden die Originalwerte eines Datensatzes auf optische Parameter ab. So ermöglichen sie den Einblick in das Innere des Volumens durch die Vergabe von Transparenz. Um interessante Bereiche von der Umgebung abzuheben, bietet sich die Zuordnung von Farbe anhand der gemessenen Grauwerte an. TFs bestimmen somit maßgeblich die Aussagekraft eines direkten Volumenrenderings. Die Definition einer geeigneten Funktion ist ein schwieriger und oft aufwändiger Prozess. Dieses Kapitel beschäftigt sich mit den Anforderungen, welche an die zu verwendenden TFs gestellt werden. Neben den Arten von TFs werden grundlegende Möglichkeiten für die Spezifikation von TFs vorgestellt.

4.1. Anforderungen an die verwendeten Transferfunktionen

Eine Vorgabe für diese Arbeit war die Verwendung von TFs für die Extraktion einzelner, interessierender Strukturen aus einem Datensatz und deren Abbildung auf Farbe und Transparenz. Dabei soll die gesamte Struktur sichtbar sein, weshalb die direkte Volumenvisualisierung gewählt wurde. Die Grenzen der Struktur sind hervorzuheben. Es ist gewünscht, dass weitere Strukturen, auch aus einem anderen Datensatz, zur Visualisierung hinzugefügt werden können. Die verwendeten TFs müssen für die Darstellung der drei, in Kapitel 2 vorgestellten, Modalitäten geeignet sein. Für eine bessere Separierbarkeit von homogenen Regionen innerhalb eines oder mehrerer Datensätze, soll neben den Messwerten der Datensätze die Gradienteninformation als Parameter für die Spezifikation der TF genutzt werden. Dies macht die Definition einer TF für jeden Datensatz notwendig. Weiterhin soll es möglich sein, eine TF zu definieren, die auf zwei Datensätze angewendet werden kann und deren Intensitätswerte als Parameter nutzt. Der Aufwand für die Definition der TF und den nötigen Veränderungen ist für den Nutzer so gering

wie möglich zu halten. Allerdings soll der Nutzer dennoch die TF und damit die Visualisierung an seine Fragestellung anpassen können. Veränderungen an der TF müssen möglichst einfach durchführbar sein und die Visualisierung danach schnell aktualisiert werden. Bei Veränderungen an der TF soll dem Nutzer ein Feedback gegeben werden, so dass er seine Einstellungen anhand des Ergebnisses überprüfen und ggf. rückgängig machen kann. Die Reduktion des Aufwands für den Nutzer spielt vor allem im Hinblick auf kaum zu standardisierende Aufnahmen wie MRT- oder PET-Daten eine Rolle, da für diese keine Vorgaben von TFs gemacht werden können und für jeden Datensatz eine eigenen TF spezifiziert werden muss.

4.2. Eindimensionale Transferfunktionen

Eindimensionale (1D) Transferfunktionen verwenden nur eine Eigenschaft des Datensatzes für die Zuordnung optischer Parameter. Ein typisches Beispiel ist die Vergabe von Transparenz durch eine Transparenz-Transferfunktion (engl. *Opacity Transfer Function*, Abk. TTF) anhand der berechneten Gradientenstärke (Betrag des Gradientenvektors).

$$\text{Transparenz-Transferfunktion} : \text{Gradientenstärke} \rightarrow \text{Transparenz } \alpha \quad (4.1)$$

Durch eine solche Funktion ist es möglich Daten, die sich in der Nähe einer Kante befinden undurchsichtig zu machen und so die Kante hervorzuheben. In [HRH03] werden 1D-TTFs über die Skalarwerte (Intensität) des Datensatzes wie folgt definiert:

$$\text{Transparenz-Transferfunktion} : \text{Skalarwert} \rightarrow \text{Transparenz } \alpha \quad (4.2)$$

Eine solche TTF wird auch als globale TTF bezeichnet, da sie jedem Voxel im Datensatz unabhängig von seiner Umgebung eine Transparenz zuweist. Analog dazu kann die Vergabe von Farbwerten anhand des gemessenen Skalarwertes (z.B. Hounsfield Unit im CT) im Bild erfolgen.

$$\text{Farb-Transferfunktion} : \text{Skalarwert} \rightarrow \text{RGB} \quad (4.3)$$

Die Vergabe von Farbe ermöglicht es dem Betrachter Objekte voneinander zu unterscheiden. Die Trennung von Objekten, die sich im Grauwertbereich überschneiden, ist durch eine 1D-Transferfunktion jedoch nicht möglich. Aus diesem Grund wurden von Marc Levoy 1988 mehrdimensionale TFs eingeführt (vgl. [Lev88]). Durch die Erweiterung des Merkmalsraums war es möglich, Gewebe besser zu unterscheiden. Dies begründet auch die Anforderung, dass die zu verwendenden TFs über zwei Parameter definiert sein sollen.

4.3. Gradientenbasierte mehrdimensionale Transferfunktionen

Mehrere Eigenschaften eines Datensatzes für die Vergabe optischer Parameter wurden erstmals von Marc Levoy in [Lev88] verwendet. Darin beschäftigte er sich mit der Zuordnung von Transparenz, um Kanten zwischen benachbarten Geweben in medizinischen Daten hervorzuheben. Die von ihm beschriebene *Opacity Transfer Function* verwendet als Eingabeparameter den Skalarwert des Datensatzes und die berechnete Gradientenstärke, um daraus den Transparenzwert α zu ermitteln. Seine Funktion sieht damit wie folgt aus:

$$\text{Transparenz-Transferfunktion} : \text{Skalarwert} \times \text{Gradientenstärke} \rightarrow \text{Transparenz } \alpha \quad (4.4)$$

Indem Levoy sich auf CT-Datensätze beschränkte, war es ihm möglich, drei Annahmen bezüglich der zugrunde liegenden Daten zu treffen. So nahm er an, dass die im Volumen vorhandenen Gewebe jeweils auf kleine Bereiche um einen dazugehörigen bekannten Grauwert (HU) abgebildet werden. Weiterhin legte er fest, dass ein Gewebe höchstens zu zwei anderen Geweben benachbart ist. Die letzte Annahme schränkt die Nachbarschaft noch weiter ein. Sie besagt, dass wenn die Gewebe nach dem Grauwert sortiert vorliegen, jedes Gewebe nur Gewebe berührt, die auch in der Sortierung benachbart sind. Mit diesen Annahmen war es ihm möglich, eine 2D-Transferfunktion für die Transparenz zu definieren. Abbildung 4.1 zeigt diese Funktion.

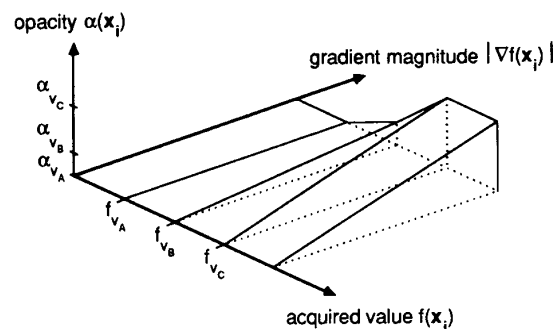


Abbildung 4.1.: Dreidimensionaler Plot von Levoy's 2D-Transparenz-Transferfunktion. Die Achse mit den Datenwerten des Bildes zeigt nach rechts unten. Auf der Achse die nach rechts oben zeigt ist die Gradientenstärke abgetragen. Die senkrechte Achse trägt die Transparenz α für die gegebene Wertekombination ab. Ist $\alpha = 0$, so wird das betreffende Voxel durchsichtig und bei $\alpha = 1$ opak dargestellt. In dieses Koordinatensystem wurden drei bekannte Gewebewerte f_{v_A} , f_{v_B} und f_{v_C} mit der dazugehörigen Transparenz eingetragen. Für Datenwerte, die zwischen den bekannten Geweben liegen, wird der Transparenzwert α linear interpoliert. Quelle: [Kin99].

Die Gewebewerte aus der CT-Aufnahme werden mit f_{v_i} , $i = 1 \dots n$ bezeichnet, wobei jedes i der Werteverteilung eines der n Gewebe im Volumen entspricht. Jedem Gewebe v_i soll über die Skalarwerte eine Transparenz α_i zugewiesen werden, um die einzelnen

Gewebe in der Visualisierung unterscheiden zu können. Für Skalarwerte, die zwischen zwei Gewebe v_i und v_{i+1} fallen, wird die Transparenz zwischen den dazugehörigen Transparenzwerten α_i und α_{i+1} linear interpoliert. Als zweiter Parameter beeinflusst die Gradientenstärke die Transparenz, indem sie den Transparenzwert α_i skaliert. Je höher der Gradientenbetrag, desto undurchsichtiger erfolgt die Darstellung der Voxel. Dadurch werden homogene Bereiche durchsichtig und somit ausgeblendet sowie Kanten in der Visualisierung hervorgehoben.

Die Visualisierung durch gradientenbasierte TFs verfolgt das Ziel, die Kanteninformation mit in den Visualisierungsprozess einzubeziehen. Diese kann dabei für die Bestimmung der Transparenz oder für die Farbvergabe herangezogen werden. Die Gradienteninformation soll dazu verwendet werden interessierende homogene Regionen voneinander und vom Rest des Volumens zu unterscheiden und die Grenzen zwischen den Regionen hervorzuheben. Dazu können auch ein- und mehrdimensionale TFs gleichzeitig verwendet werden, wie z.B. eine Kombination aus 1D-Transparenz- und 2D-Farb-Transferfunktion. Die Grundlage für gradientenbasierte TFs legten Kindlman et al. [KD98], welche die Idee von Marc Levoy erweiterten. Sie verwenden die erste und zweite Ableitung der gemessenen Datenwerte für eine halbautomatische Definition von TTFs. Dieser Abschnitt soll das Konzept der verwendeten gradientenabhängigen TFs näher beleuchten. Dazu wird zunächst das zugrunde liegende Kantenmodell erläutert und anschließend die Berechnung der Gradienteninformation und das daraus resultierende 2D-Histogramm beschrieben. Die folgenden Betrachtungen werden für den eindimensionalen Fall durchgeführt, können aber ohne Beschränkung der Allgemeingültigkeit auch auf den zwei- und dreidimensionalen Fall ausgeweitet werden.

4.3.1. Verwendetes Kantenmodell

Diese Arbeit stützt sich auf das von Kindlmann et al. in [KD98] vorgeschlagene Modell einer idealen Kante. Das Modell geht davon aus, dass sich die gemessenen Eigenschaften im Datensatz an einer Kante stark ändern. Diese Diskontinuität wird allerdings durch die in Abschnitt 2.5.1 beschriebene Bandpassfilterung der Daten verwischt. Die Veränderung der Kanteninformation kann durch einen Gaußfilter modelliert werden. Die Grafik 4.2 zeigt das Modell der idealen Kante als Stufenfunktion und das Resultat nach der Faltung mit der Gaußfunktion.

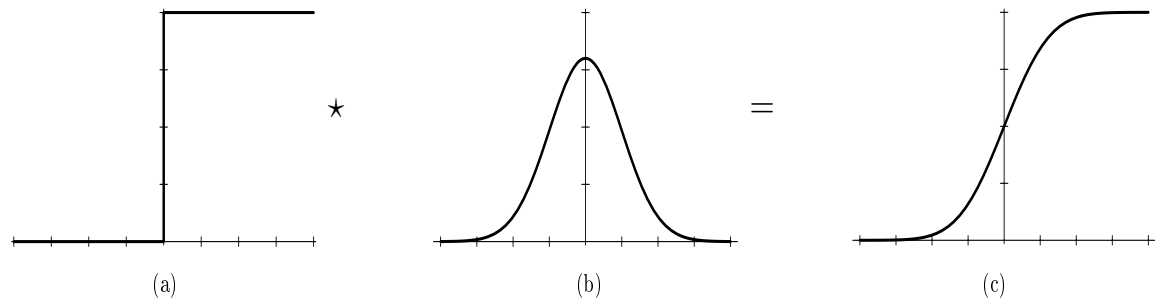


Abbildung 4.2.: Im idealen Fall ist der Übergang zwischen zwei Geweben eine Stufenfunktion mit nur 2 Grauwerten wie in (a) dargestellt. Die Abzisse trägt in den drei Darstellungen die Position im Bild ab, während die Ordinate die Grauwerte zeigt. Durch die Bandpassfilterung bei der Bildrekonstruktion wird die Kante allerdings geglättet. Dies kann durch einen Gaußfilter (b) angenähert werden. Die Kante im Volumen erstreckt sich demnach über einen Grauwertbereich und einen Teil des Bildes, wie in (c) dargestellt. Quelle: [Kin99].

4.3.2. Definition der Gradienteninformation

Kindlmann et al. [KD98] beschreiben die Gradienteninformation als erste und zweite Ableitung ($D_{\widehat{\nabla}f}f$ und $D_{\widehat{\nabla}f}^2f$) der Bildfunktion $f(x)$ in Richtung des Gradienten $\widehat{\nabla}f$. Sie nutzen dabei die Tatsache aus, dass Gradienten immer senkrecht zu einer Kante stehen. Somit ist die Richtung des Gradienten abhängig von der Position x an der die Ableitungen berechnet werden. Dies wird in Abbildung 4.3 verdeutlicht. Die erste und

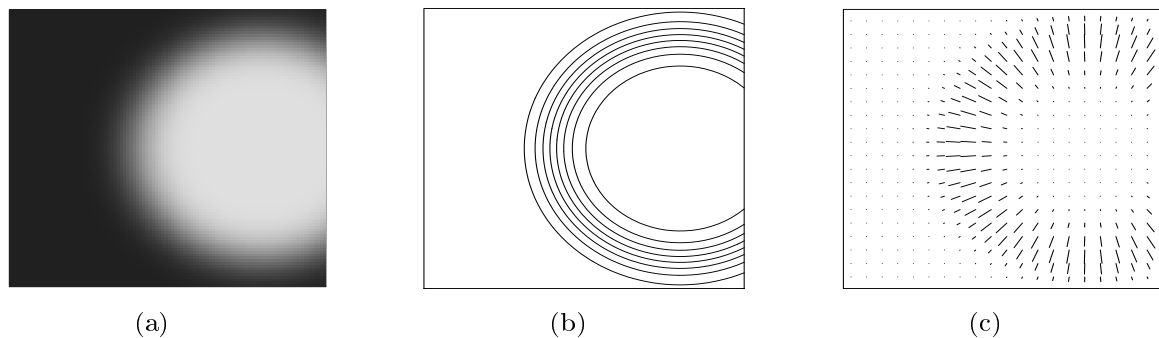


Abbildung 4.3.: Für den bandpassgefilterten Übergang zwischen zwei Geweben (a) können über die Messwerte $f(x)$ Isolinien (b) berechnet werden. Diese verlaufen durch die Bildpunkte, die den gleichen Intensitätswert besitzen. In (c) wurde die Gradienteninformation für einige Bildpunkte berechnet. Die Längen der Linien zeigen die Stärke des Gradienten an der jeweiligen Position und die Ausrichtungen zeigen die Richtung des stärksten Gefälles an. Es ist ersichtlich, dass die Gradientenrichtung $\widehat{\nabla}f$ immer senkrecht zu der Isolinie steht, die durch den jeweiligen Punkt verläuft. Quelle: [Kin99].

zweite Ableitung werden im Folgenden abgekürzt mit $f'(x)$ und $f''(x)$ bezeichnet. Das Diagramm in Abbildung 4.4 zeigt den Verlauf der Messwerte $f(x)$ sowie der ersten und zweiten Ableitung ($f'(x)$ und $f''(x)$) bei der Abtastung eines Bereiches senkrecht zu einer Kante zwischen zwei Objekten im eindimensionalen Fall.

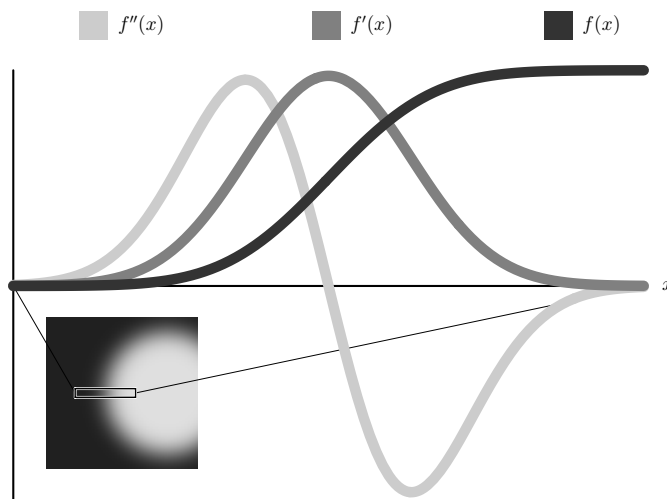


Abbildung 4.4.: Abtastung innerhalb einer Bildzeile senkrecht zu einer Bandpassgefilterten idealen Kante. Am Ort der Kante hat $f'(x)$ sein Maximum und $f''(x)$ beschreibt einen Nulldurchgang. Quelle: [Kin99].

Wie bereits beschrieben, verwischt die Bandpassfilterung die ideale Kante, wodurch neue Grauwerte im Bild entstehen. Hierdurch ergibt sich die abgebildete Funktion $f(x)$ aus der Stufenfunktion. In der Mitte der Grenze zwischen den Geweben hat $f'(x)$ ihr Maximum und $f''(x)$ beschreibt einen Nulldurchgang. Dies ermöglicht die genaue Lokalisation der Kante, auch wenn diese durch die Bandpassfilterung verschmiert wurde (vgl. [KD98]). Da die TF unabhängig von der Position auf das gesamte Volumen angewendet werden soll, ist es nötig eine Repräsentation der Gradienteninformation zu finden, welche ohne die räumliche Information auskommt. Da sowohl die Werte der bandpassgefilterten Bildfunktion $f(x)$ als auch $f'(x)$ vom Ort x entlang der Kante abhängig sind, können sie in einem 3D-Koordinatensystem mit den Achsen x , $f(x)$ und $f'(x)$ abgetragen werden. Abbildung 4.5 zeigt diesen Plot für den Verlauf der Funktionswerte entlang einer bandpassgefilterten idealen Kante.

Wird dieser 3D-Plot parallel zur x -Achse projiziert, so entsteht die gesuchte Beziehung zwischen $f(x)$ und $f'(x)$. Die Eliminierung der Positionsinformation x ist möglich, da sich die Messwerte $f(x)$ bei der Abtastung senkrecht zur Kante monoton erhöhen und somit eine nicht-lineare eineindeutige Beziehung zwischen der Position x und dem Messwert $f(x)$ existiert. Somit kann $f'(x)$ auch als Funktion des Messwertes $f(x)$ dargestellt werden. Kindlmann et al. zeigen dies in [KD98] ebenso für die zweite Ableitung. Aufgrund der Anforderung für die Visualisierung eine 2D-Transferfunktion basierend auf Intensität und Gradientenbetrag zu verwenden, wird die zweite Ableitung nicht benötigt und hier nicht weiter untersucht.

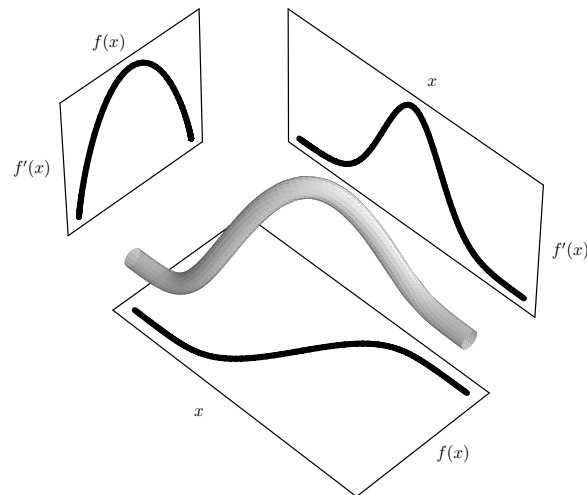


Abbildung 4.5.: Werden für den Messwert sowie die erste und zweite Ableitung jeweils eine Koordinatenachse verwendet und in dieses Koordinatensystem die Messwerte bei der Abtastung senkrecht zur Kante abgetragen, resultiert daraus ein dreidimensionales Plot der Kanteninformation. Quelle: [Kin99].

4.3.3. Histogramme über Intensität und Gradientenstärke

Die im vorhergehenden Abschnitt beschriebene Relation zwischen den Messwerten $f(x)$ und der ersten Ableitung $f'(x)$ wird in einem Histogramm über Intensität $f(x)$ und dem Betrag des Gradienten, also dem Betrag der ersten Ableitung $|f'(x)|$ (Gradientenstärke, engl. *Gradient Magnitude*), sichtbar gemacht. Das Histogramm wird durch ein regelmäßiges Gitter repräsentiert. Auf der x-Achse des Histogramms sind die im Volumen vorkommenden Messwerte $f(x)$ abgetragen. Die y-Achse repräsentiert die Gradientenstärke $|f'(x)|$. Die Elemente des 2D-Histogramms werden Bins genannt und repräsentieren einen kleinen Teil des Wertebereiches der abgetragenen Größen. Der Wertebereich den ein Bin einschließt, ergibt sich aus folgender Formel:

$$\begin{aligned} \text{BinGrößeIntensität} &= \text{MaximumIntensität} / \text{AnzahlBinsIntensität} \\ \text{BinGrößeGradSt} &= \text{MaximumGradSt} / \text{AnzahlBinsGradSt} \end{aligned} \quad (4.5)$$

In der Literatur wird meist mit einer festen Anzahl von Bins (z.B. 256 in jede Dimension) gearbeitet. Dies ermöglicht es, die Histogramme mehrerer Datensätze miteinander zu vergleichen und so mit vordefinierten TFs (engl. *Presets*) zu arbeiten (vgl. [HST+04]). Die Anzahl der Bins kann für die beiden Dimensionen auch unterschiedlich gewählt werden. Um Artefakte in Form von leeren Bins im Histogramm zu vermeiden, kann die Anzahl der Bins an die im Datensatz vorkommenden Werte angepasst werden. Allerdings können so keine vordefinierten TFs mehr verwendet werden. Auf die Auswirkungen leerer Bins auf die Qualität des Histogramms und Lösungsvorschläge wird im Abschnitt 5.3.3 näher eingegangen.

Ein zweidimensionales Intensitäts-/ Gradientenstärke-Histogramm repräsentiert die im vorhergehenden Abschnitt gezeigte Beziehung zwischen den Datenwerten $f(x)$ und der Stärke des Gradienten $|f'(x)|$, da die Abtastung eines Bildes oder Volumendatensatzes an diskreten Positionen eine Annäherung des Histogramms an den oben gezeigten Plot ergibt. Dies wird aus Abbildung 4.6 ersichtlich.

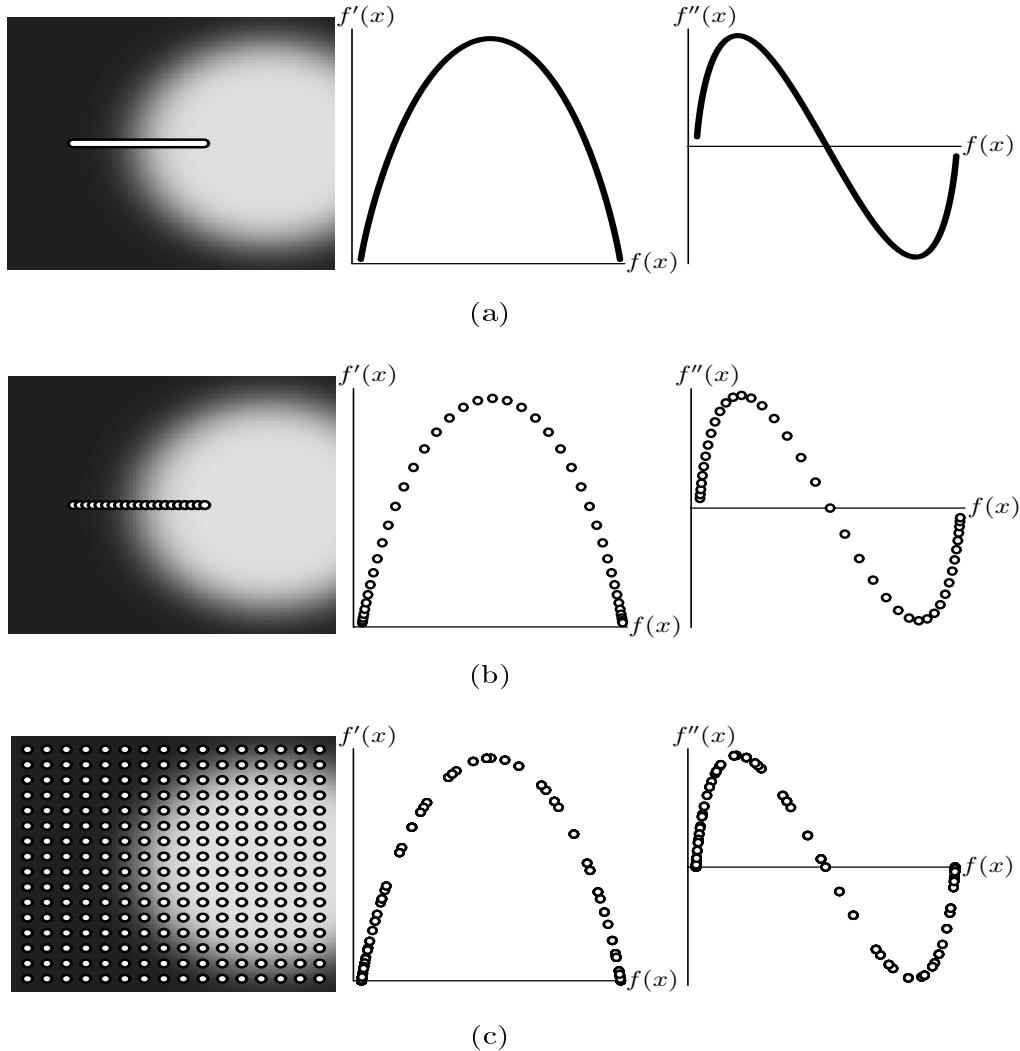


Abbildung 4.6.: Die Abtastung eines Bildes resultiert in einer Bogenstruktur im zweidimensionalen Intensitäts-/ Gradientenstärke-Histogramm. Abbildung (a) zeigt das Ergebnis einer kontinuierlichen Abtastung senkrecht zur Kante, während in (b) die Abtastung nur an diskreten Positionen senkrecht zur Kante erfolgt. Trotz der Reduktion der Abtastpunkte bildet das Histogramm dennoch die Beziehung zwischen $f(x)$ und $f'(x)$ ab. Diese Bogenstruktur bleibt selbst dann erhalten, wenn die Abtastung nur noch an diskreten Punkten, die gleichmäßig über das gesamte Bild verteilt sind, durchgeführt wird (s. Abb. (c)). Quelle: [Kin99].

Berechnung der Gradientenstärke

Kindlmann et al. [KD98] nutzen die von Höhne et al. [HB] vorgeschlagenen zentralen Differenzen für die Berechnung der ersten Ableitung.

$$\begin{aligned}
 f'(x) &\approx \frac{f(x+d) - f(x-d)}{2d} \\
 &= \frac{1}{2d}f(x+d) - \frac{1}{2d}f(x-d) \\
 &= -\frac{1}{2d}f(x-d) + \frac{1}{2d}f(x+d) \\
 &= [f(x-d) \ f(x) \ f(x+d)]^T \begin{bmatrix} -\frac{1}{2d} & 0 & \frac{1}{2d} \end{bmatrix} \quad (4.6)
 \end{aligned}$$

Mit einer Schrittweite $d = 1$ ergibt sich die folgende Filtermaske $[-0,5 \ 0 \ 0,5]$ für die Berechnung der ersten Ableitung in x-Richtung. Um mit ganzzahligen Werten zu rechnen, wird der Normierungsschritt $(1/2d)$ häufig erst nach der Berechnung des Filterergebnisses durchgeführt. Damit sieht die Filtermaske dann wie folgt aus $[-1 \ 0 \ 1]$ und der Normierungswert beträgt 2. Dies entspricht einer 1D-Sobelfiltermaske, wie sie in der Bildverarbeitung sehr häufig für die Kantendetektion verwendet wird. Wird diese Filtermaske auf die Nachbarzeilen ausgeweitet und dort nur halb so stark gewichtet, so entsteht die 2D-Filtermaske für die x-Richtung.

$$x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

Die Filtermaske für die Detektion von Kanten in y-Richtung ergibt sich analog dazu.

$$y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Für die Normierung des Filterergebnisses muss durch die Summe der Beträge der Elemente des Filterkerns (hier 8) geteilt werden. Die Filterung eines Volumendatensatzes erfolgt über 3D-Filtermasken in alle drei Raumrichtungen. In Abbildung 4.7 sind diese drei Filtermasken graphisch dargestellt.

Das Ergebnis der Filterung mit den drei 3D-Sobelfiltermasken ist ein Vektor. Der Betrag dieses Vektors entspricht der gesuchten Gradientenstärke. Um diesen Betrag zu normieren, muss nach der Berechnung der Differenzen durch eine Filtermaske das Ergebnis durch die Summe der Beträge der Kernelemente (32 in 3D) geteilt werden. Weiterhin sollte bei der Berechnung die Ausdehnung der Voxellemente, das sogenannte *Spacing* h , mit einbezogen werden. Die Berechnung der ersten Ableitung erfolgt dann, wie in Gleichung 4.7 gezeigt, in Weltkoordinaten.

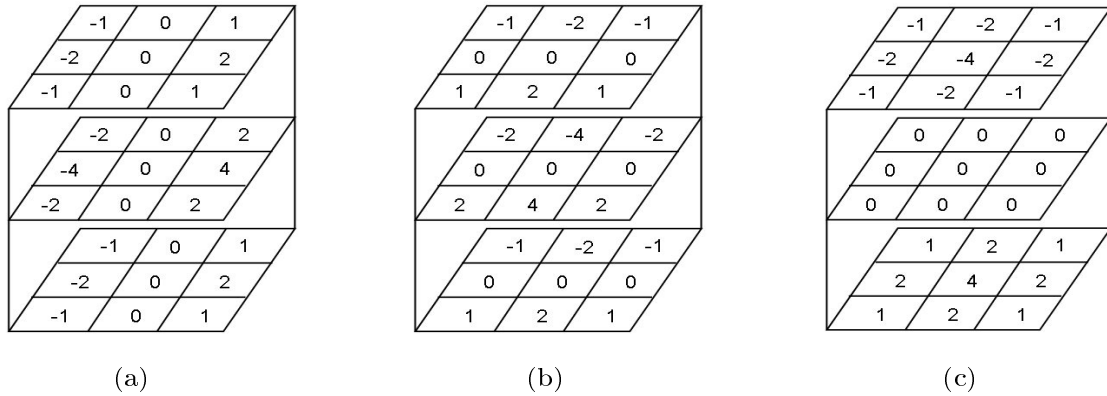


Abbildung 4.7.: 3D-Sobeloperatoren für die Berechnung der ersten Ableitung aus Volumendatensätzen. Quelle: [Stö04].

$$\begin{aligned}
 f'(x) &\approx \frac{f(x + dh) - f(x - dh)}{2dh} \\
 &= [f(x - dh) \ f(x) \ f(x + dh)]^T \begin{bmatrix} -\frac{1}{2dh} & 0 & \frac{1}{2dh} \end{bmatrix} \quad (4.7)
 \end{aligned}$$

Mit $d = 1$ und einer Voxelgröße h von 0,4 mm ergibt sich dann folgende 1D-Filtermaske $[-1, 25 \ 0 \ 1, 25]$ oder ein Normierungswert von 2,5. Um *Presets* von gradientenbasierten TFs bei gleicher Fragestellung an die Visualisierung zu verwenden, ist zusätzlich eine Standardisierung der Rekonstruktion der Bilddaten notwendig (vgl. 2.5.1 und [Stö04]). Das Ergebnis der Filterung mit den 3D-Sobeloperatoren und der Berechnung des Vektorbetrags ist ein zweiter Volumendatensatz, dessen Voxelwerte der Gradientenstärke an der jeweiligen Position im Originalvolumen entsprechen.

Generierung des 2D-Histogramms

Nach der Berechnung der Gradienteninformation wird das zweidimensionale Intensitäts-/Gradientenstärke-Histogramm berechnet, indem die beiden Volumen an festen Punkten (jedes Voxel) abgetastet werden. Die sich ergebende Wertekombination wird anschließend im Histogramm abgetragen. Die zugehörige Position im Histogramm, an der der Zähler um eins erhöht werden muss, ergibt sich aus den beiden Werten des aktuellen Voxels und kann über die folgenden Formeln bestimmt werden:

$$\begin{aligned}
 BinIntensität &= [VoxelIntensität / BinGrößeIntensität] \\
 BinGradSt &= [VoxelGradSt / BinGrößeGradSt] \quad (4.8)
 \end{aligned}$$

In Abbildung 4.8 wird schematisch gezeigt, wie die Erzeugung eines 2D-Histogramms abläuft.

$\text{Max}(I) = 2549$, $\text{Max}(\|f'\|) = 933$, Histogrammauflösung = 20×14

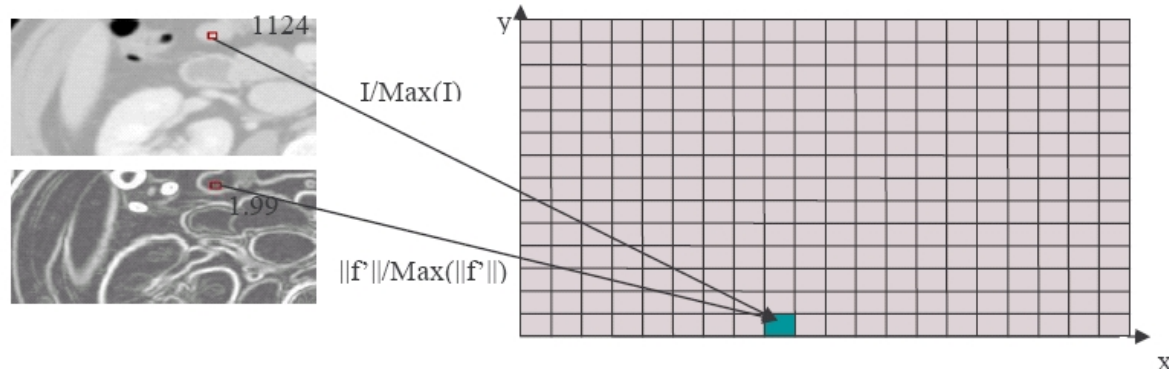


Abbildung 4.8.: Anhand der vorliegenden Bildinformation und der Histogramminformation (Anzahl der Bins und deren Bereiche) kann, wie hier beispielhaft gezeigt, das Bin bestimmt werden, dem das aktuelle Voxel zugeordnet werden muss. Quelle: [Stö04].

Das resultierende Histogramm kann in ein Bild umgewandelt werden, indem die Anzahl der Voxel, die aufgrund ihrer Eigenschaften einem Bin zugeordnet wurden, als Grauwert interpretiert wird. Je höher die Anzahl der Voxel in einem Bin ist, desto dunkler ist das entsprechende Pixel im Bild dargestellt. Ein 2D-Histogramm über Intensität und Gradientenstärke ist in Abbildung 4.9 gezeigt.



Abbildung 4.9.: Grauwertdarstellung des Intensitäts-/ Gradientenstärke-Histogramms eines CT-Schädel Datensatzes. Die Abzisse trägt die Intensität ab, während die Ordinate die Gradientenstärke repräsentiert.

In [Stö04] wird vorgeschlagen, die Werte des Histogramms mit dem natürlichen Logarithmus zu skalieren. Dies vertet die Darstellung der Histogramme für den Betrachter optisch auf, da Ausreißer mit einer hohen Anzahl an Voxeln stark herunterskaliert werden, während Bins mit einer geringen Anzahl kaum von der Skalierung betroffen sind. Neben der Anzeige des Histogramms kann auch die Qualität des Histogramms selbst verbessert werden. Möglichkeiten dazu werden in Abschnitt 5.3.3 aufgezeigt.

Untersuchung der Eigenschaften zweidimensionaler Histogramme

2D-Histogramme tragen die Häufigkeiten der im Datensatz auftretenden Wertekombinationen ab. Die Beziehung zwischen abgetragener Intensität $I(f(x))$ und dem Betrag des Gradienten $|f'(x)|$ wird über Bögen im Histogramm repräsentiert. Die Bögen ergeben sich, da bestimmte Wertekombinationen häufiger auftreten. Diese Kombinationen repräsentieren den Übergang zwischen zwei Geweben innerhalb des Volumens. Die Intensitäten der betreffenden Gewebe bilden die Basis des Bogens. Da an der Basis die Gradientenstärke annähernd Null ist, entsprechen die dort abgetragenen Voxel dem Inneren eines Gewebes. Dort können demnach die Intensitäten der Gewebe abgelesen werden. So ist z.B. in Histogrammen von CT-Aufnahmen eine direkte Zuordnung von Geweben wie z.B. Knochen möglich. Zu beachten ist allerdings, dass je nach Qualität des Datensatzes, die Basis eines Bogens über sehr viele Intensitätswerte ausgedehnt sein kann, was eine eindeutige Zuordnung erschwert. Der höchste Punkt eines Bogens repräsentiert die Mitte der Kante zwischen zwei Geweben. An dieser Stelle tragen beide Gewebe mit der gleichen Wahrscheinlichkeit zur Anzahl der Voxel der betreffenden Bins bei. Da an der Spitze des Bogens auch die Gradientenstärke am höchsten ist, repräsentieren diese Bins alle Voxel im Volumen, die der Kante am nächsten sind. Diese Eigenschaften des zweidimensionalen Intensitäts-/Gradientenstärke-Histogramms werden in Abbildung 4.10 an einem einfachen Beispiel mit zwei verschiedenen „Geweben“ und einem Hintergrund verdeutlicht.

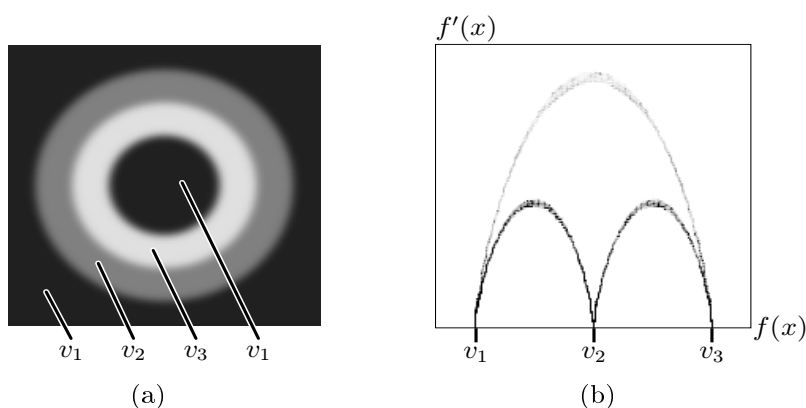


Abbildung 4.10.: Abbildung (a) zeigt ein künstlich erzeugtes Schichtbild mit zwei verschiedenen Geweben (Grauwerte v_2 und v_3) vor einem Hintergrund (v_1). Im Bild existiert jeweils eine Kante zwischen diesen Strukturen. Wird das Bild regelmäßig an jedem Pixel abgetastet und die Grauwert- / Gradientenstärke-Kombination im 2D-Histogramm abgetragen, so ergibt sich der in (b) dargestellte Plot mit drei Bögen. Die Bögen beginnen und enden an den Grauwerten der drei Strukturen. Jeder Bogen repräsentiert demnach eine Kante im Bild. Der Bogen zwischen v_1 und v_3 ist am höchsten, da dort der Grauwertunterschied und somit die Gradientenbeträge am größten sind. Quelle: [Kin99].

In [Stö04] werden die Bogenstrukturen der 2D-Histogramme sehr genau beschrieben, da ein Verfahren zur Spezifikation von TFs basierend auf der Detektion dieser Bögen untersucht wurde. Das Ziel gradientenbasierter Verfahren ist es, TFs zu definieren, indem Bereiche (z.B. die Bogenstrukturen oder Rechtecke) aus dem 2D-Histogramm ausgewählt und diese auf optische Parameter abgebildet werden. Es wird versucht Bereiche zu finden, die interessierende Strukturen im Volumen repräsentieren. Abhängig von den zugrunde liegenden Daten kann dies allerdings sehr schwer sein, da nicht immer eine Bogenstruktur erkennbar ist. Da in MRT-Aufnahmen Knochen und Luft fast im gleichen Grauwertbereich liegen, ist eine Unterscheidung im Histogramm unmöglich. Das Resultat ist ein einziger großer Bogen zwischen wasserhaltigen Geweben und Regionen wie Luft und Knochen ohne Wasser (vgl. [RBS05]). Das Histogramm kann den Nutzer in einem solchen Fall kaum bei der Spezifikation einer TF für die Darstellung wichtiger Strukturen unterstützen. Im folgenden Abschnitt wird eine Möglichkeit vorgestellt, wie die Auswahl der Bereiche aus dem Histogramm automatisiert und somit der Aufwand für den Nutzer verringert werden kann.

4.4. Spatialized Transfer Functions

Die von Roettger et al. [RBS05] vorgestellten *Spatialized Transfer Functions* (STF) zählen zu den mehrdimensionalen gradientenbasierten TFs. Sie wurden entwickelt, um die Spezifikation von TFs zu erleichtern. Roettger et al. präsentieren einen halbautomatischen Ansatz, der in dieser Arbeit sowohl für die Definition gradientenbasierter TFs als auch für die multimodale Visualisierung realer medizinischer Datensätze untersucht wird.

Kindlmann et al. haben die Positionsinformation bei der Generierung des zweidimensionalen Intensitäts-/ Gradientenstärke-Histogramms eliminiert (vgl. Abschnitte 4.3.2, 4.3.3 und [KD98, KKH01]), da Histogramme als statistisches Mittel eine abstraktere Form der Beschreibung von Daten darstellen. Im Ansatz von Roettger et al. wird diese Positionsinformation bei der Generierung der STF aus dem Histogramm mit eingebracht. Sie kann die Definition der TF unterstützen, da Strukturen innerhalb eines Volumens sowohl über ihre messbaren Eigenschaften als auch als räumlich miteinander verbundene Regionen an einer bestimmten Position im Volumen definiert sind. Die Ortsinformation ermöglicht es, das zweidimensionale Histogramm so zu klassifizieren, dass die entstehenden Regionen im Histogramm zusammenhängende Strukturen im Datensatz repräsentieren. Diese werden dadurch im Histogramm für den Betrachter erkennbar gemacht. Durch die Verwendung der Ortsinformation für die Generierung der TF wird demnach eine Segmentierung der Wertebereiche, welche von der TF abgebildet werden, durchgeführt (vgl. [RBS05]). Über die Positionsinformation der einzelnen Voxel geht auch die Größeninformation der zusammenhängenden Strukturen mit in die Generierung der STF ein. So kann die Spezifikation an die in den Abschnitten 2.2.2, 2.3.2 und 2.4.2 vorgestellten Daten und abgebildeten Körperregionen angepasst werden.

In den nachfolgenden Abschnitten wird zunächst die halbautomatische Generierung der STF beschrieben, wobei näher auf die Histogrammberechnung und die verwendete Ortsinformation eingegangen wird. Die Generierung des 2D-Histogramms und der Ortsinformation werden parallel in einem Vorverarbeitungsschritt durchgeführt. Anschließend werden die automatische Transparenzzuweisung sowie die Interaktion mit dem Histogramm, welche für die Generierung einer STF nötig ist, erläutert.

4.4.1. Automatische Generierung der Ortsinformation

Die Idee der STFs liegt darin, die Information von welchem Ort ein gemessener Datenwert innerhalb des Volumens stammt, mit in den Generierungsprozess der TF einzubeziehen. Durch die Klassifikation des zweidimensionalen Intensitäts-/ Gradientenstärke-Histogramms anhand dieser Ortsinformation wird die Spezifikation der TF erleichtert. Die berechnete Ortsinformation wird dazu benutzt, die Bins, deren Voxel im Volumen genügend dicht beieinander liegen, zu einem Bereich zusammenzufassen. Damit wird der Nutzer bei der Suche nach Bereichen, die interessierende Strukturen repräsentieren, unterstützt.

Die in den Intensitäts-/ Gradientenstärke-Histogrammen auftretenden Bögen sind in realen medizinischen Datensätzen häufig sehr schwach repräsentiert und kaum voneinander zu unterscheiden. Dies hängt mit der Rekonstruktion der Daten und der Bandpassfilterung zusammen. Außerdem wird ein Gewebe nie auf genau einen Intensitätswert abgebildet. Im Falle von MRT-Aufnahmen werden mehrere Strukturen (Luft und Knochen) auf die gleichen Grauwerte abgebildet. Während sich andere Ansätze auf die nutzergeführte Markierung der Bögen (vgl. [Stö04]) oder anders geformter Bereiche des Histogramms (vgl. [KKH01] und [HST+03]) spezialisiert haben, verfolgen die STFs einen anderen Weg. Es wird versucht das Histogramm automatisch in möglichst viele Bereiche zu unterteilen, wobei diese Bereiche unterschiedliche Strukturen im Volumen repräsentieren. Diese Klassifikation erfolgt unter Verwendung der Ortsinformation. Es werden keine 5-dimensionalen TFs (Intensität, Gradientenstärke, x-,y- und z-Koordinate) definiert, sondern die Ortsinformation wird in Form von Schwerpunkten und der Varianz um diese Schwerpunkte gespeichert. Die Ortsinformation wird dazu genutzt, die Farbzuordnung der STF zu generieren.

Nach [RBS05] wird die Ortsinformation in Form eines Schwerpunktes und der dazugehörigen örtlichen Varianz der Voxel für jedes Bin im Histogramm repräsentiert. Die Berechnung der Ortsinformation erfolgt im Weltkoordinatensystem. Dies ermöglicht es zu untersuchen, ob eine STF, die auf den Ortsinformationen eines Datensatzes basiert, auch auf andere Datensätze mit einer abweichenden Voxelgröße angewendet werden kann. Voraussetzung hierfür ist natürlich, dass die beiden Datensätze die gleiche Körperregion darstellen und das die Visualisierung die gleiche Fragestellung beantworten soll. Die Generierung der Ortsinformation wird einmalig in einem Vorverarbeitungsschritt parallel zur Berechnung des 2D-Histogramms ausgeführt. Seien $p_i(f, g)$, $i = 1, \dots, n$ die Positionen der n Voxel in Weltkoordinaten aus dem Datensatz, die in das Bin im Histogramm mit $H(f, g) = n$ eingetragen werden. Dann kann aus diesen Positionsinformationen für jedes Bin ein Schwerpunkt wie folgt berechnet werden (vgl. [RBS05]):

$$b(f, g) = \frac{1}{n} \sum_{i=1}^n p_i(f, g) \quad (4.9)$$

Der Schwerpunkt wird in Form eines Vektors mit den drei Raumkomponenten gespeichert. Die Varianz der Voxelpositionen um den Schwerpunkt wird in [RBS05] wie folgt berechnet:

$$\begin{aligned} v(f, g) &= \frac{1}{n} \sum_{i=1}^n \|p_i(f, g) - b(f, g)\|^2 \\ &= \frac{1}{n} \sum_{i=1}^n [(p_{ix}(f, g) - b_x(f, g))^2 + \\ &\quad (p_{iy}(f, g) - b_y(f, g))^2 + (p_{iz}(f, g) - b_z(f, g))^2] \end{aligned} \quad (4.10)$$

Im Gegensatz zum Schwerpunkt hat die Varianz demnach nur eine Komponente. Aus Gleichung 4.10 ergibt sich, dass der Schwerpunkt für die Berechnung der Varianz bereits bekannt sein muss. Die Kombination aus Schwerpunkt und Varianz eines Histogrammbins wird auch als Tupel bezeichnet.

4.4.2. 2D-Histogramme für die Spatialized Transfer Functions

Generierung des zweidimensionalen Histogramms

Die Erzeugung des zweidimensionalen Intensitäts-/ Gradientenstärke-Histogramms $H(f, g)$ erfolgt wie in Abschnitt 4.3.3 beschrieben. Je nach Qualität des vorliegenden Datensatzes, kann das erzeugte Histogramm starkes Rauschen und Bins mit einer Voxellanzahl geringer als 5 beinhalten. Dies wirkt sich negativ auf die spätere Klassifikation des Histogramms aus. Um die Qualität des Histogramms zu verbessern, erfolgt vor der Histogrammgenerierung eine Überabtastung (engl. *Oversampling*) des Datensatzes. Hierbei wird die Voxellanzahl in jede der drei Raumrichtungen verdoppelt und die Ausdehnung der Voxel in jede Richtung halbiert. Insgesamt wird die Anzahl der Voxel somit verachtacht. Die Werte der neuen Abtastpunkte werden aus den Originalvoxelwerten durch trilineare Interpolation ermittelt. Roettger et al. [RBS05] schlagen in ihrer Arbeit eine weitere Möglichkeit der Rauschreduktion vor, die auch in dieser Arbeit verwendet wurde. Dabei wird ein Voxel nicht nur in das, anhand seiner Eigenschaften ermittelte, Bin des Histogramms eingetragen, sondern auch in alle benachbarten Bins. Dies ist möglich, da benachbarte Regionen im Volumen auf benachbarte Regionen im Histogramm abgebildet werden (vgl. [RBS05]). Wird der Radius der Nachbarschaft $k = 2$ gesetzt, so sind das die 24 nächsten Nachbarn.

Die Verbindung von Überabtastung und k -Nachbarschaft steigert die Anzahl der Voxel die in das Histogramm gezählt werden enorm und verbessert somit die Qualität sowohl des Histogramms, als auch der anschließenden Klassifikation (s. Abb. 4.11).

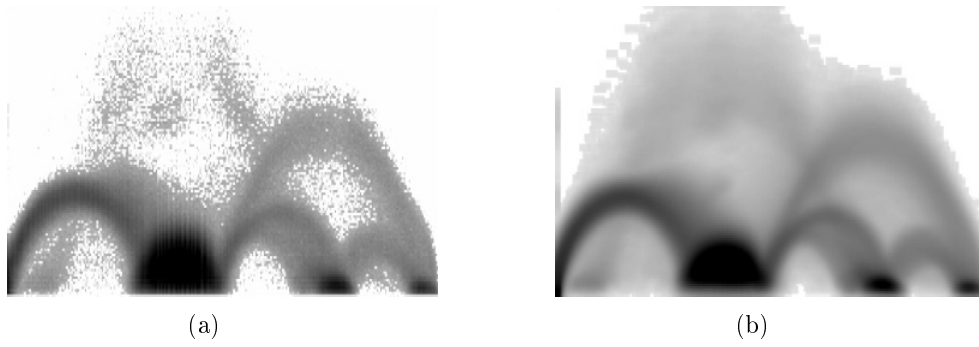


Abbildung 4.11.: Diese Abbildung verdeutlicht den Effekt von Überabtastung und k -Nachbarschaft auf das 2D-Histogramm anhand des synthetischen CT-Zahndatensatzes. Durch die Überabtastung und die k -Nachbarschaft wird die Anzahl der Voxel, die den Bins zugeordnet werden stark erhöht ((b)). Die für ein Intensitäts-/ Gradientenstärke-Histogramm typische Bogenstruktur wird in Teilen des Histogramms erst durch die Überabtastung und die Anwendung der k -Nachbarschaft ersichtlicher. Quelle: [RBS05].

Klassifikation des 2D-Histogramms

Um die Bereiche im Histogramm zu finden, welche zusammenhängende Strukturen im Volumen repräsentieren, wird die berechnete Ortsinformation dazu verwendet, das aufgefüllte Histogramm zu klassifizieren. Hierzu muss nach [RBS05] ein Klassifikationsparameter r definiert werden. Dieser sogenannte Klassifikationsradius legt die maximale Ausdehnung der Struktur im Volumen fest, die durch einen Bereich im Histogramm definiert wird. Somit wird es möglich die Klassifikation an die in den Volumen vorkommenden Strukturen anzupassen. So sind z.B. in einem CT-Kopfdatensatz die Schädelknochen wesentlich kleiner als das zu extrahierende Skelett im Thoraxbereich.

Einem beliebigen Referenztuplel $T_0 = (f, g)$ werden bei der Klassifikation alle Tuplel T des Histogramms zugeordnet, deren Abstand der Schwerpunkte zum Referenzschwerpunkt nicht größer als der Radius r ist ($\|b(T) - b(T_0)\| < r$). Eine weitere Voraussetzung für die Zusammengehörigkeit zweier Tuplel ist die Varianz um den Schwerpunkt. Mit zunehmender Varianz wird es unwahrscheinlicher, dass ein Bin die gleiche Struktur im Histogramm repräsentiert, da die räumliche Nähe nicht gewährleistet ist. Um diese beiden Voraussetzungen für die Berechnung der Klassifikation greifbar zu machen, wird von Roettger et al. die Verwendung folgender Klassifikationsnorm vorgeschlagen:

$$\begin{aligned}
 N(T, T_0) &= \|b(T) - b(T_0)\| + |v(T_0) - v(T)| \\
 &= (b_x(T) - b_x(T_0))^2 + (b_y(T) - b_y(T_0))^2 + \\
 &\quad (b_z(T) - b_z(T_0))^2 + (v(T_0) - v(T))^2
 \end{aligned} \tag{4.11}$$

Anhand dieser Norm wird entschieden, ob ein Tuplel T zur gleichen Struktur wie das Referenztuplel T_0 gehört und somit auch zum gleichen Bereich klassifiziert werden kann. Das gesamte Histogramm wird unter Verwendung der Norm $N(T, T_0)$ klassifiziert. Dazu

werden die Bins zunächst absteigend nach der Anzahl der dazugehörigen Voxel sortiert. Beginnend bei dem noch nicht klassifizierten Bin mit der höchsten Anzahl, werden alle Bins mit geringerer Anzahl, die noch keinem Bereich angehören und deren Norm N kleiner als der Radius r ist, der Klasse dieses Bins zugeordnet. Dieser Vorgang wird wiederholt, bis das gesamte Histogramm klassifiziert ist.

```
1: begin
2: unclassifiedBins := (AllBins ∈ Histogramm);
3: unclassifiedBins.sort();
4: unclassifiedBins.goToBegin();
5: classifiedBins := ∅;
6: while unclassifiedBins ≠ ∅ do
7:   begin
8:     referenceBin = unclassifiedBins.getBinWithHighestCount();
9:     activeColor = getRandomColor();
10:    referenceBin.setColor(activeColor);
11:    classifiedBins.addBin(referenceBin);
12:    unclassifiedBins.deleteBin(referenceBin);
13:    while (examinedBin = unclassifiedBins.getNextBin()) ≠ NULL do
14:      begin
15:        if Norm(examinedBin, referenceBin) ≤ Radius then
16:          begin
17:            examinedBin.setColor(activeColor);
18:            classifiedBins.addBin(examinedBin);
19:            unclassifiedBins.deleteBin(examinedBin);
20:          end
21:        end
22:      unclassifiedBins.goToBegin();
23:    end
24: end.
```

Bei der Betrachtung eines klassifizierten Histogramms (s. Abb. 4.12) fallen vor allem die komplexen Formen der ermittelten Bereiche auf. Eine manuelle Bereichsauswahl durch den Nutzer wäre mit dieser Genauigkeit nicht möglich.

Automatische Farbzweisung durch räumliche Information

Die erzeugte Farbtransferfunktion (FTF) entspricht in ihrer Größe exakt dem 2D-Histogramm. Sie weist einem Voxel mit einer bestimmten Intensitäts- und Gradientenkombination (f, g) einen RGB Farbwert zu und ist definiert als:

$$FTF_{RGB}(f, g) = C_{RGB}, \forall T \in H(f, g) : N(T, T_0) < r \quad (4.12)$$

Der Farbwert C_{RGB} wird in [RBS05] während der Klassifikation zufällig aus dem Intensitäts- und Gradientenwert des Referenzstupels bestimmt und allen Bins des dazugehörigen Bereichs zugewiesen. Wird das gesamte klassifizierte Histogramm im Rendingschritt der Visualisierungspipeline als FTF $FTF_{RGB}(f, g)$ verwendet, so kann eine automatische Visualisierung eines Datensatzes erzeugt werden. Das klassifizierte Histogramm wird dem Nutzer unter Verwendung der zufällig erzeugten Farbwerte, wie in Abbildung 4.12 dargestellt, angezeigt.

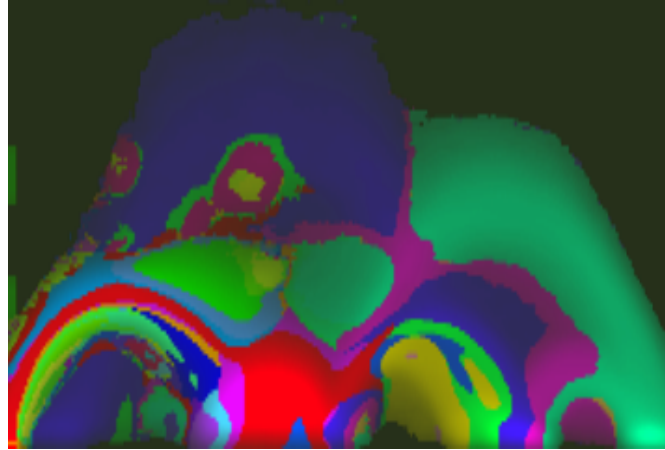


Abbildung 4.12.: Darstellung eines klassifizierten Intensitäts-/ Gradientenstärke-Histogramms des synthetischen CT-Zahndatensatzes unter Verwendung zufällig erzeugter Farben für die Klassen. Quelle: [RBS05].

Automatische Zuweisung von Transparenz

Um Bereiche mit geringem Gradienten auszublenden und so die Grenzen zwischen den Geweben in der Visualisierung hervorzuheben, wird dem Datensatz in [RBS05], neben der Farbzweisung für die Visualisierung, noch eine Transparenz zugeteilt. Dabei wird die Gradientenstärke als Parameter verwendet. Die Zuweisung erfolgt über eine 1D-TTF, die wie folgt definiert ist:

$$TTF_{\alpha}(g) = g \times c_{\alpha} \quad (4.13)$$

Der Parameter g ist die berechnete Gradientenstärke des zu rendernden Voxels und c_{α} eine zuvor global definierte konstante Transparenz.

4.4.3. Interaktion mit der Transferfunktion

Um einen Datensatz zu untersuchen und die STF entsprechend der Visualisierungsfragestellung anzupassen, wird in [RBS05] eine Interaktionsmöglichkeit mit dem klassifizierten Histogramm vorgeschlagen. Für die Definition einer angepassten STF müssen einzelne Bereiche des klassifizierten Histogramms ausgewählt werden. Dazu muss der Nutzer die ihn interessierenden Bereiche in der Histogrammvisualisierung anklicken. Ausgewählte Bereiche werden in der Darstellung des klassifizierten Histogramms hervorgehoben, indem sie mit gesättigten Farben dargestellt werden. Es können nacheinander sowohl meh-

rere Bereiche aktiviert als auch deaktiviert werden. Nur die vom Nutzer ausgewählten Bereiche des 2D-Histogramms werden für die Visualisierung durch die STF genutzt. Die Form der ausgewählten Bereiche ergibt sich durch die automatische Klassifikation und berücksichtigt somit die Eigenschaften der Daten sowie die berechnete Ortsinformation. Andere Arbeiten sind in der Form der gewählten Bereiche limitiert. So wird in [Stö04] und [HST+03] nach Bögen gesucht und in [KKH01] rechteckige und dreieckige Regionen verwendet. Problematisch werden diese Ansätze, wenn das generierte 2D-Histogramm keine ausreichende Information in Form von Bogenstrukturen oder anderen Maxima bietet, die als Anhaltspunkt für die Platzierung des auszuwählenden Bereichs dienen. Die Auswahl interessierender Bereiche setzt ein bestimmtes Maß am Expertenwissen voraus. Der Nutzer muss die Lage und Form der Bereiche im Histogramm kennen, welche die ihn interessierende Struktur im Volumen repräsentieren. Durch die halbautomatische Klassifikation des Histogramms wird der Aufwand reduziert, indem die Form der Bereiche bereits vorgegeben wird.

Eine weitere Interaktionsmöglichkeit bietet die Änderung des Klassifikationsradius r . Wird dieser durch den Nutzer variiert, so erfolgt eine erneute Klassifikation des Histogramms. Da hierzu nicht nocheinmal alle Voxel des Datensatzes betrachtet werden müssen, sondern nur auf dem wesentlich kleineren Histogramm gearbeitet wird, kann die Änderung des Radius interaktiv erfolgen. Somit wird der Nutzer bei der Exploration des Histogramms unterstützt. Ein kleinerer Radius bewirkt eine Erhöhung der Anzahl detektierter Bereiche im Histogramm. Dies bewirkt jedoch, dass große Strukturen, die im Volumen vorkommen, durch mehrere Bereiche im Histogramm repräsentiert werden. Dies führt dazu, dass all diese Bereiche im Histogramm ausgewählt werden müssen, um eine Struktur im Datensatz vollständig in der Visualisierung anzuzeigen. In diesem Fall wäre eine solche Struktur aus Voxeln mit unterschiedlicher Einfärbung zusammengesetzt. Im Umkehrschluss heißt das, dass ein zu großer Radius dazu führt, dass die Bereiche im Histogramm zu groß werden und somit kleinere Strukturen in der Visualisierung verloren gehen, da sie zusammen mit anderen Strukturen angezeigt werden.

4.4.4. Spatialized Transfer Functions für multimodale Visualisierungen

Für die Anwendung von STFs zur Erstellung multimodaler Visualisierungen werden in [RBS05] 2D-STFs basierend auf den Intensitätswerten der beiden darzustellenden Datensätze vorgeschlagen. Bis auf die Art der im Histogramm abgetragenen Daten verläuft die Generierung des Histogramms wie zuvor beschrieben. Auch die Interaktion und somit die Spezifikation der STF gleichen den oben beschriebenen Abläufen. Aufgrund der Möglichkeit STFs auch für die Erstellung multimodaler Volumenvisualisierungen nutzen zu können, wurden die STFs als Grundlage für diese Arbeit gewählt (vgl. Abschnitt 5.2).

4.5. Spezifikation von Transferfunktionen

Die Definition von geeigneten TFs ist eine schwierige und häufig zeitaufwändige Aufgabe der direkten Volumenvisualisierung. Mit zunehmender Dimensionalität der TFs steigt dieser Aufwand erheblich an. Aus diesem Grund wurden mehrdimensionale TFs bisher noch nicht in klinische Workstations integriert. Das Ziel der Forschung auf dem Gebiet der TFs ist die Vereinfachung und Beschleunigung der Spezifikation. Dieser Umstand gilt als eines der 10 schwerwiegendsten Probleme in der Volumenvisualisierung. In der Literatur wird zwischen vier verschiedenen Ansätzen für die Lösung dieses Problems unterschieden (vgl. [PLB+01]).

Trial & Error Verfahren, bei denen der Nutzer die Stützstellen der TF manuell vorgeben muss und somit sehr viele Parameter einzustellen hat, können aufgrund des hohen Spezifikationsaufwandes für diese Arbeit nicht verwendet werden.

Design Galleries werden bei der Spezifikation von TFs verwendet, um den Aufwand für den Nutzer auf ein Minimum zu reduzieren. Durch dieses bildorientierte Verfahren reduziert sich die Interaktion auf die Auswahl einer geeigneten Visualisierung. Die Anforderung, einzelne Strukturen des Datensatzes hervorzuheben und weitere hinzuschalten zu können, verhindert allerdings die Anwendung der bildorientierten Spezifikation durch *Design Galleries*.

Die datenorientierte Spezifikation kann sowohl mit (kantenbasiert) als auch ohne ein zugrunde liegendes Modell der Daten erfolgen (Kontur Spektrum). In dieser Arbeit wird ein gradientenbasierter Ansatz verfolgt und auf das in Abschnitt 4.3 beschriebene Kantenmodell und die Gradienteninformation zurückgegriffen. Im Folgenden werden die für eine Spezifikation von TFs zur Verfügung stehenden Hilfsmittel vorgestellt.

Die Spezifikation einer TF umfasst drei Teilaufgaben. Zum ersten müssen die Wertebereiche der verwendeten Parameter (z.B. Intensität) festgelegt werden, die von der TF auf optische Eigenschaften abgebildet werden sollen. Zum zweiten müssen eben diese optischen Eigenschaften für den betreffenden Bereich angegeben werden. Im dritten Schritt muss die resultierende Visualisierung bewertet werden, um die Eignung der TF zu überprüfen. Für diese drei Aufgaben stehen dem Anwender verschiedene Hilfsmittel zur Verfügung, die im folgenden vorgestellt werden.

4.5.1. Passive Hilfsmittel für die Spezifikation

Passive Hilfsmittel unterstützen den Nutzer bei der Spezifikation von TFs, indem sie ihm zum einen ein Feedback über ausgeführte Tätigkeiten geben und z.B. Bereiche hervorheben, die von Änderungen betroffen sind. Zum anderen können sie ihn dabei unterstützen die Eigenschaften eines Datensatzes zu verstehen. Die in dieser Arbeit verwendeten passiven Hilfsmittel sind:

Numerische Ausgaben Die Ausgabe von Zahlen kann dem Nutzer einerseits wichtige Informationen über den Datensatz liefern, indem z.B. der minimale und maximale Grauwert im Volumen oder die Anzahl der Bins im Histogramm angezeigt werden. Andererseits bieten sie die Möglichkeit, ein Feedback über ausgeführte Aktionen zu

geben. So kann die Position, an der der Nutzer eine Auswahl getroffen hat (engl. *Picking*) ebenso wie die Anzahl der Voxel, die zu einem ausgewählten Bin gehören, präsentiert werden.

Histogramme Histogramme werden bei der Spezifikation von TFs häufig zur Orientierung in den Daten eingesetzt. Eindimensionale Histogramme stellen die Häufigkeiten eines im Volumen auftretenden Merkmals dar. Sie können als Hilfsmittel für die Spezifikation von 1D-Transferfunktionen, die nur über die Intensität oder den Gradientenbetrag definiert sind, verwendet werden. Im Histogramm wird anhand von lokalen Maxima ersichtlich, welche Grauwerte oder Gradienten besonders häufig vorkommen. Daraus kann z.B. bei CT-Datensätzen auf Gewebe geschlossen werden, da die gemessenen Hounsfield Werte bestimmten Geweben zugeordnet sind (vgl. Abschnitt 2.2.1). Werden Gradienten-Histogramme verwendet, so lassen sich evtl. Grenzen zwischen Geweben ausmachen, welche durch eine Transparenz-Transferfunktion hervorgehoben werden sollen. Durch logarithmische Skalierung werden kleine, lokale Maxima hervorgehoben und die TF kann daran angepasst werden (vgl. [PM04]).

Werden zwei- oder mehrdimensionale Histogramme wie z.B. bei der Definition von mehrdimensionalen gradientenbasierten TFs verwendet, so hilft die im Histogramm enthaltene Information (z.B. Lage von Bögen im Intensitäts-/ Gradientenstärke-Histogramm) dabei, relevante Bereiche auszuwählen. Histogramme werden dazu häufig als Hintergrundbild in der graphischen Oberfläche eines Tools für die Spezifikation von TFs verwendet. In [RBS05] und dieser Arbeit wird das klassifizierte 2D-Histogramm auf der Programmoberfläche dargestellt und dient dem Nutzer als Grundlage für die Spezifikation der STF.

Weitere passive Hilfsmittel, die häufig in der Literatur zu finden sind, werden im Folgenden kurz vorgestellt:

Schicht-Previews als Feedback Schichtbilder werden als passives Hilfsmittel bei der Definition von TFs verwendet. Sie ermöglichen ein schnelles Feedback über die Einfärbung der Bereiche im Schichtbild, die von der aktuell definierten TF betroffen sind. Dies ist effektiv, da nicht das gesamte Volumen neu gerendert werden muss. Der Einbau einer Scrollmöglichkeit durch das Volumen bietet sich an, um die zu erwartende Qualität der Visualisierung besser abschätzen zu können. Feedbacks in Form von Schichtbildern werden in verschiedenen Ansätzen verwendet (vgl. [CKLG98, PLB⁺01, TLM03]).

Referenz-Transferfunktionen Werden TFs für standardisierte Daten wie z.B. einheitlich rekonstruierte CT-Datensätze unter Beachtung der gleichen Fragestellung definiert, so ist es möglich, eine bereits bestehende TF, auch auf den aktuell untersuchten Datensatz anzuwenden. Die hierzu notwendigen Anpassungen können z.B. über eine Angleichung des Referenzhistogramms, welches der Spezifikation der TF zugrunde lag, ermittelt werden. Die resultierende Transformation wird anschließend auf die bestehende TF angewendet, um diese auf den aktuellen Datensatz zu adaptieren. Aufgrund der vielen Parameter, die das Histogramm beeinflussen, wurde dieser Ansatz bisher nur bei stark standardisierten CT-Angiographiedatensätzen

eingesetzt (vgl. [RSHSG00] und [HST⁺04]). In diesen Aufnahmen sind die Blutgefäße mit Kontrastmittel gefüllt und heben sich so eindeutig vom restlichen Gewebe ab. Das Ziel der Visualisierung ist die alleinige Darstellung der Blutgefäßstrukturen.

4.5.2. Interaktive Hilfsmittel für die Spezifikation

Mit Hilfe interaktiver Hilfsmittel ist es dem Nutzer möglich, direkt Einfluss auf die Parameter der TF oder der Visualisierung zu nehmen. In dieser Arbeit wurden die folgenden interaktiven Hilfsmittel verwendet:

Pick-Funktionalität In dieser Arbeit wird die *Pick*-Funktionalität für die Definition der 2D-TF verwendet, indem Bereiche in der Visualisierung des klassifizierten Histogramms vom Benutzer angeklickt und dadurch zur TF hinzugefügt werden. Weiterhin kann durch den Einsatz einer *Pick*-Funktion aus der Schichtbilddarstellung ein aktives Hilfsmittel gemacht werden, indem interessierende Punkte im Schichtbild angeklickt werden können und die Position der dazugehörigen Wertekombination im 2D-Histogramm markiert wird (vgl. [Stö04]). Der Nutzer kann sich auf diese Weise sehr effektiv mit dem Datensatz und der Repräsentation in Form des Histogramms vertraut machen. Die *Pick*-Funktionalität wird in der Literatur auch auf das visualisierte 3D-Volumen angewendet (vgl. [KKH01]).

Numerische Eingaben Numerische Eingabefelder und dazugehörige Hilfsmittel wie Schieberegler (engl. *Slider*) ermöglichen es dem Nutzer, die Parameter der TF zu bestimmen. So kann er zum Beispiel den Wert für die globale Transparenz verändern und somit Einfluss auf die Durchsichtigkeit nehmen. Weiterhin hat er die Möglichkeit Parameter für die Erstellung der TF vorzugeben. Dies kann z.B. der Radius für die Klassifikation bei STFs sein, wodurch er das klassifizierte Histogramm auf den Datensatz oder seine Bedürfnisse anpassen kann.

Weitere interaktive Hilfsmittel, die häufig in der Literatur zu finden sind, werden im Folgenden kurz vorgestellt:

Kontrollpunkte Kontrollpunkte sind in Verbindung mit der *Pick*-Funktionalität ein interaktives Hilfsmittel für die Spezifikation eindimensionaler TFs. Besonders häufig werden sie in graphischen Benutzeroberflächen in Verbindung mit einem Histogramm des Parameters der TF verwendet. Durch ihre Lage über dem Histogramm und vom Benutzer angegebene Eigenschaft bestimmen sie die Merkmale der TF (vgl. [PM04]). Verwendung finden Kontrollpunkte z.B. im frei verfügbaren Visualisierungsprogramm VolView des Unternehmens Kitware Inc..

Komponentenfunktionen Komponentenfunktionen sind eine Erweiterung der Idee der Kontrollpunkte. Sie ermöglichen eine genauere und vielseitigere Spezifikation der TF (vgl. [CKLG98]). Allerdings erfordern sie auch einen sehr hohen Interaktionsaufwand, vor allem im mehrdimensionalen Fall (vgl. [Tap04]). Da dieser Spezifikationsaufwand möglichst reduziert werden soll, können Komponentenfunktionen in dieser Arbeit nicht verwendet werden.

5. Entwurf und Implementierung des Visualisierungssystems

Spatial information is important, because a feature by definition is a spatially connected region in the volume domain with a unique position and certain statistical properties. If we only use the statistical information ... we effectively ignore the most important part of a features definition.

ROETTGER ET AL., AUS [RBS05]

In den vorangegangenen Kapiteln wurden die Möglichkeiten, die bei der Konzeption eines multimodalen Visualisierungssystems zur Verfügung stehen, vorgestellt sowie anhand ihrer Eigenschaften und den Anforderungen an das zu entwickelnde Visualisierungssystem bewertet. Aus den gewählten Komponenten ergibt sich ein Konzept für das Visualisierungssystem, welches in diesem Kapitel beschrieben werden soll. Anhand dieses Konzeptes wird anschließend die Implementierung der multimodalen Visualisierung innerhalb des bestehenden Bildverarbeitungsprogramms erläutert.

5.1. Entwurf des multimodalen Visualisierungssystems

Mit dem entwickelten Visualisierungssystem soll es möglich sein Daten, der drei in Kapitel 2 vorgestellten bildgebenden Verfahren, in einer 3D-Darstellung abzubilden. Da die Strukturen aus den Datensätzen vollständig dargestellt werden sollen, wurde die direkte Volumenvisualisierung aus Abschnitt 3.4 gewählt. Diese erfüllt auch die Anforderung, dass keine aufwändige und fehleranfällige Vorbereitung der Daten erfolgen muss. Um die Fusion der Daten zu automatisieren und die Definition einer Fusionsfunktion unnötig zu machen, wurden die *Property Fusion* Pipeline und die *Color Fusion* Pipeline aus Abschnitt 3.4.3 gewählt. Aus der Wahl dieser beiden Fusionszeitpunkte ergeben sich zwei verschiedene Wege für die Definition der verwendeten STF's zur multimodalen Visualisierung. In der ersten Variante werden die beiden Datensätze bei der *Property Fusion*

bereits zu Beginn der Pipeline zu einem Datensatz mit zwei Eigenschaften fusioniert. Hierzu werden für jeden Voxel lediglich die beiden korrespondierenden Intensitätswerte gespeichert. Im anderen Ansatz erfolgt die unabhängige Verarbeitung der Datensätze durch die *Color Fusion* Pipeline, bei der die beiden Visualisierungen erst beim Rendering fusioniert werden. Bei der Fusion wird von beiden Datensätzen sowohl die Farbinformation als auch die Transparenz gleich gewichtet.

Für eine direkte Volumenvisualisierung müssen Transferfunktionen definiert werden, welche die Eigenschaften der Datensätze auf Farbe abbilden. Aus den Anforderungen geht hervor, dass die TFs über zwei Merkmale der Daten definiert sein sollen. Es werden die Intensität und der Gradientenbetrag eines Datensatzes oder die Intensitäten zweier Datensätze verwendet. Aus diesem Grund wird auf gradientenbasierte mehrdimensionale TFs zurückgegriffen. Die Möglichkeit einzelne Strukturen aus den Datensätzen zu extrahieren und beliebig weitere zur Visualisierung hinzuzufügen, wird durch die verwendeten *Spatialized Transfer Functions* geboten. Diese können für Datensätze aller drei bildgebenden Verfahren definiert werden.

Das Konzept der Transferfunktionsdefinition umfasst zwei verschiedene Ansätze, da zwei Fusions Pipelines gewählt wurden. In [RBS05] wird vorgeschlagen unter Verzicht auf die Gradienteninformation eine STF für zwei darzustellende Datensätze zu definieren (*Property Fusion*). Daneben ist es möglich für jeden Datensatz eine STF über Intensität und Gradientenbetrag zu definieren und die resultierenden Visualisierungen zu fusionieren (*Color Fusion*). In Abschnitt 5.2 werden die beiden Ansätze zur Verwendung der STFs für die multimodale Visualisierung näher erläutert.

Die Darstellung der Datensätze erfolgt über einen *Ray Casting* Algorithmus. Mit Hilfe dieses Ansatzes kann die Visualisierung ohne Parametrisierung durch den Nutzer und ohne Vorbereitung der Daten realisiert werden. Weiterhin kann das *Ray Casting* für die beiden zuvor genannten Ansätze verwendet werden.

Der erarbeitete Entwurf des multimodalen Visualisierungssystems ist mit den verwendeten Komponenten in Abbildung 5.1 übersichtlich dargestellt.

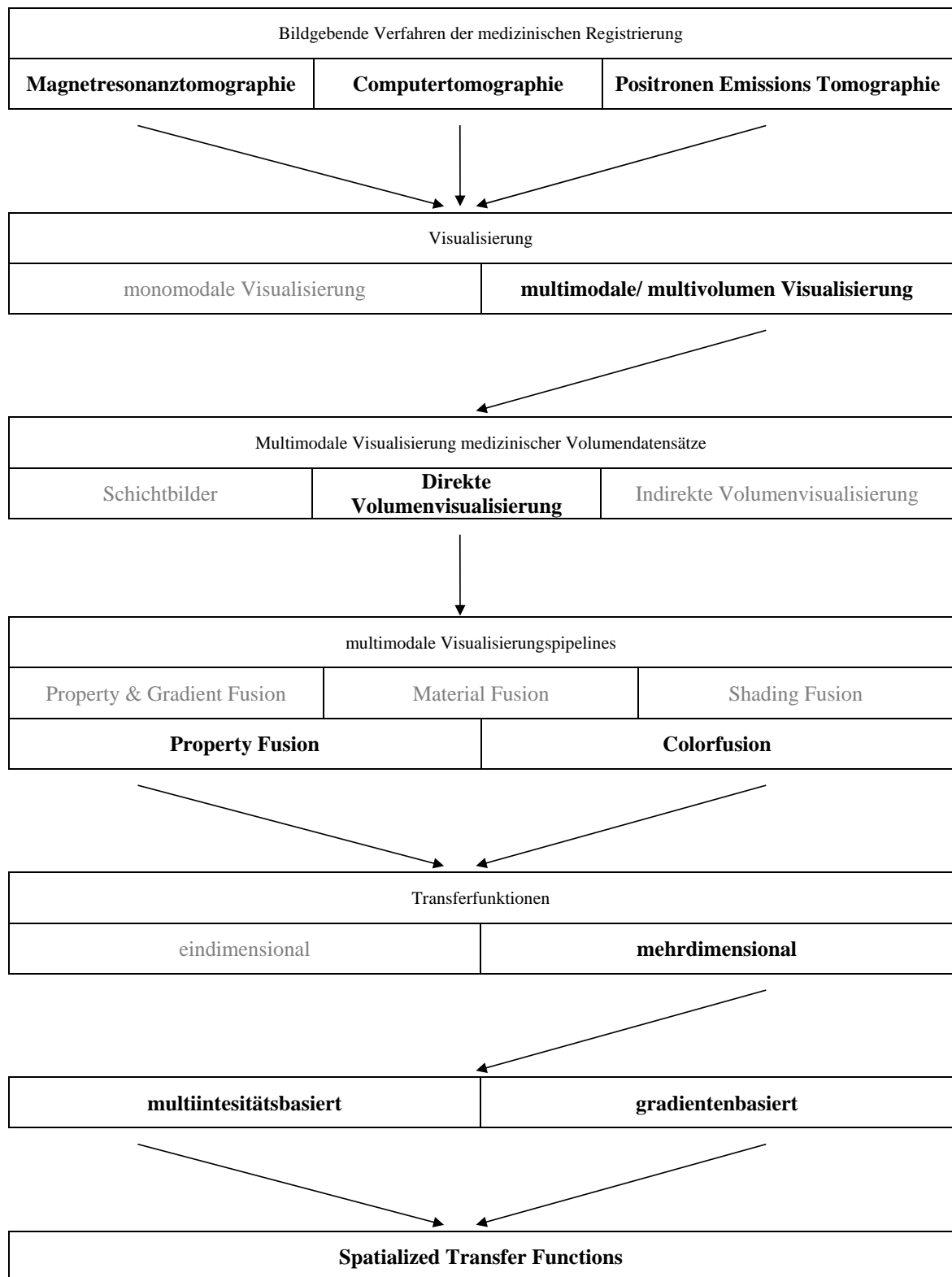


Abbildung 5.1.: Darstellung der ausgewählten Visualisierungsmöglichkeiten.

5.2. Spatialized Transfer Functions für die multimodale Volumenvisualisierung

Die Wahl der beiden Fusionszeitpunkte, einmal am Anfang und einmal am Ende der Visualisierungspipeline, haben dazu geführt, dass zwei Ansätze für die Spezifikation der TF definiert und umgesetzt werden mussten. Es sollte zum einen die Möglichkeit gegeben sein, eine gemeinsame TF für die beiden Datensätze zu erstellen. Diese basiert auf den Intensitätswerten der beiden Datensätze und wird im Folgenden als multiintensitätsbasierte TF bezeichnet. Zum anderen sollte der Nutzer für jeden Datensatz eine TF definieren können, die den Intensitätswert und die Gradienteninformation des jeweiligen Datensatzes als Merkmale nutzt (gradientenbasierte TF). Die in Abschnitt 4.4 vorgestellten STF wurden in das Visualisierungssystem eingebaut, da mit ihnen beide Ansätze umgesetzt werden können (vgl. Abschnitt 4.4.4 und [RBS05]). Die vom Nutzer definierten STF unterscheiden sich je nach Ansatz im Aufbau des Histogramms, das der Spezifikation zugrunde liegt.

Das 2D-Histogramm einer gradientenbasierten STF wurde in Abschnitt 4.4 bereits vorgestellt. Auf der x-Achse werden die Intensitäten des Datensatzes und auf der y-Achse die Gradientenbeträge abgetragen. Auch die multiintensitätsbasierten STF arbeiten mit 2D-Histogrammen, in denen allerdings auf beiden Achsen die Intensitätswerte der zu visualisierenden Datensätze abgetragen werden. Abbildung 5.2 zeigt den Unterschied zwischen den beiden Histogrammen.

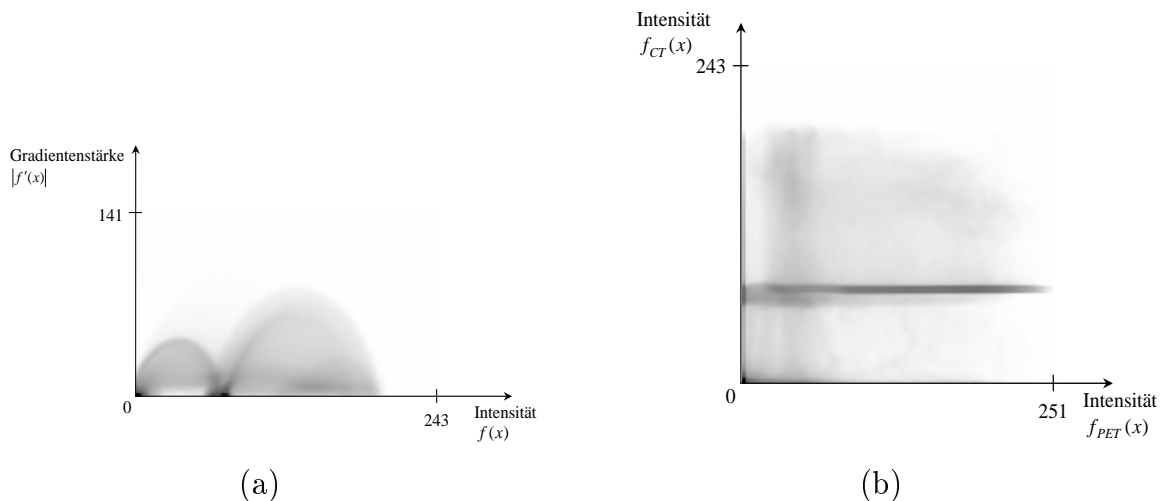


Abbildung 5.2.: Vergleich der Histogramme des gradientenbasierten und des multiintensitätsbasierten Ansatzes. Während das Intensitäts-/ Gradientenstärke-Histogramm (a) eines CT-Datensatzes die typischen Bogenstrukturen aufweist, ist im Intensitäts-/ Intensitäts-Histogramm (b) von CT- und PET-Daten keine regelmäßige Form erkennbar.

Die Berechnung der Ortsinformation und die Klassifikation des Histogrammes der multiintensitätsbasierten STFs erfolgt analog zu den gradientenbasierten STFs (vgl. Abschnitt 4.4). Ein Bereich im klassifizierten Histogramm einer multiintensitätsbasierten STF kennzeichnet somit eine zusammenhängende Region, die sich durch bestimmte Intensitätskombinationen in den beiden registrierten Datensätzen auszeichnet. Die Formen der klassifizierten Bereiche besitzen nicht die charakteristische Bogenstruktur wie die Bereiche der gradientenbasierten Histogramme (vgl. Abb. 5.3). Somit wird eine manuelle Definition der Bereiche, wie in [KKH01] vorgeschlagen erschwert, da kaum Anhaltspunkte für die Platzierung der Bereiche gegeben sind. Dem Nutzer wäre es somit nur schwer möglich, die Form der interessierenden Strukturen in einem multiintensitätsbasierten Histogramm manuell anzugeben. Die Spezifikation der multiintensitätsbasierten STF wird somit erst durch die automatische Klassifikation anhand der Ortsinformation ermöglicht.

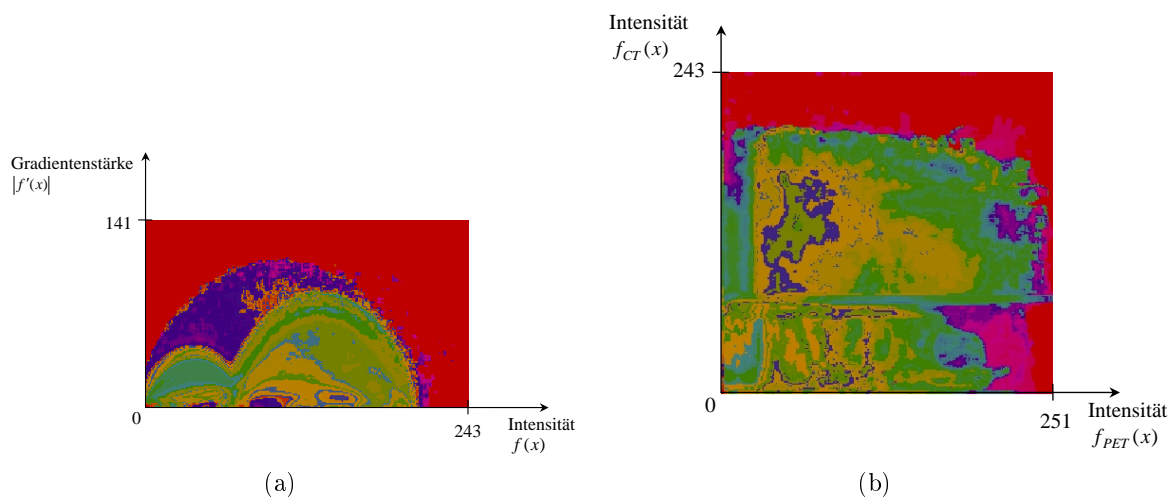


Abbildung 5.3.: Vergleich der klassifizierten Histogramme des gradientenbasierten und des multiintensitätsbasierten Ansatzes. Nach der Klassifikation ist erkennbar, dass die Form der Bereiche im Intensitäts-/ Intensitäts-Histogramm keiner Regelmäßigkeit folgt.

Der Ablauf der Spezifikation für die multimodale Volumenvisualisierung mit gradientenbasierten STFs verläuft analog zu dem in Abschnitt 4.4 erläuterten monomodalen Vorgehen. Allerdings wird für jeden der zwei Datensätze ein 2D-Histogramm separat berechnet und mit der entsprechenden Ortsinformation klassifiziert. Der Nutzer muss nun in jedem Histogramm die interessierenden Bereiche auswählen, woraufhin in der 3D-Visualisierung des Datensatzes die dazugehörige Struktur angezeigt wird. Die Visualisierung erfolgt getrennt für beide Datensätze. Dadurch ist es dem Nutzer möglich, zwischen den beiden Datensätzen zu wechseln und auch nachträglich weitere Strukturen zur Visualisierung hinzuzufügen. Hat der Nutzer die Definition der beiden gradientenbasierten STFs abgeschlossen, so kann er die beiden 3D-Visualisierungen zusammen im gleichen Ausgabebereich anzeigen und so durch eine *Color Fusion* eine multimodale Volumenvisualisierung erzeugen lassen. Auch bei der gemeinsamen Visualisierung ist es möglich, Strukturen aus den Datensätzen zur gemeinsamen Darstellung hinzuzufügen oder zu entfernen. Weiterhin kann wieder zur monomodalen Visualisierung gewechselt werden.

Mit den gradientenbasierten STFs ist es möglich die Transformation der Registrierung erst vor dem Schritt *Rendering* der Visualisierungspipeline durchzuführen. Dies ermöglicht eine schnelle Aktualisierung der Visualisierung, nachdem die Transformation durch den Nutzer geändert wurde, da die verwendete STF nicht wie im multiintensitätsbasierten Fall erneut spezifiziert werden muss, sondern weiterverwendet werden kann. Somit erfüllt die Visualisierung unter Verwendung gradientenbasierter STFs die Anforderung an die Visualisierungskomponente, eine Möglichkeit für die nachträgliche interaktive Veränderung der Transformation zu bieten. Ein weiterer Vorteil der gradientenbasierten STFs liegt in der Verwendung der Gradienteninformation. Sie ermöglicht es bei geeigneter Klassifikation sowohl die Oberfläche einer Struktur als auch das homogene Innere getrennt darzustellen.

Im multiintensitätsbasierten Ansatz wird für die multimodale Visualisierung zweier Datensätze nur ein 2D-Histogramm erstellt und klassifiziert. Der Vorteil der multiintensitätsbasierten STFs besteht in der Verringerung des Aufwands für den Nutzer, da dieser nur eine STF für die Erstellung einer multimodalen Visualisierung beider Datensätze definieren muss. In Kapitel 6 werden multimodale Visualisierungen der gleichen Datensätze, die mit den beiden zuvor genannten Ansätzen erstellt wurden, miteinander verglichen.

5.3. Implementierung der Visualisierungskomponente

Dieses Kapitel beschäftigt sich mit der Umsetzung des Entwurfs für ein multimodales Visualisierungssystem. Die zu entwickelnde multimodale 3D-Visualisierung sollte als Komponente in ein bestehendes Bildverarbeitungssystem eingegliedert werden. Diese Medical Imaging Platform (MIP) genannte Software wird am Fraunhofer IGD für Forschungsaufgaben verwendet. Die MIP wird in C++ geschrieben und basiert auf den drei im folgenden Abschnitt vorgestellten Programmbibliotheken. Die in Abschnitt 2.6 beschriebene Registrierung ist in der MIP als *Mutual Information* Registrierung bereits implementiert, so dass auf die damit berechneten Transformationsparameter bei der Visualisierung zugegriffen werden konnte.

5.3.1. Allgemeines

Programmierzwerkzeuge

Die Implementierung des Visualisierungssystems erfolgte ausschließlich mit frei verfügbaren Programmbibliotheken, die eine plattformunabhängige Implementierung der Software ermöglichen. Die implementierte Visualisierungskomponente kann in drei Bereiche aufgeteilt werden.

Dies ist zunächst die Programmoberfläche, über die der Nutzer die nötigen Einstellungen der Parameter vornehmen kann und die Visualisierung dargestellt wird. Für die Programmierung der Nutzeroberfläche wird *wxWidgets* in der Version 2.4.2 verwendet. Mit dieser objektorientierten Programmbibliothek wurden Standardeingaben und -ausgaben, wie z.B. Textfelder, Schieberegler und *Checkboxes* sowie die Interaktion durch den Nutzer implementiert. Daneben stellt sie auch eine Schnittstelle zur verwendeten Visualisierungsbibliothek zur Verfügung. Diese Schnittstelle ermöglicht den problemlosen Einbau der Visualisierungsfunktionalität in die Programmoberfläche.

Der zweite Bereich umfasst die Vorverarbeitung der Daten bis zur Generierung der Transferfunktionen. Hierzu wurde die von der Kitware Inc. angebotene Version 2.0.1 des *Insight Toolkit* (ITK) verwendet. Dieses stellt neben grundlegenden Datenstrukturen und sogenannten Adaptoren für die Umwandlung zwischen den Datenstrukturen auch umfangreiche Bildverarbeitungsfunktionen zur Verfügung. Eine sehr wichtige Eigenschaft des ITK ist der Pipeline-Aufbau und die Einhüllung der Funktionalität in sogenannte Filter. Diese verarbeiten ein Eingabebild und wenden darauf die implementierte Operation an, um ein Ausgabebild zu erzeugen. Sie werden daher im ITK auch von der Klasse `itk::ImageToImageFilter` abgeleitet. Alle im Folgenden verwendeten Filter und Objekte aus der ITK-Programmbibliothek sind durch ein `itk::` vor dem Namen gekennzeichnet. Für die Implementierung der *Spatialized Transfer Functions* mussten einige der bestehenden Filterklassen um zusätzliche Funktionalität erweitert bzw. vorhandene Funktionen angepasst werden. Auf solche Veränderungen wird in den betreffenden Abschnitten näher eingegangen. Filter, die für die Implementierung des Konzepts von ITK-Filtern abgeleitet wurden und modifizierte ITK-Filter haben ein `itk` vor dem Filternamen. In einer ITK-Filterpipeline werden Filter miteinander verkettet, indem die Ausgabe eines Filters direkt als Eingabe eines anderen Filters gesetzt wird. So ist es möglich,

komplexe Verarbeitungsabläufe zu implementieren und bei Bedarf weitere Filter hinzuzufügen oder vorhandene Filter zu entfernen. Während der Ausführung des Programms, wird nur der letzte Filter der Pipeline aufgerufen, woraufhin dieser rekursiv alle anderen Filter aufruft. Falls mehrere Prozessoren zur Verfügung stehen, wird die Verarbeitung der Daten durch einen Filter über ein sogenanntes *Multi Threading* parallelisiert und somit beschleunigt. ITK basiert auf sogenannten *Template* Parametern und erlaubt damit die Verarbeitung von Volumendatensätzen unterschiedlicher Datentypen. Beispielsweise ist ein `itk::Image` Objekt, welches für die Speicherung von Bilddaten verwendet wird über den Datentyp und die Dimensionalität definiert `itk::Image<Datentyp, Dimension>`. Durch sogenannte *Cast* Filter können Datentypen ineinander umgewandelt werden, wobei ggf. Informationsverluste entstehen können.

Im dritten Teil erfolgt die multimodale Visualisierung mit Hilfe der TFs. Das hierfür genutzte *Visualization Toolkit* (VTK, Version 4.4) basiert auf der Graphikbibliothek `OpenGLTM` und stammt ebenso wie ITK von Kitware Inc.. Über eine Schnittstelle zwischen den beiden Bibliotheken ist es möglich, die verarbeiteten Daten aus einer ITK-Filterpipeline zur Visualisierung in VTK-Datenstrukturen umzuwandeln und an VTK-Komponenten wie *Mapper* und *Renderer* zu übergeben. Die Generierung einer Visualisierung mit VTK folgt der in Abschnitt 3.3 vorgestellten allgemeinen Visualisierungspipeline. Über die Schnittstelle zu `wxWidgets` können die Daten dann in der Programmoberfläche dargestellt und mit ihnen interagiert werden. Mit `wxWidgets`, ITK und VTK wurde für jeden der drei Bereiche eine eigene Programmbibliothek verwendet, die auf die jeweiligen Zwecke spezialisiert ist.

Graphische Benutzerschnittstelle

Die Benutzeroberfläche für die multimodale Visualisierungskomponente lässt sich grob in drei Bereiche untergliedern. Das in Abbildung 5.4 rot umrandete Feld beinhaltet die Eingabemöglichkeiten für die Parametrisierung der Visualisierung und die numerischen Ausgaben, welche dem Nutzer als Feedback dienen. Im oberen Teil dieses Bereichs sind verschiedene *Checkboxes* (Kontrollkästchen) und *Buttons* (Schaltflächen) angeordnet. Diese dienen dazu den Aufbau der Filterpipeline zu bestimmen, die Histogrammklassifizierung zu starten („*Compute Histogram*“) oder die Ansicht des Histogramms zurückzusetzen („*Reset Camera*“). Die Schaltfläche „*Write Rendering*“ ermöglicht es, eine erstellte Visualisierung als Bild zu speichern. Weiterhin kann über den *Slider* (Schieberegeler) über der Histogrammdarstellung der Radius für die Klassifikation des Histogramms verändert werden. Die Darstellung des klassifizierten 2D-Histogramms innerhalb dieses Bereichs ermöglicht es dem Nutzer bestimmte Klassen auszuwählen (engl. *picking*), indem er mit der linken Maustaste ein Pixel anklickt und anschließend die Taste „s“ auf der Tastatur betätigt. Grundsätzlich sind der rechten und der mittleren Maustaste die Funktionen zoomen und verschieben zugeordnet. Unterhalb des 2D-Histogramms können die Histogramminformationen abgelesen werden. Dazu gehören die Größe des Histogramms, die Anzahl der Klassen und die numerischen Angaben zur aktuell ausgewählten Klasse. Weiterhin lässt sich für das dargestellte Volumen die globale Transparenz mit Hilfe des *Sliders* „*Opacity*“ einstellen und die gradientenbasierten Visualisierungen können durch die *Checkbox* „*Join Volumes*“ zu einer gemeinsamen Darstellung vereint werden.

Der blaue Bereich stellt die jeweiligen visualisierten Volumen dar, welche, wie bereits beschrieben, verschoben und vergrößert werden können. Der linken Maustaste ist in diesem Fall die Funktion zugewiesen worden die Darstellung zu rotieren, wodurch die Vorteile der 3D-Darstellung genutzt werden können. Oberhalb im grün umrandeten Teil sind die Schichtbilder des Ausgangsdatensatzes abgebildet, die der Nutzer mit Hilfe des dazugehörigen *Sliders* anwählen kann.

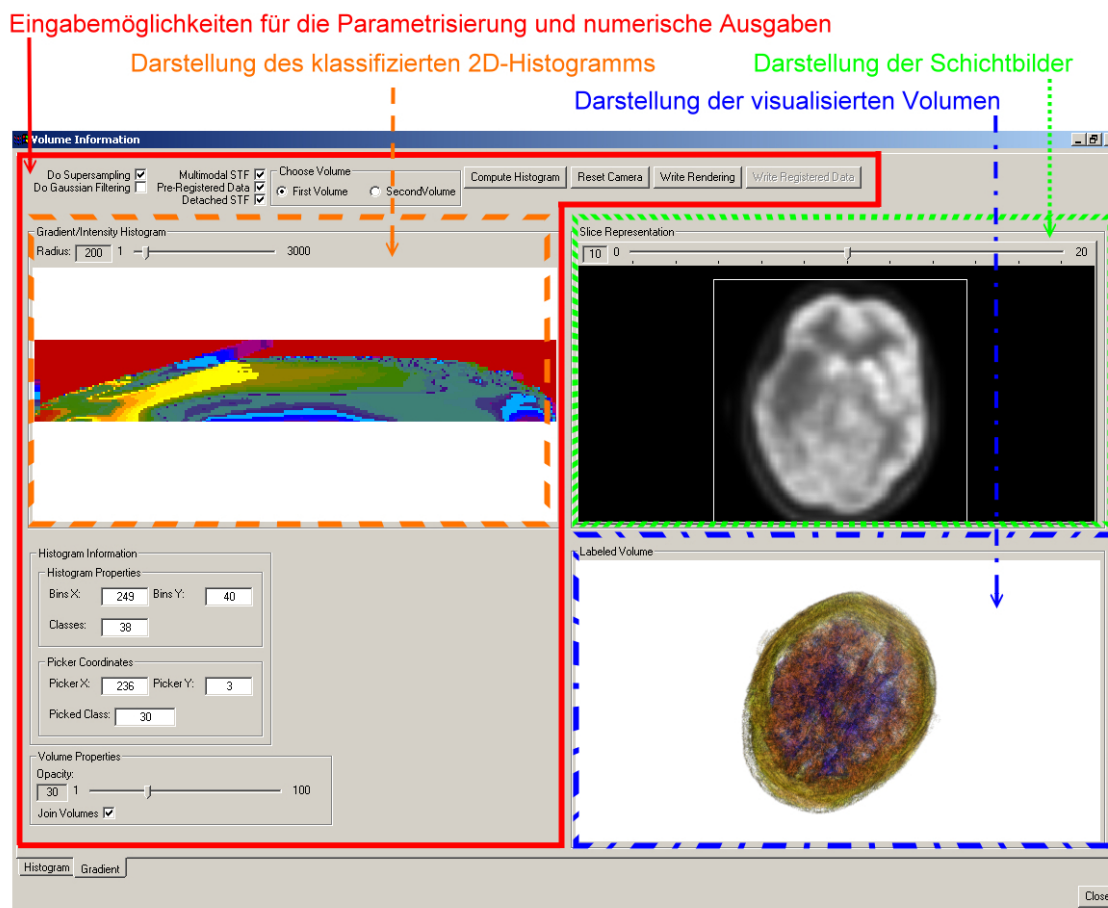


Abbildung 5.4.: Benutzeroberfläche der implementierten Volumenvisualisierungskomponente.

Die Benutzeroberfläche bietet die Möglichkeit zwischen den beiden implementierten Visualisierungsansätzen zu wählen und so den Programmablauf zu beeinflussen. Die erste Möglichkeit ist eine monomodale Darstellung. Dazu ist keine Checkbox-Auswahl zwingend erforderlich. Die Überabtastung und der Gaußfilter können jedoch in jedem Fall ausgewählt werden. Für eine multimodale Visualisierung mit einer STF (multiintensitäts-basierter Ansatz) muss zusätzlich die *Checkbox* „*Multimodal STF*“ ausgewählt werden. Die meisten Einstellungen müssen für den gradientenbasierten Ansatz (zwei STFs) vorgenommen werden. Zunächst ist neben der *Checkbox* „*Multimodal STF*“ auch „*Detached STF*“ („getrennte STF“) zu wählen. Über den *Radio Button* (Auswahlfeld) „*First Volume*“ wird das erste Volumen für die Visualisierung ausgewählt und nach der Berechnung („*Compute Histogram*“) können Klassen aus dem dazugehörigen Histogramm zur betreffenden STF hinzugefügt werden. Die resultierende Visualisierung wird anschließend im

betreffenden Bereich der Oberfläche dargestellt. Das gleiche geschieht nach Auswahl des „*Second Volume*“ und anklicken der Schaltfläche „*Compute Histogram*“ für den zweiten Datensatz. Die Darstellungen der Schichtbilder, des Histogramms und der Visualisierung erfolgen dann für den zweiten Datensatz. Wurden für beide Datensätze separate Visualisierungen erstellt, können diese über die *Checkbox* „*Join Volumes*“ vereinigt und gemeinsam angezeigt werden. Auch die resultierende multimodale Visualisierung kann als Bild gespeichert werden. Die Funktionalität wird anhand von Videosequenzen auf der beiliegenden CD noch einmal verdeutlicht.

Programmablaufpläne

In den folgenden Abschnitten wird die Implementierung der Visualisierungskomponente anhand der gradientenbasierten STFs detailliert beschrieben. Der multiintensitätsbasierte Ansatz unterscheidet sich in folgenden Punkten vom gradientenbasierten Ansatz.

- Anwendung der Transformation und Berechnung der überschneidenden Regionen der Volumendatensätze bereits vor der Vorverarbeitung unter Verwendung von trilinear Interpolation (vgl. Abschnitt 5.3.6)
- Keine Berechnung der Gradientenstärke
- Aufbau eines Datensatzes mit zwei Intensitätswerten aus den beiden zu visualisierenden Volumen
- Generierung lediglich eines Intensitäts-/ Intensitäts-Histogramms
- Berechnung der Ortsinformation lediglich einmal durchgeführt
- Generierung und Anwendung lediglich einer STF

Um die Programmschritte zu veranschaulichen und die Unterschiede zwischen den beiden umgesetzten Ansätzen zu verdeutlichen wurden zwei Programmablaufpläne erzeugt. In der Abbildung 5.5 wird der gradientenbasierte Ansatz unter Verwendung von zwei TFs dargestellt. Abbildung 5.6 zeigt den multiintensitätsbasierten Ansatz unter Verwendung einer STF als Farbtransferfunktion (FTF).

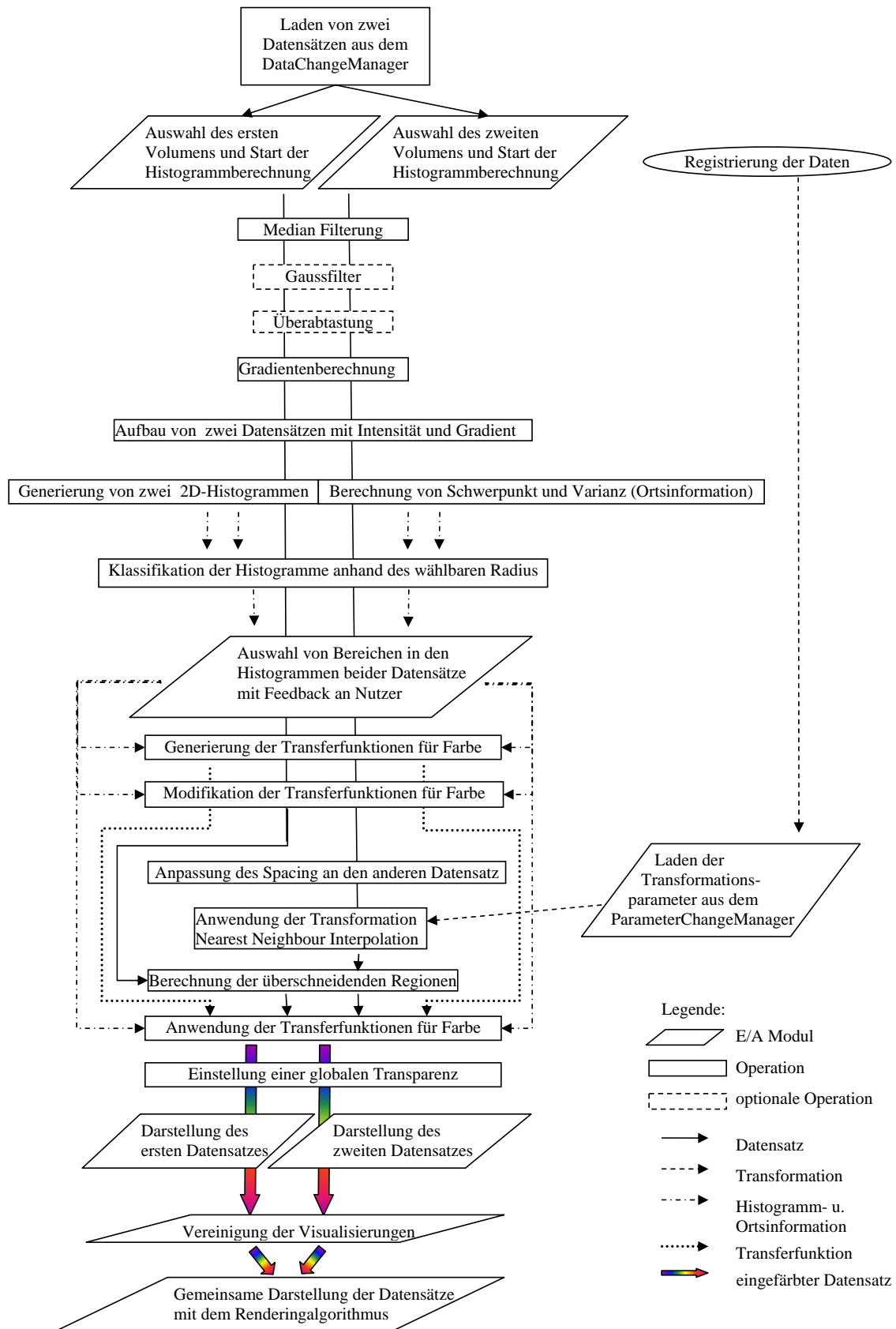


Abbildung 5.5.: Darstellung des Programmablaufs für die gradientenbasierte STF Generierung unter Verwendung von zwei Transferfunktionen.

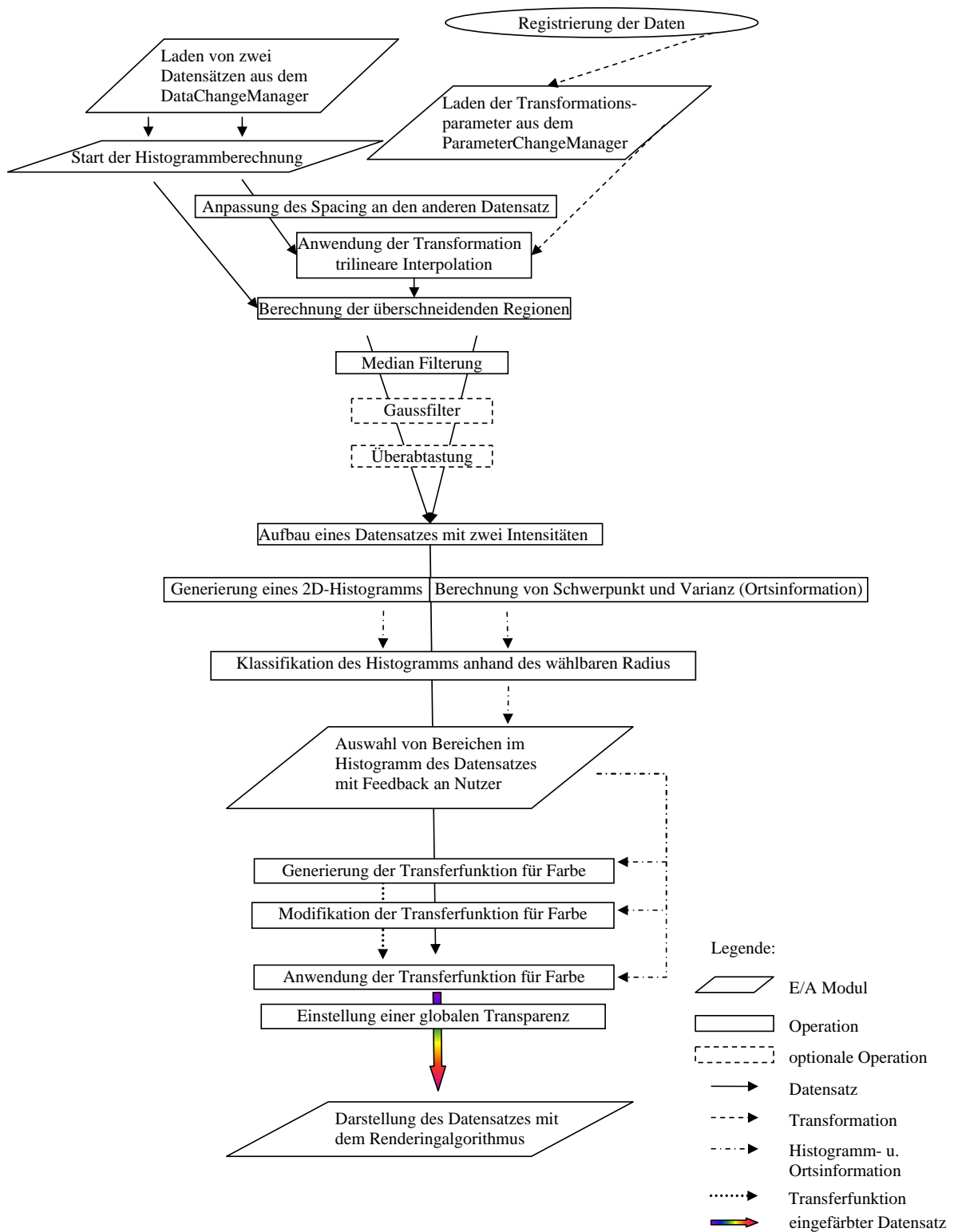


Abbildung 5.6.: Darstellung des Programmablaufs für die multiintensitätsbasierte STF Generierung unter Verwendung einer Transferfunktion.

Verwendung zweidimensionaler Transferfunktionen im VTK

In der VTK Version 4.4, welche für die Implementierung verwendet wurde, ist es nicht möglich mehrdimensionale Transferfunktionen an den Renderingalgorithmus zu übergeben. Die dafür notwendigen Erweiterungen der zugrunde liegenden Graphikbibliothek OpenGLTM wurden in dieser Version noch nicht integriert. Ein ähnliches Problem wird in [GK96] beschrieben. Für die Lösung des Problems wird vorgeschlagen die mehrdimensionale TF in mehrere eindimensionale TFs aufzuspalten und diese für das Rendering zu nutzen. Aufgrund der Eigenschaften der in dieser Arbeit verwendeten STFs kann dieser Lösungsansatz nicht verfolgt werden. Dies soll an einem einfachen Beispiel aus [KPI⁺03] in Abbildung 5.7 verdeutlicht werden. Durch die verwendete automatische Klassifikation und die daraus resultierenden Formen der gewählten Bereiche sind die STFs nicht linear und somit auch nicht separabel. Aus diesem Grund musste die Anwendung der zweidimensionalen STFs innerhalb der Software vor dem Rendering implementiert werden.

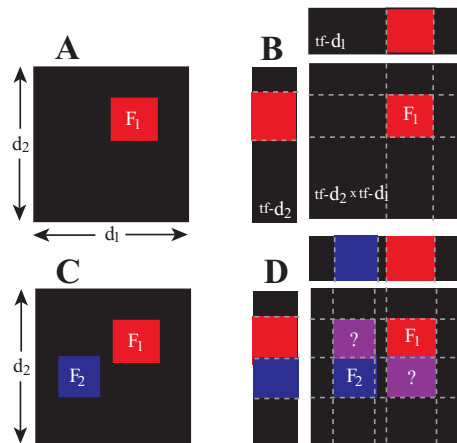


Abbildung 5.7.: Die in A dargestellte 2D-TF mit einem rechteckigen Wertebereich, welcher die betreffenden Voxel auf die Farbe rot abbildet, kann wie in B dargestellt in zwei 1D-TFs aufgespalten werden. Die Anwendung der beiden 1D-TFs für die Darstellung eines Datensatzes entspricht dem Produkt der beiden TFs. Wird die 2D-TF, wie in C gezeigt, um ein weiteres Merkmal erweitert, so ist die Separierbarkeit nicht mehr gegeben. Die Multiplikation der beiden resultierenden 1D-TFs ergibt nicht mehr die ursprüngliche 2D-TF. Hieraus ergibt sich eine falsche Abbildung des Datensatzes. Die Klassifikation des Wertebereichs der STFs ist noch weitaus komplexer (s. Abb. 5.3), weshalb eine Softwareimplementierung der TF gewählt wurde. Quelle: [KPI⁺03].

Für die Anwendung einer STF innerhalb der Software muss nach der Spezifikation und Modifikation der STF über das gesamte darzustellende Volumen iteriert und jeder Voxel anhand der resultierenden STF eingefärbt werden. Für die Einfärbung des Histogramms und der Visualisierung wurden in dieser Arbeit 1D-FTFs verwendet. Diese sind über einen ganzzahligen Wert, die sogenannte Klassennummer definiert. Die Klassennummern werden bei der Klassifikation der 2D-Histogramme an die klassifizierten

Bereiche vergeben, wobei alle Bins eines Bereichs die gleiche Klassennummer besitzen (vgl. Abschnitt 5.3.3). Wird ein Bereich ausgewählt und so die STF definiert bzw. modifiziert, muss allen Voxeln des zu visualisierenden Volumens, welche in die Bins des ausgewählten Bereichs fallen, die Klassennummer als Intensitätswert zugewiesen werden (vgl. Abschnitt 5.3.5). Anschließend wird das modifizierte Volumen mit der 1D-FTF an den Renderer übergeben (vgl. Abschnitt 5.3.7).

Eine Iteration über das Volumen ist jedoch für eine schnelle Aktualisierung der Visualisierung nicht anwendbar. Aus diesem Grund wurde eine Datenstruktur für den direkten Zugriff auf die Voxel eines Bins erstellt. Hierfür wurde eine `std::multimap` verwendet. Die `std::multimap` ist ein ausgeglichener Binärbaum, dessen Elemente über zwei Parameter definiert sind (vgl. [Mul]). Anhand des ersten Parameters können die Elemente der Datenstruktur gruppiert werden. Der zweite Parameter beschreibt die Eigenschaft der einzelnen Elemente. Für die Speicherung der Voxelkoordinaten, welche in ein Bin des Histogramms fallen, wurde eine `std::multimap` über einen `unsigned short` und einen `unsigned int` Wert definiert. Der erste Wert entspricht der Position des Bins im Histogramm und wird wie folgt berechnet:

$$Position_{Bin} = (BinIndex_y * AnzahlBins_x) + BinIndex_x \quad (5.1)$$

Der zweite Wert beschreibt die Position des Voxels, welches dem Bin zugeordnet wird und ergibt sich wie folgt:

$$Position_{Voxel} = (VoxelIndex_z * VolumeSize_y * VolumeSize_x) + (VoxelIndex_y * VolumeSize_x) + VoxelIndex_x \quad (5.2)$$

Indem die Positionen über eine ID und nicht den Index gespeichert werden, wird der Speicherbedarf reduziert. Neben den beiden Parametern muss die Art der Sortierung der Elemente in der Baumstruktur angegeben werden. Diese ist über den ersten Parameter definiert und in dieser Arbeit als aufsteigende Sortierung implementiert. Durch die Sortierung ist ein effizienter Zugriff auf die Elemente einer Gruppe möglich. Für die Aktualisierung der Visualisierung wird nun für jedes Bin des gewählten Bereiches über dessen Elemente iteriert und den betreffenden Voxeln die Klassennummer als Wert zugewiesen. Somit ist ein direkter Zugriff auf die zu verändernden Voxel möglich. Die Implementierung über eine `std::multimap` beschleunigt die Spezifikation der STF sehr stark, da die Auswahl von Bereichen aus dem klassifizierten Histogramm ein häufig ausgeführter Arbeitsschritt ist, dessen Zeitaufwand die Dauer der Generierung einer Visualisierung neben der Vorverarbeitung am stärksten beeinflusst.

Als Beispiel soll ein Volumendatensatz mit 369 x 267 x 131 Voxeln auf einem PC mit 2 Pentium4 Prozessoren, jeweils mit 1,1 GigaHertz Taktrate und 2GB RAM visualisiert werden. Die Vorverarbeitung des Datensatzes dauert ohne die Generierung der `std::multimap` 24,9 Sek. und die Auswahl eines Bereiches mit anschließender Aktualisierung des Volumens dauert 95,5 Sek. (inkl. Rendering des Volumens). Die lange Aktualisierungszeit ergibt sich, da bei der Iteration über das Volumen nur einer der beiden Prozessoren verwendet wird. Wird eine `std::multimap` bei der Histogrammgenerierung aufgebaut, dann erhöht sich die Vorverarbeitungszeit auf 47,4 Sek.. Die Zeit für die Auswahl eines Bereiches aus dem Histogramm und der Aktualisierung der Vi-

sualisierung verringert sich allerdings auf 1,2 Sek. (inkl. 0,9 Sek. für das Rendering des Volumens). Demnach rentiert sich in diesem Beispiel die erhöhte Vorverarbeitungszeit bereits nach der Auswahl des ersten Bereiches. Bei kleineren Datensätzen (z.B. 256 x 256 x 21) verringert sich die Zeit für die Aktualisierung von 9 Sek. auf 0,3 Sek. wobei die Vorverarbeitungszeit von 4 Sek. auf 8 Sek. ansteigt.

Mit der Wahl der `std::multimap` wurde demnach eine effiziente Lösung für das Problem der schnellen Aktualisierung der Visualisierung nach Modifikation der STF gefunden. Der Hauptnachteil der `std::multimap` liegt im hohen Speicherverbrauch. Neben den 6 Byte im Speicher für jedes Voxel des Volumens wird sehr viel Speicher für die Verwaltung der Datenstruktur belegt.

5.3.2. Vorverarbeitung der Daten

Die Vorverarbeitung der Daten verfolgt zwei Ziele. Zum einen soll die Qualität des Histogramms, das für die Spezifikation einer STF erzeugt wird, verbessert werden. Zum anderen wird die Qualität des Datensatzes und damit der resultierenden Visualisierung optimiert. Auf den Umfang der Vorverarbeitung kann der Nutzer Einfluss nehmen, indem er angibt, welche Filter in die ITK-Filterpipeline aufgenommen werden sollen. Die Verarbeitung der Daten erfolgt anschließend automatisch, da die Reihenfolge der verwendeten Filter sowie deren Parametrisierung vorgegeben sind.

Die Vorverarbeitung umfasst in jedem Fall eine Medianfilterung, um das Rauschen in den Datensätzen zu reduzieren. Der Nutzer kann entscheiden, ob anschließend noch eine Gaußfilterung und eine Überabtastung der Daten vorgenommen werden sollen. Die anschließende Berechnung der Gradientenstärke erfolgt nur im gradientenbasierten Ansatz und wird bei der multiintensitätsbasierten STF Generierung ausgelassen. Auf die Schritte der automatischen Vorverarbeitung und die damit erreichten Verbesserungen wird im Folgenden näher eingegangen.

Laden der Daten

Das Einladen der Volumendatensätze erfolgt über die MIP. Die Daten werden anschließend im sogenannten `MIPDataChangeManager` in einem eigens für die MIP entwickelten Datenformat vorgehalten. Aus diesem `MIPDataChangeManager` wird beim Start der Visualisierungskomponente eine Kopie des ersten Datensatzes in Form eines dreidimensionalen `itk::Image` erzeugt. Dies ist der Standard Datentyp für die Speicherung von Bildinformationen im ITK und gleichzeitig die Eingabe für die Filterpipeline. Neben den eigentlichen Datenwerten, werden im `itk::Image` auch die Metainformationen, wie Auflösung, Voxelgröße und die durch die Filter zu verarbeitende Region des Bildes gespeichert.

Indem zunächst nur ein Datensatz eingeladen wird, wird die Visualisierung von Datensätzen, denen keine weiteren registrierten Volumen zugeordnet sind, ermöglicht. Wenn der Nutzer die Checkbox für die Spezifikation einer multimodalen TF auswählt, wird der zweite Datensatz aus dem `MIPDataChangeManager` kopiert und in ein `itk::Image` umgewandelt. Die beiden Volumendatensätze bilden die Grundlage für die Spezifikation der STFs und werden zunächst von der ITK-Filterpipeline verarbeitet. Im beschriebenen

gradientenbasierten Ansatz erfolgt die Verarbeitung separat, wobei der zu verarbeitende Datensatz über die „Choose Volume“ Auswahl auf der Programmoberfläche bestimmt wird. Im Anschluss wird die Berechnung für den gewählten Datensatz durch die „Compute Histogram“ Schaltfläche gestartet (s. Abb. 5.4). Im multiintensitätsbasierten Ansatz durchlaufen die beiden Volumendatensätze die Filterpipeline parallel.

Medianfilterung

Eine Medianfilterung zur Reduktion des Rauschens bildet die erste Stufe der ITK-Filterpipeline und ist in der ITK-Klasse `itk::MedianImageFilter` implementiert. Dem Filter muss das Eingabebild, sowie die Größe der zu verwendenden Filtermaske übergeben werden. Für die Filterung wird eine $3 \times 3 \times 1$ Filtermaske verwendet. Die Filterung findet demnach in einer 8er-Nachbarschaft ausschließlich innerhalb der Schichtbilder statt, es wird nicht über die Schichten hinweg gefiltert. Eine Filterung über die Schichten würde die Bildinformation innerhalb einer Schicht aufgrund der anisotropen Voxelausdehnung in den Dimensionen verwaschen und so die Qualität des resultierenden Datensatzes verschlechtern. Die verwendete Filtermaske ist in Tabelle 5.1 abgebildet.

f_1	f_2	f_3	\times	27	28	32	\implies	27	28	32
f_4	f_5	f_6		26	50	30		26	28	30
f_7	f_8	f_9		25	27	29		25	27	29

Tabelle 5.1.: Links: Median Filtermaske des verwendeten $3 \times 3 \times 1$ Median Filters. Mitte: Beispiel mit 9 Grauwerten und einem zu hohen Intensitätswert, der sich durch Rauschen ergibt. Rechts: Der Verrauschte Voxelwert wurde korrigiert.

Der Medianfilter sortiert die neun Grauwerte f_1, f_2, \dots, f_9 der Voxel aus der 8er-Nachbarschaft aufsteigend. Der Wert des Voxels in der Mitte der Filtermaske f_5 wird durch den Grauwert in der Mitte der Sortierung $f_{\lfloor 9/2 \rfloor}$ ersetzt.

An den Bildrändern wird bei der Medianfilterung die sogenannte Neumann-Regel angewendet. Diese liefert für Punkte ausserhalb des Bildes die Werte der nächstliegenden Bildpunkte. Die Laufzeit des Medianfilters erhöht sich linear mit der Bildgröße und der Größe der verwendeten Filtermaske. Der Medianfilter wurde in dieser Arbeit verwendet, da er die Kanten in den Bilddaten erhält.

Gaußfilterung

Falls vom Nutzer gewünscht, können die Datensätze nach der kantenerhaltenden Medianfilterung noch mit einem Gaußfilter gefaltet werden. Dazu muss auf der Programmoberfläche die *CheckBox* „Do Gaussian Filtering“ aktiviert werden (s. Abb. 5.4). Dies bietet sich bei Datensätzen mit sehr starkem Rauschen sowie CT-Datensätzen die „hart“ rekonstruiert wurden an (vgl. Abschnitt 2.5.1). Die Gaußfilterung verbessert in diesem Fall die Qualität der später generierten gradientenbasierten 2D-Histogramme. Zum einen wird durch die Rauschreduktion die Anzahl der Voxel mit hohem Gradienten reduziert, welche in der späteren Darstellung zu Artefakten (Rauschen) führen können. Zum anderen werden die „hart“ rekonstruierten Kanten, die nur durch relativ wenige Voxel repräsentiert werden, geglättet. Durch das verschmieren der Gradienteninformation werden mehr

Voxel einer Kante zugeordnet. Dies bewirkt eine glattere Form der klassifizierten Bereiche und in der gradientenbasierten Visualisierung werden nach Auswahl der Bereiche, die eine Kante repräsentieren (hoher Gradient), mehr Voxel angezeigt.

Für die Gaußfilterung wurde der `itk::DiscreteGaussianImageFilter` in die ITK-Filterpipeline eingebunden. Durch die direkte Verknüpfung der Filter über ihre Eingabe- und Ausgabebilder ermöglicht die ITK-Filterpipeline das Einhängen des `itk::DiscreteGaussianImageFilter` je nach Bedarf. Der verwendete Gaußfilter arbeitet in Weltkoordinaten, d.h. er beachtet die Voxelausdehnung bei der Berechnung. Er wird über die Varianz σ und einen maximal zulässigen Fehler bei der Annäherung der Gauß-Dichtefunktion parametrisiert. Um durch den Verwischungseffekt der Gaußfilterung keine wichtige Bildinformation (z.B. schwache Kanten) zu verlieren, wurde die Varianz für diese Arbeit auf 0,1 festgelegt und der maximale Fehler liegt bei 0,01. Aus der Varianz und der Voxelausdehnung wird anschließend über die Dichtefunktion der dreidimensionalen Gauß-Verteilung die verwendete Filtermaske G berechnet.

$$G(x, y, z) = \frac{1}{(2\pi)^2 \sigma^3} e^{-\frac{x^2+y^2+z^2}{2\sigma^2}} \quad (5.3)$$

Dabei wird ausgenutzt, dass die 3D-Gauß-Dichtefunktion in drei 1D-Funktionen zerlegt werden kann. Diese werden jeweils zu einer 1D-Filtermaske diskretisiert. Die Filtermaske wird dabei so lange vergrößert, bis der Fehler, der bei der Annäherung an die Gaußverteilung gemacht wird, geringer als der zulässige Fehler ist. Die Größe der Filtermaske variiert demnach für unterschiedliche Datensätze und bei anisotropen Daten auch über die drei Raumdimensionen. Durch die Vorgabe der Varianz σ und keiner konkreten Filtermaske erfolgt die Gaußfilterung angepasst an die Daten und somit relativ gleichmäßig für alle Datensätze. Die verwendeten 3D-Filtermasken haben meist eine Größe von $3 \times 3 \times 3$ bis $5 \times 5 \times 5$ Voxeln, je nach Voxelausdehnung im Datensatz. Auch die Gaußfilterung nutzt an den Bildrändern die Neumann-Regel. Da bei der Gaußfilterung nicht ganzzahlige Datenwerte entstehen können, müssen die Eingabebilder zunächst vom ITK-Datentyp `itk::Image<unsigned short, 3>` in `itk::Image<double, 3>` umgewandelt werden. Nach der Filterung werden die Ausgabebilder wieder in `unsigned short` umgewandelt. Für die Änderung des Datentyps wird der `itk::CastImageFilter` verwendet. Bei der Umwandlung von `double` in `unsigned short` entstehen Diskretisierungsfehler, die aber aufgrund der kürzeren Verarbeitungszeit und dem geringeren Speicherbedarf von `unsigned short` Daten toleriert werden können.

Überabtastung

Eine große Schichtdicke wirkt sich negativ auf die Qualität der Visualisierung aus. Weiterhin kann durch eine geringe Auflösung des Datensatzes die im abgebildeten Objekt vorhandene Kanteninformation verloren gehen oder die Kante bei der Abtastung (vgl. Abschnitt 2.5.1) gar nicht erfasst werden. Eine zu geringe Anzahl von Voxeln, die in das Histogramm eingetragen werden, verschlechtert außerdem die Qualität des Histogramms und hat negativen Einfluss auf die spätere Klassifikation. Die Überabtastung wirkt diesen Effekten etwas entgegen, soll den Partialvolumeneffekt in den Daten ausgleichen und die Visualisierung optisch aufwerten. Der Nutzer kann daher im Anschluss

an die Median- und optionale Gaußfilterung eine Überabtastung der Daten, durch Auswahl der entsprechenden *CheckBox* „*Do Supersampling*“ auf der Programmoberfläche, in die ITK-Filterpipeline einhängen (s. Abb. 5.4). Die Überabtastung wird mit dem `itk::ResampleImageFilter` realisiert.

Im Falle einer Überabtastung wird die Voxelgröße der Datensätze in jede Raumrichtung halbiert und die Anzahl der Voxel verdoppelt. Daraus resultiert eine Verachtfachung der Datenmenge, wodurch sich der Speicherbedarf und der Rechenaufwand drastisch erhöhen. Die Überabtastung verbessert allerdings im Gegenzug auch maßgeblich die Qualität der zu erzeugenden 2D-Histogramme und deren Klassifikation sowie der Visualisierung. Die notwendige Interpolation erfolgt trilinear. Da bei ungerader Pixelanzahl oder ungerader Voxelgröße Randvoxel ohne Bildinformation entstehen können, wird dem `itk::ResampleImageFilter` ein *DefaultPixelValue* von 0 übergeben, welcher diesen Voxeln zugewiesen wird. Der Effekt der Überabtastung auf das klassifizierte Histogramm wird aus Abbildung 5.8 deutlich.

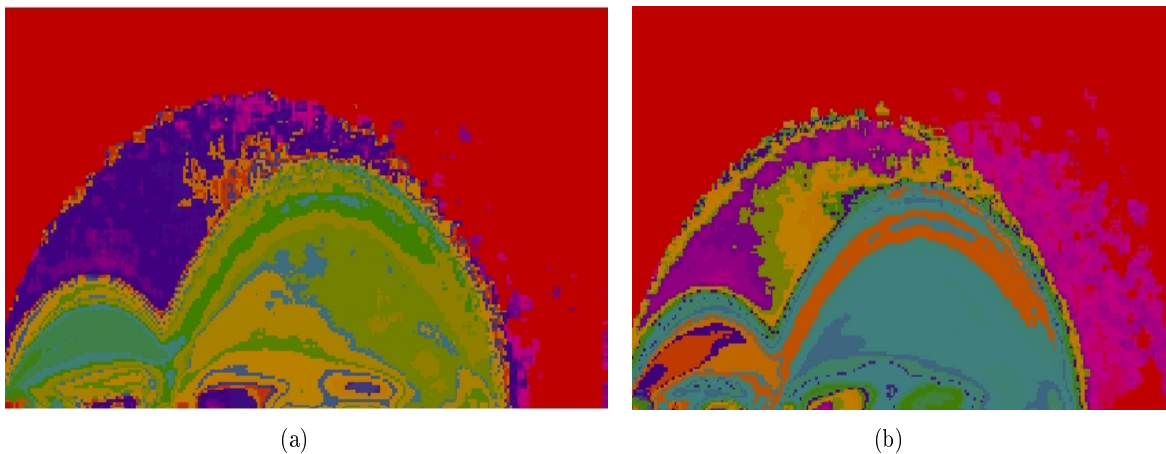


Abbildung 5.8.: Auswirkung der Überabtastung auf das klassifizierte Histogramm eines CT-Datensatzes. Durch die Erhöhung der Voxelanzahl werden die klassifizierten Bereiche im Histogramm des überabgetasteten Datensatzes (b) homogener und neue Teile des Histogramms werden bei der Klassifikation erschlossen (rosafarbene Bereiche auf der rechten Seite in (b)).

Berechnung der Gradientenstärke

Für die gradientenbasierten STFs ist die Berechnung der Gradienteninformation aus den Bilddaten ein notwendiger Schritt. Mit Hilfe der Gradientenstärke und den Intensitätswerten der Voxel der beiden Datensätze können die zwei benötigten 2D-Histogramme berechnet werden. Die Berechnung der Gradienteninformation erfolgt über den `itk::GradientMagnitudeImageFilter`, welcher im gradientenbasierten STF-Ansatz in die ITK-Filterpipeline eingehängt wird. Um die Berechnung des Gradientenbetrags zu standardisieren, erfolgt diese unter Beachtung der Voxelausdehnung (vgl. Abschnitt 4.3.3). Die Berechnung des Gradientenbetrags ist im verwendeten ITK-Filter, wie in Abschnitt 4.3.3 beschrieben, über die drei Raumdimensionen durch zentrale Differenzen realisiert. Das

Ergebnis der Gradientenfilterung sind zwei weitere Volumen mit den gleichen Abmessungen wie die verarbeiteten Datensätze. In jedem Voxel wird die Gradientenstärke als `unsigned short` Wert gespeichert. Nach der Berechnung der Gradientenstärke liegen somit neben den beiden Ausgangsdatsätzen zwei weitere Volumen mit der Gradienteninformation vor.

5.3.3. Generierung der 2D-Histogramme und der Ortsinformationen

An diesem Punkt der Visualisierungspipeline liegen im gradientenbasierten Ansatz vier Volumendatsätze vor, aus denen zwei Intensitäts-/ Gradientenstärke-Histogramme berechnet werden. Im Falle der multiintensitätsbasierten STFs liegen lediglich zwei bereits miteinander registrierte Volumendatsätze mit den gemessenen Datenwerten vor (vgl. Abschnitt 5.3.6), mit denen ein Intensitäts-/ Intensitäts-Histogramm aufgefüllt wird.

Generierung der 2D-Histogramme unter Verwendung der k-Nachbarschaft

Für jeden Datensatz erfolgt in diesem Abschnitt der Visualisierungspipeline eine parallele Berechnung des 2D-Histogramms und der Ortsinformation, welche für die Klassifikation des Histogramms benötigt wird. Die beiden Arbeitsschritte werden im Folgenden für einen Datensatz erläutert. Für die Implementierung wurde eigens der `itkHistogram2DGeneratorFilter` von der Klasse `itk::ImageToImageFilter` abgeleitet. Neben der Überabtastung des Datensatzes hat sich als vorteilhaft herausgestellt, die Anzahl der Bins je Histogrammrichtung an die im Volumen vorkommenden Intensitäts- und Gradientenwerte anzupassen. Dies verhindert die Bildung von Artefakten wie z.B. leeren Bins im Histogramm (vgl. Abschnitt 4.3.3). Im `itkHistogram2DGeneratorFilter` wird die Größe des erzeugten 2D-Histogramms an die zugrunde liegenden Daten angepasst und auf maximal 256 Bins in jede Richtung beschränkt. Für die Anpassung werden die maximale Intensität f_{max} des Datensatzes und im gradientenbasierten Fall die maximale Gradientenstärke g_{max} ermittelt. Bei der Verarbeitung von 8-Bit Daten sind diese Maximalwerte häufig kleiner als 256. In diesem Fall wird die Anzahl der Bins auf die ermittelten Maximalwerte gesetzt. Diese Verringerung der Anzahl der Bins verhindert das Auftreten von Streifenartefakten im Histogramm. Die Streifen entstehen durch Bins, welche Wertekombinationen repräsentieren, die nicht in den Volumen auftreten und daher leer bleiben. Weiterhin wird durch die Anpassung der Histogrammgröße die Klassifikation des Histogramms verbessert, da einem Bin mehr Voxel zugeordnet werden und somit die statistische Aussagekraft der Ortsinformation steigt. Der Effekt der Anpassung wird in Abbildung 5.9 verdeutlicht.

Nachdem die Anzahl der Bins des Histogramms festgelegt ist, werden die Grenzen des jeweiligen Wertebereichs, den ein Bin repräsentiert, berechnet. Diese Information wird anschließend dem `itkListSampleToSTFHistogramGenerator` für die Generierung des 2D-Histogramms übergeben. Dieser Generator wurde von der Klasse `itk::ListSampleToHistogramGenerator` abgeleitet. Der entscheidende Unterschied zum ITK-Filter ist die Erweiterung um die Berechnung von Schwerpunkt und Varianz in einem Arbeitsschritt parallel zur Histogrammgenerierung.

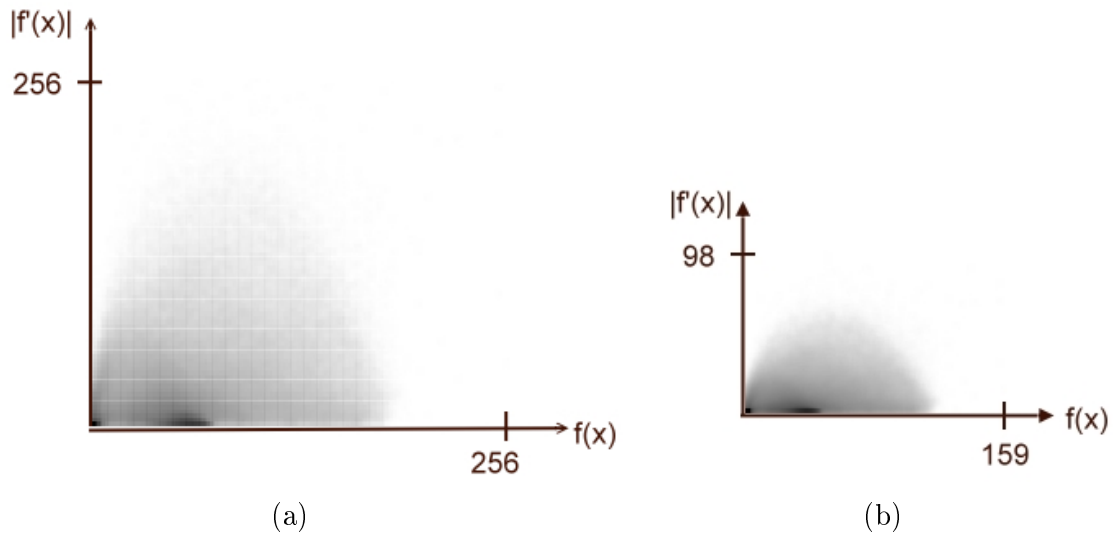


Abbildung 5.9.: Auswirkung der Anpassung der Histogrammgröße anhand eines MRT-Datensatzes. Durch die Anpassung (b) wird die Anzahl der Bins reduziert und Streifenartefakte wie in (a) werden eliminiert.

Für die Verarbeitung durch die ITK-Histogrammgeneratoren ist die Vereinigung der beiden Datensätze zu einem Volumen mit Intensitäts- und Gradientenwert je Voxel notwendig. Im multiintensitätsbasierten Ansatz wird an dieser Stelle ein Volumen mit zwei Intensitätswerten je Voxel generiert. Für die Vereinigung der Datensätze wird der sogenannte `itk::ScalarToArrayCastImageFilter` verwendet. Weiterhin können die Histogrammgeneratoren die Bilddaten nur in Form von Listen verarbeiten. Für diese Umwandlung wurde mit dem `itkSTFImageToListAdaptor` eine modifizierte Version des `itk::ImageToListAdaptor` implementiert, welcher in der Lage ist, die vereinten Volumendaten zu verarbeiten. Die entstehende Liste wird anschließend in das 2D-Histogramm abgetragen.

Die Wertekombinationen werden, wie in Abschnitt 4.3.3 beschrieben, durch den `itkListSampleToSTFHistogramGenerator` in das dazugehörige Bin abgetragen. Das jeweilige Bin wird in der ITK-Umsetzung über eine Binärsuche ermittelt. Im `itkListSampleToSTFHistogramGenerator` erfolgt auch der Aufbau der am Ende des Abschnitts 5.3.1 beschriebenen `std::multimap`. Dieser werden Elemente zugewiesen, welche als Eigenschaften die Position des Bins im Histogramm und die Position des aktuellen Voxels im Volumendatensatz besitzen. Es ist zu erwähnen, dass diese Datenstruktur nur benötigt wird, da das verwendete VTK keine 2D-FTFs unterstützt. Wird die Zuweisung von 2D-FTFs möglich, müssten die Voxelwerte nicht mehr modifiziert werden. Die Originalintensitäten können dann, wie in Abschnitt 3.5.1 beschrieben, vom Renderingalgorithmus bei der Komposition über eine FTF auf Farbwerte und über eine TTF auf Transparenzwerte abgebildet werden.

Die Qualität des 2D-Histogramms kann gesteigert werden, indem das betreffende Voxel nicht nur dem ermittelten Bin, sondern auch wie in [RBS05] vorgeschlagen seiner k -Nachbarschaft zugeordnet wird. In der implementierten Visualisierungskomponente wird eine Nachbarschaft von $k = 2$, also die 24 nächsten Bins, verwendet. Durch diesen Schritt

erhöht sich die Anzahl der Voxel, die einem Bin zugeordnet werden, sehr stark (Faktor 24). Liegt ein Bin am Rand des Histogramms, so wird nur der Zähler der Nachbarn, die innerhalb des Histogramms liegen, um eins erhöht. Insgesamt wird das Histogramm durch die Verwendung der k -Nachbarschaft geglättet. Dies äußert sich nach der Klassifikation in einer geringeren Anzahl an Bereichen. Weiterhin wird durch die Erhöhung der Voxelanzahl eines Bins die Ortsinformation repräsentativer. Hierdurch wird die Form der Bereiche homogener. Der Effekt der k -Nachbarschaft wird aus Abbildung 5.10 ersichtlich.



Abbildung 5.10.: Auswirkung der k -Nachbarschaft auf das 2D-Histogramm eines CT-Datensatzes. Je dunkler das Pixel im Bild des Histogramms, desto mehr Voxel wurden dem betreffenden Bin zugeordnet. Durch Verwendung der k -Nachbarschaft (b) wird die Bogenstruktur im Intensitäts-/Gradientenstärke-Histogramm erkennbar.

Berechnung der Ortsinformation in Form von Schwerpunkt und räumlicher Varianz

Neben der Zuordnung des Voxels zum betreffenden Histogrammbin und seinen 24 Nachbarn wird im `itkListSampleToSTFHistogramGenerator` parallel die Ortsinformation für diese Bins berechnet. Diese wird, wie in Abschnitt 4.4.1 beschrieben, in Form eines Schwerpunkts der Voxelkoordinaten und der dazugehörigen Varianz ermittelt. Die Varianz $v(f, g)$ der n Voxelkoordinaten $p_i(f, g), i = 1, \dots, n$, die zu einem Bin gehören, ist dabei das arithmetische Mittel der Abstandsquadrate vom berechneten Schwerpunkt $b(f, g)$. Die Berechnung der Varianz der Voxelpositionen um den Schwerpunkt lässt sich unter Beachtung der Rechenregeln für das arithmetische Mittel über den Zerlegungssatz wie folgt beschleunigen:

$$\begin{aligned}
v(f, g) &= \frac{1}{n} \sum_{i=1}^n \|p_i(f, g) - b(f, g)\| \\
&= \frac{1}{n} \sum_{i=1}^n [(p_{ix}(f, g) - b_x(f, g))^2 + \\
&\quad (p_{iy}(f, g) - b_y(f, g))^2 + (p_{iz}(f, g) - b_z(f, g))^2] \\
&= \frac{1}{n} \sum_{i=1}^n (p_{ix}(f, g) - b_x(f, g))^2 + \\
&\quad \frac{1}{n} \sum_{i=1}^n (p_{iy}(f, g) - b_y(f, g))^2 + \frac{1}{n} \sum_{i=1}^n (p_{iz}(f, g) - b_z(f, g))^2 \\
&= \left(\frac{1}{n} \sum_{i=1}^n p_{ix}(f, g)^2\right) - b_x(f, g)^2 + \\
&\quad \left(\frac{1}{n} \sum_{i=1}^n p_{iy}(f, g)^2\right) - b_y(f, g)^2 + \left(\frac{1}{n} \sum_{i=1}^n p_{iz}(f, g)^2\right) - b_z(f, g)^2 \\
&= \left(\frac{1}{n} \sum_{i=1}^n (p_{ix}(f, g)^2 + p_{iy}(f, g)^2 + p_{iz}(f, g)^2)\right) - \\
&\quad b_x(f, g)^2 - b_y(f, g)^2 - b_z(f, g)^2
\end{aligned} \tag{5.4}$$

$$\tag{5.5}$$

Roettger et al. [RBS05] haben Gleichung 5.4 für die Berechnung der Varianz verwendet. Diese setzt voraus, dass der Schwerpunkt zum Zeitpunkt der Varianzberechnung bereits bekannt ist. Somit muss der Schwerpunkt getrennt von der Varianz berechnet werden. Dadurch ist es notwendig, zweimal über das Volumen, für das die Ortsinformation berechnet werden soll, zu iterieren. Für die Normierung von Schwerpunkt und Varianz muss zusätzlich zweimal über das Histogramm iteriert werden. Durch die Umformung in Gleichung 5.5 und deren Implementierung im `itkListSampleToSTFHistogramGenerator` wird es möglich, die Varianz gleichzeitig mit dem Schwerpunkt zu berechnen, wodurch nur noch einmal über das Volumen iteriert werden muss. Auch die Normierung der Werte kann in einer Iteration über die Histogrammelemente erfolgen. Durch diese Optimierung war es möglich, die Rechenzeit, welche für die Vorverarbeitung der Daten bis zum Beginn der STF-Spezifikation durch den Nutzer notwendig ist, um 25% zu reduzieren. Die Berechnung des Histogramms mit Schwerpunkt und Varianz ist im Anhang in Listing A.2 gezeigt.

Drei Faktoren verbessern die Qualität der berechneten Ortsinformation. Zunächst wird durch die Adaptation des Histogramms auf Datensätze mit weniger als 256 Graubzw. Gradientenwerten die Anzahl der Bins reduziert, wodurch im Durchschnitt mehr Voxel einem Bin zugeordnet werden. Noch größeren Einfluss haben allerdings die Überabtafung, welche durch den Nutzer in die ITK-Filterpipeline eingefügt werden kann und die Verwendung der k -Nachbarschaft bei der Generierung des Histogramms. Überabtafung und k -Nachbarschaft erhöhen die Anzahl der Voxel im Histogramm um ein vielfaches, wodurch die statistische Aussagekraft der Ortsinformation gesteigert wird.

5.3.4. Klassifikation und Darstellung der 2D-Histogramme

Klassifikation der 2D-Histogramme

Um dem Nutzer die Spezifikation der STF zu erleichtern, erfolgt eine automatische Klassifikation der generierten 2D-Histogramme in Bereiche, die zusammenhängende Strukturen mit bestimmten Eigenschaften (Intensitäten und ggf. Gradientenstärke) im Datensatz repräsentieren (vgl. Abschnitt 4.4.2). Damit entfällt die aufwändige Angabe der Form der Bereiche durch den Nutzer, da sich die Interaktion auf die Auswahl der extrahierten Bereiche beschränkt.

Sortierung der Histogrammelemente Für die Klassifikation der Histogramme müssen die Bins zunächst anhand der Anzahl der dazugehörigen Voxel absteigend sortiert werden. Dazu wird der von Shell vorgeschlagene und nach ihm benannte Shellsort-Algorithmus verwendet (vgl. [She59] und [SHE]). Die sortierten Bins werden in einem sogenannten Standardvektor (`std::vector`), der neben der Position des Bins im 2D-Histogramm und der dazugehörigen Anzahl der Voxel auch die Ortsinformation speichert, aufgenommen. Zu jedem Bin wird weiterhin festgehalten, ob es bereits zu einer Klasse gehört, oder noch klassifiziert werden muss. Die Klassifikation beginnt mit dem Bin mit der höchsten Anzahl an Voxeln, das automatisch der ersten Klasse zugeordnet wird. Die Klassifikation wird fortgesetzt, bis alle Bins einer Klasse angehören (vgl. Abschnitt 4.4.2). Wie bereits erwähnt verbessert sich die Aussagekraft der Varianz und des Schwerpunkts mit steigender Anzahl der Voxel. Die Sortierung der Bins verbessert somit das Klassifikationsergebnis, da für die Überprüfung auf Zusammengehörigkeit der Bins die statistische Ortsinformation des Referenzbins mit der höchsten Anzahl an Voxeln herangezogen wird.

Distanznorm als Maß der Korrespondenz Wie in Abschnitt 4.4.2 und [RBS05] beschrieben, erfolgt die Klassifikation anhand eines Klassifikationsradius r und der Distanznorm N . Die Norm ist über die beiden aktuell untersuchten Bins T_0 und T definiert. T_0 ist das Referenzbin der aktuell zu erweiternden Klasse (aufgrund der Sortierung auch das Bin mit der höchsten Voxelanzahl in der Klasse) und T ist ein noch nicht klassifiziertes Bin aus dem Histogramm. Die Norm N ist wie folgt über die Schwerpunkte und Varianzen der beiden Bins definiert:

$$\begin{aligned} N(T, T_0) &= \|b(T) - b(T_0)\| + |v(T_0) - v(T)| \\ &= (b_x(T) - b_x(T_0))^2 + (b_y(T) - b_y(T_0))^2 + \\ &\quad (b_z(T) - b_z(T_0))^2 + (v(T_0) - v(T))^2 \end{aligned} \quad (5.6)$$

Die Entscheidung über eine gemeinsame Klassenzugehörigkeit von T_0 und T wird anhand eines Vergleichs zwischen dem Wert der Norm und dem Klassifikationsradius gefällt. Ist N kleiner als r , so wird das Bin T der Klasse zugeordnet, der auch das Referenzbin T_0 angehört. Die Adaptation des 2D-Histogramms verbessert das Klassifikationsergebnis, da neben dem Vergleich auch die Anzahl der Voxel des Bins T überprüft wird, um zu entscheiden, ob das Bin der Klasse hinzugeordnet wird. Bins mit weniger als 5 Voxeln werden automatisch der Klasse Hintergrund zugeordnet, da die berechnete Ortsinforma-

tion nicht hinreichend aussagekräftig für eine Klassifikation ist.

Für die Klassifikation wird ein 2D-Feld von ganzzahligen Skalarwerten der gleichen Größe wie das zu klassifizierende 2D-Histogramm in Form eines ITK-Bildes (`itk::Image`) erzeugt. In diesem Feld werden die Klassennummern der Bins gespeichert. Das Bildpixel an der Stelle des Bins mit der höchsten Anzahl an Voxeln erhält automatisch die Klassennummer 1. Die Klasse Hintergrund wird in diesem 2D-Feld durch eine 0 repräsentiert. Wird nun ein Bin T einer Klasse zugeordnet, so wird dieses Bin im Vektor der sortierten Bins als bereits klassifiziert markiert. Weiterhin wird der Bildpixel an der Position des Bins im Histogramm auf die aktuelle Klassennummer des Referenzbins T_0 gesetzt. Am Ende der Klassifikation ist das ITK-Bild demnach in Bereiche gleicher Klassennummern eingeteilt. Für einen effizienten Zugriff auf alle Bins einer Klasse wird eine `std::multimap` mit der Klassennummer und dem Index des gerade klassifizierten Bins aufgebaut. Die Elemente der `std::multimap` werden aufsteigend nach der Klassennummer sortiert.

Darstellung der klassifizierten Histogramme

Die 2D-Histogramme werden auf der Programmoberfläche durch eine VTK-Visualisierung der klassifizierten Histogramme dargestellt. Hierzu wird das zweidimensionale ITK-Bild in ein VTK-Bild (`vtkImageData`) der Klassennummern umgewandelt und unter Verwendung einer eindimensionalen FTF (`vtkColorTransferFunction`) in der dafür vorgesehenen Zeichenfläche (`wxRenderWindow`) gerendert. Die FTF besitzt sieben Stützstellen, die gleichmäßig über die Anzahl k der vergebenen Klassennummern verteilt sind. Sie ist als Regenbogenfarbskala wie in Tabelle 5.2 gezeigt über die k vergebenen Klassennummern definiert.

Klassennummer	Farbwert in RGB	Farbe
0	(1.0, 0.0, 0.0)	Rot
$k*(1/6)$	(1.0, 1.0, 0.0)	Gelb
$k*(2/6)$	(0.0, 1.0, 0.0)	Grün
$k*(3/6)$	(0.0, 1.0, 1.0)	Türkis
$k*(4/6)$	(0.0, 0.0, 1.0)	Blau
$k*(5/6)$	(1.0, 0.0, 1.0)	Violett
$k*(6/6)$	(1.0, 0.0, 0.0)	Rot




Tabelle 5.2.: 1D-Farbtransferfunktion für die Darstellung des klassifizierten 2D-Histogrammbildes.

In die Darstellung des klassifizierten 2D-Histogramms wurde die Möglichkeit eingebaut, über die rechte Maustaste zu zoomen sowie das 2D-Histogramm mit der mittleren Maustaste zu verschieben. Dies erleichtert die Bereichsauswahl aus dem klassifizierten Histogramm, bei der die Bereiche durch den Nutzer angeklickt werden müssen. Weiterhin kann die Anzeige des Histogramms über die „Reset Camera“ Schaltfläche wieder auf den Ausgangszustand zurückgesetzt werden, in dem das gesamte Histogramm gezeigt wird.

Um dem Nutzer eine Rückmeldung über die bereits gewählten Klassen zu bieten, wird ein zweites VTK-Bild mit der gleichen Größe wie das Bild des klassifizierten Histogramms verwendet. Allerdings werden diesem nur zwei Grauwerte (0 und 1 bzw. nicht

gewählt und gewählt) zugeordnet. Weiterhin erfolgt die Darstellung dieses Bildes über eine FTF und TTF. Über die TFs werden Pixel mit dem Grauwert 0 halbtransparent und grau dargestellt, während Pixel mit dem Grauwert 1 vollständig transparent visualisiert werden. Wird nun dieses Feedbackbild in der Visualisierung dem Bild des klassifizierten Histogramms überlagert, so werden die nicht ausgewählten Bereiche abgedunkelt und Bereiche, die der STF hinzugefügt wurden durch gesättigte Farben dargestellt. Die Modifikation des Feedbackbildes nach der Auswahl einer Klasse ist effizient über die `std::multimap` möglich, welche während der Klassifikation des Histogramms aufgebaut wurde.

5.3.5. Implementierung der Spatialized Transfer Functions

Generierung und Modifikation der Transferfunktionen

Dem Nutzer stehen verschiedene Interaktionsmöglichkeiten mit der Programmoberfläche zur Verfügung. Diese sollen die Spezifikation der STFs erleichtern und ihm die Möglichkeit bieten Einfluss auf die Klassifikation der Histogramme zu nehmen.

Für die Definition einer STF ist es nötig, Bereiche aus dem klassifizierten Histogramm auszuwählen und diese so der STF hinzuzufügen oder wieder zu entfernen. Die Bereichsauswahl wird über eine *Pick*-Funktion realisiert. Hierzu muss der Nutzer mit der linken Maustaste ein beliebiges Bin des ihn interessierenden Bereichs anklicken und den Bereich anschließend durch Drücken der Taste „s“ aktivieren und somit zur STF hinzufügen. Durch die *Pick*-Funktion wird zunächst bestimmt, welcher Bereich im klassifizierten Histogramm ausgewählt wurde. Dazu werden die Koordinaten des angeklickten Punktes in der VTK-Visualisierung des klassifizierten Histogramms in die Bildkoordinaten des `itk::Image` der Klassennummern umgerechnet. Die Histogrammposition des angeklickten Bins wird zusammen mit der Klassennummer des ausgewählten Bereichs als Feedback auf der Programmoberfläche ausgegeben. Für einen effizienten Zugriff auf alle Bins des gewählten Bereichs wird die während der Klassifikation generierte `std::multimap` genutzt, welche über die Klassennummer und den Index der dazugehörigen Bins definiert ist. Diese `std::multimap` ermöglicht es, alle Bins der gewählten Klasse in der Histogrammvisualisierung durch gesättigte Farben darzustellen. Hierzu wird das Feedbackbild des Histogramms an den Positionen der Bins des gewählten Bereichs auf den Grauwert 1 gesetzt und die Darstellung des Histogramms aktualisiert. Nach der Auswahl eines Bereiches erfolgt eine Aktualisierung der 3D-Visualisierung und die Voxel, welche in diesen Bereich fallen, werden zusätzlich zu bereits ausgewählten Strukturen farbig dargestellt. Wird ein bereits aktivierter Bereich ausgewählt, so wird dieser aus der STF und die betroffenen Strukturen aus der Visualisierung entfernt. Ob ein Bereich bereits ausgewählt wurde, wird in einem `std::vector` von `boolean` Variablen für jede Klasse gespeichert. Dieser wird nach einer Klassenauswahl an der betreffenden Stelle invertiert und somit wird gespeichert ob eine Klasse aktiv oder inaktiv ist.

Der Nutzer kann weiterhin den für die Klassifikation der Histogramme verwendeten Radius r über einen Schieberegler vorgeben. Mit diesem bestimmt er die maximale Größe der Strukturen, die durch einen Bereich repräsentiert werden können. Er hat somit die Möglichkeit, die Klassifikation an den vorliegenden Datensatz anzupassen. Denkbar ist z.B. die Klassifikation mit einem sehr großen Radius, was zu einer Verringerung der

Klassenanzahl führt. Indem der Nutzer sich eine STF über ein solches 2D-Histogramm definiert, kann er sich einen schnellen Überblick über den Datensatz verschaffen, da er für eine Übersichtsdarstellung nur wenige Bereiche aus dem Histogramm auswählen muss. Weiterhin hat er so die Möglichkeit die Beziehung zwischen dem ausgewählten Bereich und der resultierenden Visualisierung kennen zu lernen. Er kann so leichter Positionen im Histogramm anatomischen Strukturen zuordnen. In einer anschließenden feineren Klassifikation durch einen kleineren Radius wird der gleiche Histogrammteil in mehrere Bereiche unterteilt sein. Dennoch kann der Nutzer, diesen Bereichen Strukturen im Volumen zuordnen. Die Auswirkung von verschiedenen Radien auf die Klassifikation von 2D-Histogrammen verdeutlicht die Abbildung 5.11.

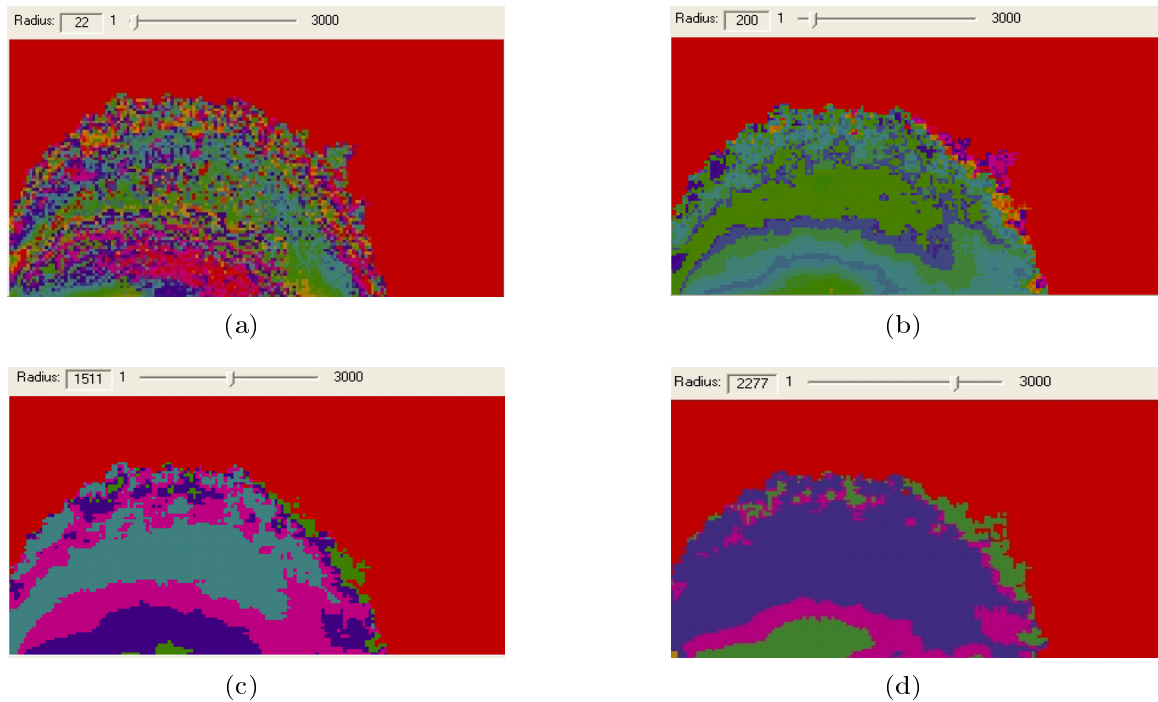


Abbildung 5.11.: Klassifikation eines 2D-Histogramms mit unterschiedlichen Radien. (a) Radius 22 mit 1002 Klassen (b) Radius 200 mit 35 Klassen (c) Radius 1511 mit 6 Klassen (d) Radius 2277 mit 5 Klassen

Der angegebene Radius wird für die Klassifikation beider im gradientenbasierten Ansatz verwendeten Histogramme genutzt. Wurde bereits ein Histogramm berechnet und klassifiziert, so erfolgt nach der Änderung des Radius eine erneute Klassifikation, welche die 2D-Bilder mit den Klassennummern aktualisiert. Die bisher erstellten 3D-Visualisierungen bleiben erhalten, bis der Nutzer beginnt, für den jeweiligen Datensatz eine neue STF basierend auf dem neu klassifizierten Histogramm zu spezifizieren.

Anwendung der Transferfunktion

Nach der Modifikation einer STF durch die Auswahl eines Bereiches muss die Transferfunktion auf das Volumen angewendet werden, damit die Veränderung auch in der Visualisierung aktualisiert wird. Da die verwendete VTK-Version, wie bereits erwähnt, keine 2D-FTFs unterstützt, wurde die Anwendung der STF in der Software implementiert.

Durch die Verwendung der beiden generierten `std::multimap` Datenstrukturen ist es möglich, effizient auf die Voxel zuzugreifen, die in den ausgewählten Bereich des Histogramms fallen. Zunächst können über die Klassennummern und die während der Klassifikation generierte `std::multimap` alle Bins des ausgewählten Bereichs ermittelt werden. Über die Indizes der betreffenden Bins können die Voxelpositionen aus der `std::multimap` ermittelt werden, welche während der Histogrammberechnung generiert wurde. Den Voxeln, welche dem ausgewählten Bereich des 2D-Histogramms zugeordnet sind, wird dann die Klassennummer des Bereichs zugewiesen. Der modifizierte Volumendatensatz wird anschließend in ein `vtkImageData` umgewandelt und an den Renderer übergeben. Für die Abbildung der Klassennummern auf Farbe muss dem Renderer weiterhin eine 1D-FTF übergeben werden. Hierzu wird die gleiche FTF wie bei der Visualisierung des Histogramms verwendet. Diese ist als Regenbogenfarbskala gleichmäßig über die Anzahl der vergebenen Klassennummern verteilt.

Damit ergibt sich ein Unterschied zum Ansatz von [RBS05], da in dieser Arbeit nicht mit zufällig erzeugten Farbwerten gearbeitet wird. Die Verwendung zufälliger Farbwerte ist aber leicht in die Histogrammklassifikation der entwickelten Visualisierungskomponente einzubauen, wenn die 2D-FTFs im VTK verfügbar werden.

Zuweisung von Transparenz Neben der FTF wird dem Renderer eine TTF übergeben. Diese ist ebenfalls über die Klassennummern definiert, weist allerdings allen Werten die gleiche Opazität zu. Die Begriffe Opazität und Transparenz sind Antonyme. Die Transparenz eines Voxels ergibt sich aus folgender Formel:

$$\text{Transparenz } \alpha = 1 - \text{Opazität} \tag{5.7}$$

Es wird eine sogenannte globale TTF verwendet (vgl. [RBS05] und [HRH03]), da der zugewiesene Transparenzwert einzig von Intensitätswert (Klassennummer) des Voxels abhängig ist und nicht von dessen Umgebung beeinflusst wird, wie im Falle einer gradientenbasierten TTF. Die Transparenz kann durch den Schieberegler „Opacity“ modifiziert werden (s. Abb. 5.4). Dazu wird der Wertebereich des Sliders $[0 \dots 100]$ auf $[0 \dots 1]$ normiert und der gewählte Wert mit dem globalen Opazitätswert 0,2 multipliziert. Der globale Opazitätswert stellt somit auch die maximal mögliche Opazität eines Voxels dar. Durch die Komposition beim Rendering des Datensatzes und der damit verbundenen Kumulation der Opazitätswerte ergeben sich auch mit einer so geringen maximalen Opazität bereits opake Visualisierungen. Da somit nur die unteren 20% des Opazitätbereichs durch den Schieberegler modifiziert werden müssen, wird es möglich die Transparenz der Visualisierung sehr genau an die Bedürfnisse anzupassen.

5.3.6. Anwendung der Transformation und Berechnung der sich überschneidenden Regionen

An dieser Stelle sei noch einmal darauf hingewiesen, dass dieser Arbeitsschritt im multiintensitätsbasierten Ansatz bereits zu Beginn der Filterpipeline ausgeführt werden muss, um die Generierung einer gemeinsamen STF zu ermöglichen. Weiterhin liegt im multiintensitätsbasierten Fall an dieser Stelle in der Visualisierungspipeline nur ein eingefärbter Volumendatensatz vor (vgl. Abschnitt 5.3.1 und Abb. 5.6). Bis auf die Art der Interpolation ist der Ablauf dieses Arbeitsschrittes allerdings in beiden Ansätzen gleich.

Für die Anwendung der Transformation müssen zunächst die berechneten Transformationsparameter aus dem `MIPParameterChangeManager` geladen und in eine Transformationsmatrix mit 4 x 4 Elementen umgewandelt werden. Diese wird benötigt, um die Datensätze entsprechend der Registrierung zueinander auszurichten, indem die berechnete rigide Transformation auf den betreffenden Datensatz angewandt wird. Die Transformationsinformation ist im `MIPParameterChangeManager` über drei Translationswerte T_x, T_y und T_z sowie drei Rotationswinkel θ_x, θ_y und θ_z um die Koordinatenachsen gespeichert. Aus diesen Werten wird die 4x4 Matrix wie folgt berechnet:

$$M = \begin{pmatrix} \cos\theta_x \cos\theta_y + \sin\theta_x \sin\theta_z \sin\theta_y & \sin\theta_x \cos\theta_y - \cos\theta_x \sin\theta_z \sin\theta_y & \cos\theta_z \sin\theta_y & T_x \\ -\sin\theta_x \cos\theta_z & \cos\theta_x \cos\theta_z & \sin\theta_z & T_y \\ \sin\theta_x \sin\theta_z \cos\theta_y - \cos\theta_x \sin\theta_y & -\cos\theta_x \sin\theta_z \cos\theta_y - \sin\theta_x \sin\theta_y & \cos\theta_z \cos\theta_y & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5.8)$$

Die in der Registrierung berechnete Transformation muss auf den registrierten Datensatz angewendet werden, um das Ergebnis der Registrierung in der Visualisierung darstellen zu können. Weiterhin ist die Transformation im multiintensitätsbasierten Ansatz notwendig, um ein gemeinsames 2D-Histogramm über beide Datensätze erstellen zu können. Um die Transformation anwenden zu können, muss zunächst die Voxelgröße (*Spacing*) der Datensätze angeglichen werden. Dazu wird der Datensatz mit der schlechteren Ortsauflösung (*Spacing*) innerhalb der Schichten durch eine Überabtastung an die Voxelgröße des anderen Datensatzes angepasst. Da die räumliche Ausdehnung des Datensatzes in Weltkoordinaten gleich bleiben muss, um Verzerrungen zu verhindern, ändert sich die Anzahl der Voxel in der neu abgetasteten Dimension. Die Überabtastung und die Anwendung der Transformation können für das zu transformierende Volumen (*Floating Image*) in einem Verarbeitungsschritt mit dem `itk::ResampleImageFilter` umgesetzt werden. Dazu müssen dem Filter neben der Transformationsmatrix und der zu erreichenden Voxelgröße je Dimension auch die Anzahl der Voxel je Dimension übergeben werden. Weiterhin ist der Filter über eine Interpolationsfunktion parametrisiert, mit der die neuen Datenwerte an den Abtastpunkten ermittelt werden. Die Interpolation erfolgt in dieser Arbeit im multiintensitätsbasierten Ansatz trilinear, da auf den original Grauwerten gearbeitet wird. Im gradientenbasierten Ansatz wird durch „Nearest Neighbour“ Interpolation der neue Abtastwert bestimmt, da im zu transformierenden Volumendatensatz die Klassennummern gespeichert sind und diese nicht modifiziert werden dürfen, da dies zu Verfälschungen der Visualisierung führen würde. Wenn bei der Transformation des Datensatzes Teile des Volumens aus der interessierenden Region heraus rotiert oder verschoben werden, gelangen Bereiche aus der „Umgebung“ des Volumens in die Regi-

on. Voxel an dieser Position haben nach der Transformation keine Bildinformation mehr (s. Abb. 5.12). Für diesen Fall muss dem `itk::ResampleImageFilter` ein Intensitätswert übergeben werden, der an diese Voxelpositionen eingetragen wird (der sogenannte *DefaultPixelValue*).

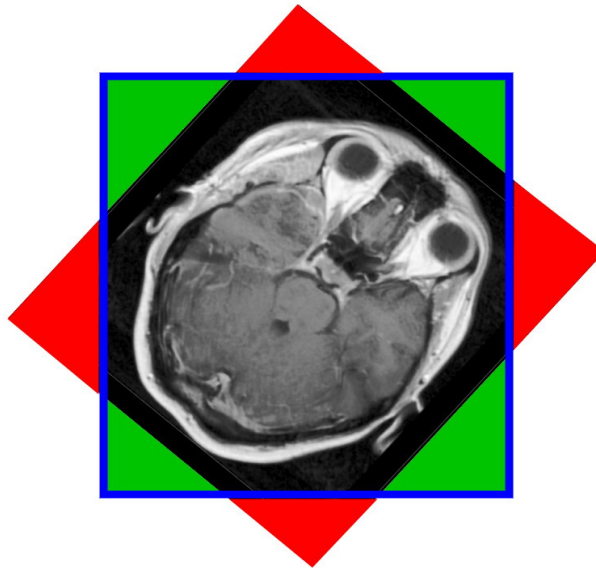


Abbildung 5.12.: Anhand dieser Abbildung soll die Auswirkung der Transformation auf die Bildinformation verdeutlicht werden. Durch die hier dargestellte Rotation des Datensatzes ist Bildinformation verlorengegangen (hier rot dargestellt). Den Positionen im Bild an denen keine Messwerte mehr vorliegen wird der *DefaultPixelValue* zugewiesen (hier grün dargestellt).

Für das Volumen, das während der Registrierung nicht verändert wurde (*Fixed Image*), erfolgt ggf. eine erneute Abtastung durch den `itk::ResampleImageFilter` um die Voxelausdehnung anzupassen. Nach der Transformation und der Abtastung liegen die beiden Datensätze im gleichen Koordinatensystem zueinander ausgerichtet vor. Je nach Voxelgröße und -anzahl der Originaldatensätze sind diese beiden Volumen allerdings unterschiedlich groß.

Um die Generierung des 2D-Histogramms für den multiintensitätsbasierten Ansatz zu ermöglichen, müssen zu Beginn der Filterpipeline die sich überschneidenden Bereiche der Datensätze ermittelt werden. Dazu muss der Teil der Datensätze abgeschnitten werden, dessen Voxeln kein Voxelwert aus dem anderen Datensatz zugeordnet werden kann. Bei der Berechnung der sich überschneidenden Bereiche wird davon ausgegangen, dass sich die abgebildeten Körperregion im Zentrum der neuen Datensätze befinden und die interessierenden Strukturen vollständig im Datensatz enthalten sind. Diese Annahmen werden, durch die mit dem `itk::ResampleImageFilter` transformierten und abgetasteten Datensätze, erfüllt. Die zu verarbeitende Region eines Datensatzes ist über einen Anfangspunkt in Bildkoordinaten (im Folgenden Minimum der Region genannt) und über die Anzahl der Voxel je Dimension definiert. Da beide Datensätze die gleiche Voxelgröße haben und der Anfangspunkt der Region in beiden Fällen am Bildursprung liegt, kann die

überlappende Region einfach über die Differenz der Voxelanzahl je Dimension bestimmt werden. Das Minimum der zu verarbeitenden Region des größeren Datensatzes wird dazu in der betreffenden Dimension um die halbe Differenz erhöht und die Anzahl der Voxel in dieser Dimension um die halbe Differenz reduziert. Die Berechnung der überlappenden Region ist im Anhang in Listing A.1 auszugsweise gezeigt.

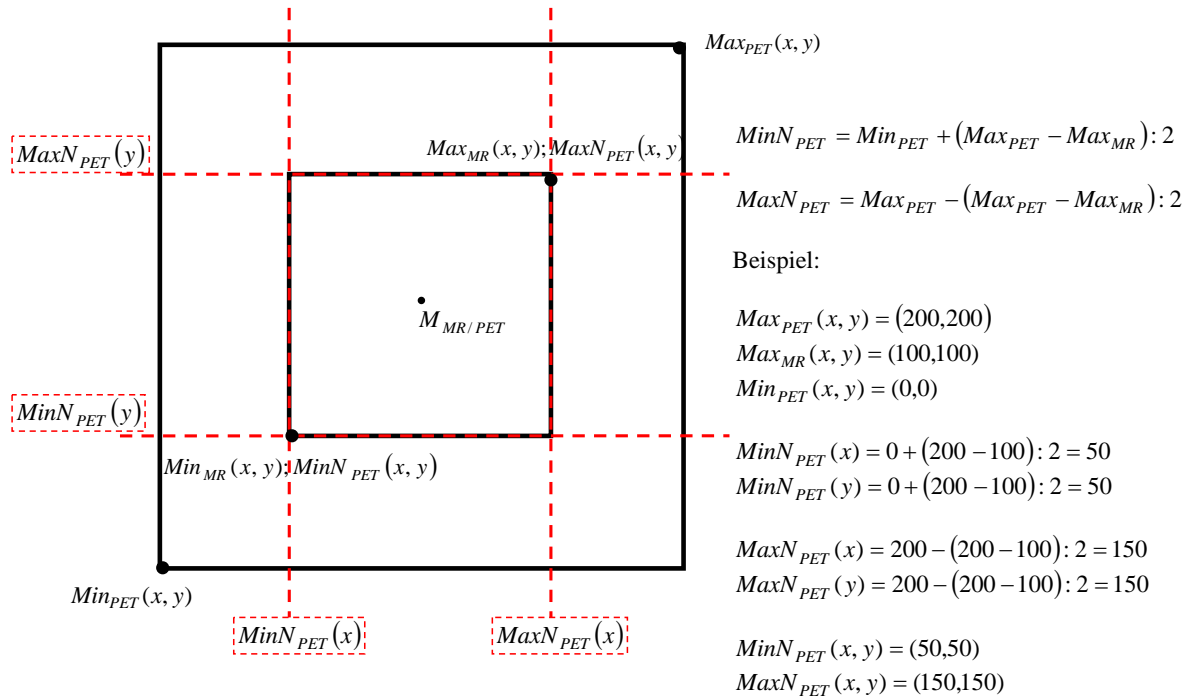


Abbildung 5.13.: Beispielhafte Berechnung der sich überschneidenden Bereiche.

Im multiintensitätsbasierten Fall liegen nach der Transformation, dem Anpassen der Voxelausdehnung und der Berechnung der sich überschneidenden Region zwei gleich große Volumendatensätze vor. In diesen sind die interessierenden Strukturen überlagert und für jeden Voxel existiert ein korrespondierendes Voxel aus dem anderen Datensatz. Durch die Schaltfläche „Write Registered Data“ ist es dem Nutzer möglich, diese Datensätze zu speichern. Dies ermöglicht es, bei einer erneuten Visualisierung anstatt der Originaldaten die bereits registrierten Datensätze zu laden und damit den Transformationsschritt und die Berechnung der überschneidenden Regionen auszulassen. Wurden bereits registrierte Daten geladen, so muss der Nutzer die *Checkbox* „Pre-Registered Data“ auswählen um den Programmablauf anzupassen. Die beiden überlagerten Volumendatensätze bilden die Grundlage für die Spezifikation der gemeinsamen STF und werden von der zuvor beschriebenen ITK-Filterpipeline verarbeitet.

Die beiden Volumendatensätze des gradientenbasierten Ansatzes werden erst direkt vor dem Rendering miteinander registriert und bilden die Grundlage für ein gemeinsames Rendering.

5.3.7. Rendering

Die Darstellung der Volumenvisualisierung erfolgt in der Zeichenfläche „*Labeled Volume*“ (s. Abb. 5.4). Für die Darstellung der Volumendatensätze wurde der *Ray Casting* Algorithmus verwendet. Dieser ist im VTK über die Klassen `vtkRenderer`, `vtkRayCastMapper` und `vtkVolumeRayCastCompositeFunction` implementiert. Dem Renderer wird das darzustellende Volumen (`vtkVolume`) übergeben. Diesem ist neben der 1D-FTF und der globalen TTF ein Mapper zugeordnet, welcher die Datenwerte auf die optischen Parameter abbildet. Die FTF und TTF werden im VTK jeweils über eine *Lookup Table* implementiert. Diese sind über zwei Parameter definiert. Der erste Parameter bestimmt konkrete Stützstellen im Wertebereich des abzubildenden Datenwertes während der zweite Parameter die zu vergebende optische Eigenschaft definiert. Liegt ein Datenwert im Volumen zwischen zwei Stützstellen, wird die zu vergebende optische Eigenschaft durch lineare Interpolation ermittelt. Um die Visualisierung dreidimensional erscheinen zu lassen, wird während des Renderings eine Beleuchtungsberechnung (engl. *Shading*) durchgeführt. Die Reflektionseigenschaften des Volumens (vgl. 3.5.2) wurden bei den VTK-Standardwerten belassen (ambient = 10%, diffus = 70% und spekulär = 20%). Die Dichte der Abtastpunkte wird in der VTK-Implementation automatisch anhand der zuletzt benötigten Renderingzeit angepasst. Wird das Volumen durch den Nutzer rotiert, verschoben oder skaliert, so wird die Auflösung so weit reduziert, bis das Rendering interaktiv aktualisiert werden kann. Nach Beendigung der Interaktion wird die Auflösung wieder erhöht. Die Interpolation der Datenwerte für die Abtastpunkte auf dem Strahl erfolgt mit dem *Nearest Neighbour* Verfahren, bei dem der Abtastpunkt den Datenwert des nächstliegenden Voxels erhält. Diese Art der Interpolation wurde gewählt, um die vergebenen Klassennummern nicht durch eine Interpolation zu verfälschen, da ansonsten Voxel eines Bereichs auf unterschiedliche Farben abgebildet werden.

6. Ergebnisse

Das folgende Kapitel soll durch Zahlen und Fakten sowie anschauliche Visualisierungsbeispiele zeigen, welche der gestellten Anforderungen mit Hilfe der implementierten Visualisierungskomponente erfüllt werden konnten. In den Abschnitten werden die verschiedenen Möglichkeiten der Volumenvisualisierung vorgestellt, welche die Software bietet und diese auf ihre Vor- und Nachteile hin untersucht. Dabei werden teilweise die Ausgangsdaten in Form von Schichtbildern gezeigt aus denen mit Hilfe der implementierten Komponente verschiedene 3D-Visualisierungen erstellt wurden.

Eine ausführliche Datentabelle mit den für die jeweiligen Ergebnisbilder verwendeten Visualisierungsparametern ist im Anhang zu finden (Tabelle A.2). Die dort angegebenen Zeiten für die verschiedenen Arbeitsschritte, wurden auf einem PC mit 2 Pentium4 Prozessoren, jeweils mit 1,1 GigaHertz Taktrate ermittelt. Der verwendete Rechner ist mit 2 GB RAM und einer Geforce4 Quadro FX Grafikkarte mit 128 MB RAM ausgestattet. Die Generierung des Histogramms wird durch die Multithreading Möglichkeit der ITK-Filter beschleunigt, da teilweise beide Prozessoren gleichzeitig Berechnungen durchführen und somit das gesamte Leistungsspektrum des Systems ausgenutzt wird.

6.1. Gradientenbasierte Visualisierungen einzelner Volumen

In diesem Abschnitt wird die Anwendbarkeit des Visualisierungssystems auf die Darstellung einzelner Volumendatensätze untersucht. Hierzu werden der MRT- und der PET-Schädeldatensatz aus „CT MRT PET“ separat unter Verwendung gradientenbasierter STFs visualisiert. Hierbei soll gezeigt werden, dass auch Visualisierungen einzelner Datensätze möglich sind und dabei Strukturen separat dargestellt werden können. Weiterhin soll Kontextinformation aus dem gleichen Datensatz hinzugefügt werden. Die zugrunde liegenden Histogramme wurden mit einem Radius von 200 mm klassifiziert. Für die Vorverarbeitung mit Berechnung der überschneidenden Region liegen keine Messzeiten vor, da die Volumendaten das Ergebnis der Anwendung einer berechneten Registrierung sind und somit bereits zuvor die überschneidende Region aus beiden Datensätzen extrahiert wurde. Weiterhin wurde dabei eine Abtastung durchgeführt, um die Voxelausdehnung des PET-Datensatzes auf die des MRT-Datensatzes anzupassen. Somit stimmen die Datensätze in räumlicher Auflösung und Voxelanzahl überein.

6.1.1. Monomodale Visualisierung eines MRT-Datensatzes

Die in Abbildung 6.1 dargestellten Visualisierungen zeigen die 3D-Ansicht der MRT-Aufnahme. Zunächst wurden aus den 37 Klassen des Histogramms fünf Klassen ausgewählt, welche die Haut repräsentieren (s. Abb. 6.1(a)). Im Anschluss daran wurden für die Visualisierungen aus Abbildung 6.1(b) und 6.1(c) sechs weitere Klassen ausgewählt, um das Gehirn zur Visualisierung hinzuzufügen. Die axiale Darstellung in Abbildung 6.1(b) zeigt sehr deutlich, wie detailliert die Hirnoberfläche dargestellt wird. Der dreidimensionale Eindruck ergibt sich vor allem aus der Beleuchtungsberechnung vor dem Rendering der Daten. Die sagittale Ansicht wirkt weniger eindrucksvoll, da die Schichtdicke zu groß ist und sich daraus Stufenartefakte ergeben.

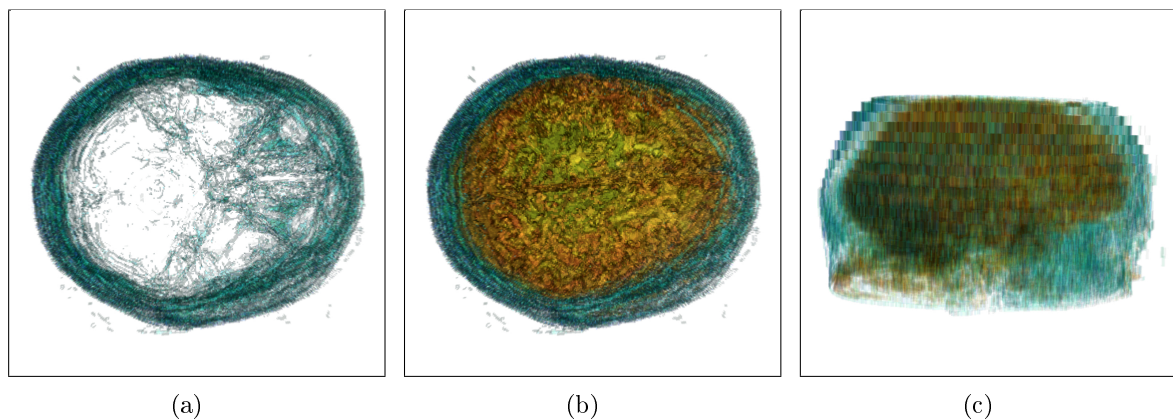


Abbildung 6.1.: Ansicht auf die monomodalen gradientenbasierten Visualisierungen eines MRT-Schädeldatensatzes.

Die ausgewählten Klassen und die dazugehörigen Strukturen in der Visualisierung sind in der Abbildung 6.2 dargestellt.

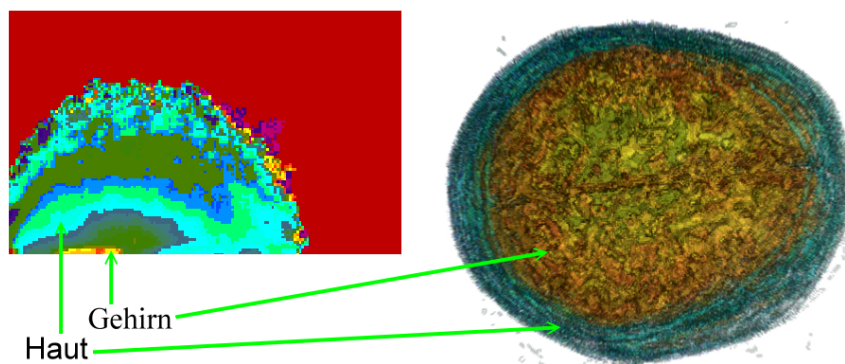


Abbildung 6.2.: Histogramm des MRT-Datensatzes mit Beschriftung der Regionen und der dazugehörigen Strukturen.

6.1.2. Monomodale Visualisierung eines PET-Datensatzes

Für den PET-Datensatz ergab sich eine Einteilung des Histogramms in 38 Klassen. Der in Abbildung 6.3(a) bis (c) dargestellte stark aktive Teil des Gehirns wird dabei durch eine dieser Klassen repräsentiert. Durch die Auswahl zweier weiterer Klassen kann dem stark aktiven Teil des Gehirns die Haut als anatomischer Kontext hinzugefügt werden. Die Abbildungen 6.3(d) bis (f) zeigen die sich daraus ergebende Visualisierung. Werden insgesamt sieben Klassen ausgewählt, so kann neben der Haut das Gehirn vollständig dargestellt werden, wie in den Abbildungen 6.3(g) bis (i) gezeigt.

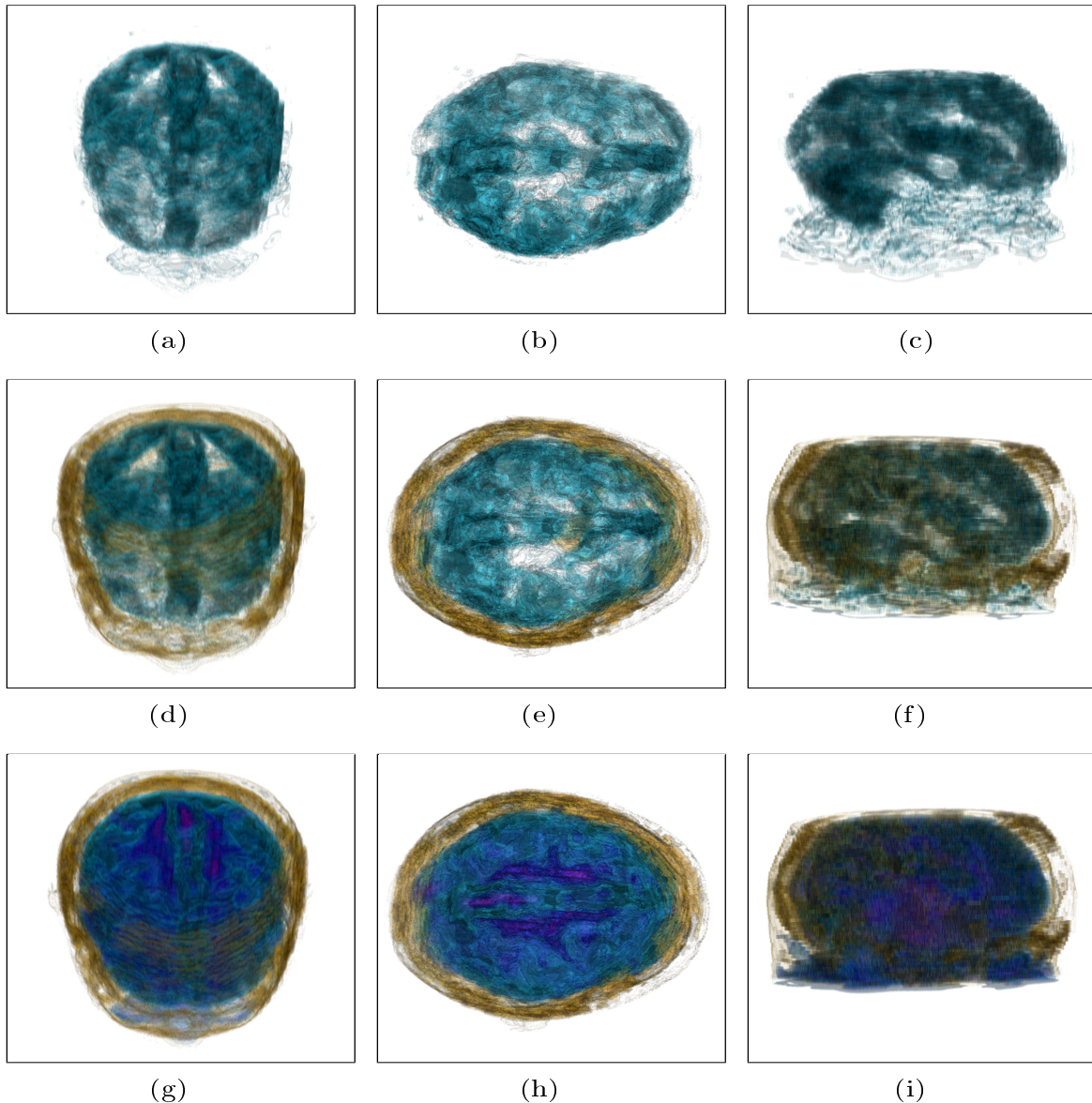


Abbildung 6.3.: Ansicht auf die gradientenbasierten Visualisierungen eines PET-Schädel Datensatzes. Jeweils coronale, axiale und sagittale Ansichten mit verschiedenen ausgewählten Klassen.

Mit diesen Bildreihen wurde die Anforderung, medizinische Volumendaten auch separat visualisieren zu können, durch die Darstellung eines MRT- und eines PET-Volumendatensatzes erfüllt. Es wurde gezeigt, dass es bei dieser Art von Visualisierung möglich ist, Kontextinformation in Form von anderen anatomischen Strukturen zur Visualisierung hinzuzufügen. Durch die Vereinigung zweier gradientenbasierter Visualisierungen, welche z.B. die Haut des Patienten darstellen ist es weiterhin möglich die Qualität der verwendeten Registrierung zu beurteilen und die Transformation ggf. zu korrigieren. Vorteilhaft ist dabei, dass die Transformation erst am Ende der Visualisierungspipeline angewendet wird und somit die generierte STF auch nach Modifikation der Transformation weitergenutzt werden kann.

6.2. Vergleich 3D-Visualisierung mit Schachbrettmuster-Schichtdarstellung

In diesem Abschnitt werden zwei miteinander registrierte CT- und PET-Datensätze aus „CT PET Thorax“ unter Verwendung gradientenbasierter STFs gemeinsam visualisiert. Die beiden verwendeten Thoraxdatensätze zeigen den Oberkörper eines Patienten und einen Teil des Kopfes. Das Ziel der Visualisierung ist die Darstellung eines Lymphknotentumors links unterhalb des Kiefers. Dieser hebt sich über die hohe Aktivität in den PET-Schichtbildern deutlich von seiner Umgebung ab. Allerdings kann im PET-Datensatz keine eindeutige Lagezuordnung vorgenommen werden. In den CT-Schichtbildern ist der Tumor hingegen nur schwer vom umgebenden Gewebe zu trennen und nur über seine räumliche Ausdehnung und die damit verbundenen morphologischen Veränderungen zu erkennen. Die erzeugten dreidimensionalen Visualisierungen sollen mit der in Abschnitt 3.2 beschriebenen Schichtbilddarstellung durch ein Schachbrettmuster verglichen werden (s. Abb. 3.2). Eine solche Schichtbilddarstellung der verwendeten Datensätze ist in Abbildung 6.4 gezeigt.

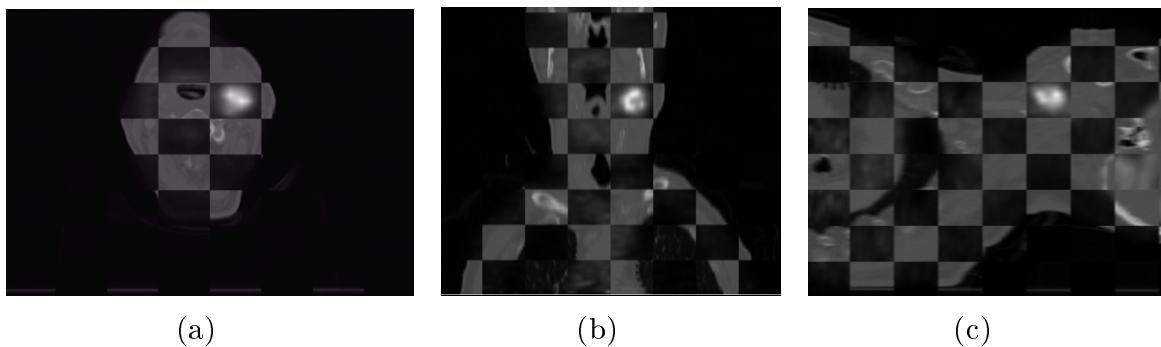


Abbildung 6.4.: Schichtbilddarstellung der zu visualisierenden CT- und PET-Datensätze unter Verwendung eines Schachbrettmusters mit 8 Unterteilungen in jede Raumrichtung. (a) axiale, (b) coronale und (c) sagittale Ansicht. Für die sagittale Ansicht musste die Reihenfolge der Datensätze getauscht werden, da das Feld, welches den Tumor aus dem PET-Datensatz zeigt, ansonsten den CT-Datensatz zeigt hätte.

Für die Schichtbilddarstellungen wurde ein Schachbrettmuster mit 8 Unterteilungen in jede Raumrichtung gewählt. Bei der sagittalen Darstellung trat der im Abschnitt 3.2 beschriebene Nachteil auf, dass die Lage der Felder des Schachbrettes bei mehr als einer Unterteilung nicht frei wählbar ist und somit an einer wichtigen Position im Schichtbild der falsche Datensatz angezeigt wurde. Um für ein besseres Verständnis den Tumor in allen drei Ansichten aus dem PET-Datensatz darzustellen, musste demnach für das sagittale Schichtbild die Reihenfolge der Datensätze getauscht werden. Allerdings wurde der Tumor anschließend in den beiden anderen Ansichten nicht mehr über die PET-Daten dargestellt. Ein weiterer Nachteil ist die rein zweidimensionale Darstellung der Daten über Schichtbilder und der damit verbundene Aufwand für die schichtweise Betrachtung. Dies ist vor allem für die Kommunikation von Diagnoseergebnissen ineffizient und zeitaufwändig. Abhilfe soll hier eine 3D-Darstellung der Daten leisten, die in Abbildung 6.5 und 6.6 gezeigt wird.

Abbildung 6.5 zeigt eine coronale Ansicht der beiden Datensätze. Zunächst wurde in 6.5(a) nur eine Klasse des PET-Histogramms ausgewählt, diese repräsentiert die hellsten Voxel im Volumen und somit die aktivsten Strukturen im Körper des Patienten. Dies allein reicht nicht aus, um Aussagen über die Lage oder die Art der Strukturen machen zu können. Durch die Auswahl einer weiteren Klasse aus dem PET-Histogramm wurde die stark aktive Hirnoberfläche als Kontextinformation aus dem gleichen Datensatz zur Visualisierung hinzugefügt (s. Abb. 6.5(b)). Eine Lagezuordnung im Verhältnis zu weiteren anatomischen Strukturen wird allerdings erst möglich, wenn Teile des CT-Datensatzes als komplementäre Information visualisiert werden (s. Abb. 6.5(c)). Die gezeigten Teile des CT-Datensatzes werden im CT-Histogramm durch eine Klasse repräsentiert. Weiterhin wurden der Visualisierung drei weitere stark aktive Bereiche aus dem PET-Datensatz hinzugefügt. In dieser Visualisierung kann die Größe des Tumors bereits abgeschätzt werden, die Beurteilung der Lage ist jedoch aufgrund der Projektion des dreidimensionalen Volumens auf die Bildebene noch nicht möglich.

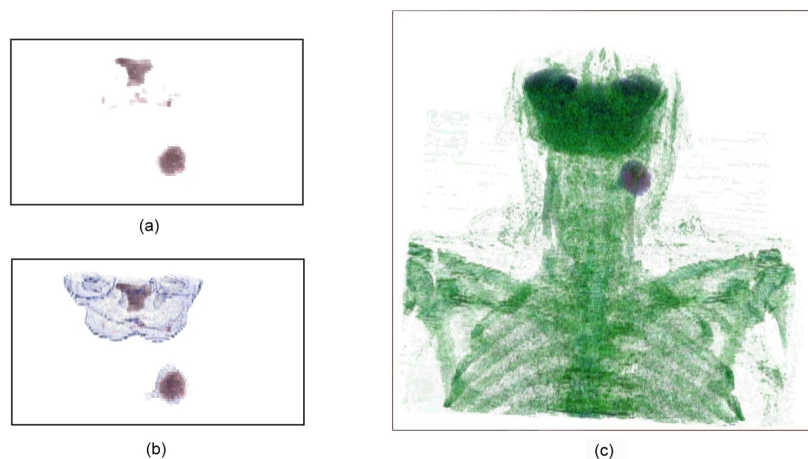


Abbildung 6.5.: Die Abbildungen zeigen die coronale Ansicht auf die aktiven Regionen aus dem PET-Datensatz ((a) und (b)) und die dazu eingblendete Komplementärinformation aus dem CT-Datensatz((c)).

Erst durch die Änderung der Blickrichtung und der damit verbundenen räumlichen Information kann die aktive Struktur als Lymphknotentumor identifiziert werden. Abbildung 6.6 verdeutlicht dies durch eine Ansicht von hinten links auf das Volumen. 6.6(a) zeigt die Strukturen, welche durch die drei Regionen mit den höchsten Intensitätswerten aus dem Histogramm des PET-Datensatzes repräsentiert werden. Diesen wurde in Abbildung 6.6(b) die anatomische Information aus dem CT-Volumen hinzugefügt.

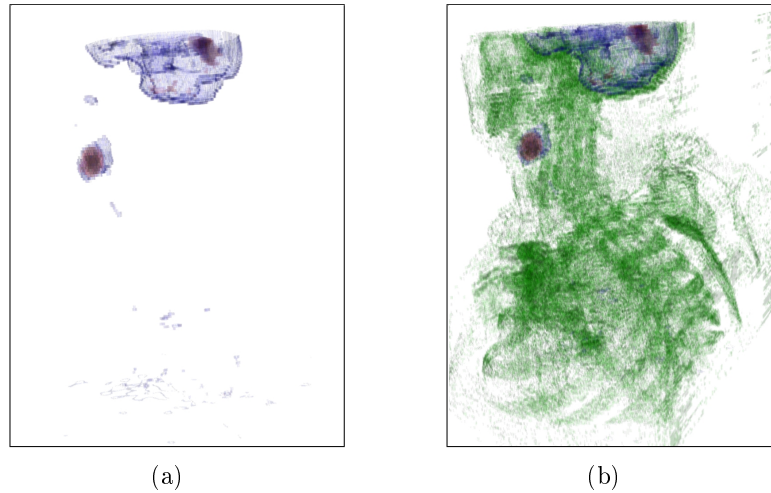


Abbildung 6.6.: Ansicht von hinten links auf die 3D-Visualisierung der CT- und PET-Thoraxdatensätze. Durch die Wahl einer weiteren Perspektive kann die Lage der interessierenden Strukturen besser identifiziert werden.

Die Zuordnung der Lage einer interessierenden Struktur wird anhand der drei Visualisierungen aus Abbildung 6.7 nocheinmal verdeutlicht. Diese soll anhand ausgewählter Blickrichtungen auf das Volumen den interaktiven Analyseprozess einer 3D-Volumenvisualisierung verdeutlichen. Während in Abbildung 6.7(a) der Eindruck entstehen könnte, der Tumor befindet sich im linken vorderen Teil des Gehirns, so ermöglicht ein Blick auf Abbildung 6.7(b) die Lagezuordnung entlang der z-Achse. Die Lage in der y-Richtung (anterior-posterior) und die Position zum Unterkiefer werden aus Abbildung 6.7(c) deutlich. Die interaktive Betrachtung einer 3D-Volumenvisualisierung ist demnach eine effektive Möglichkeit für die Kommunikation von Diagnoseergebnissen.

Mit dieser Bildreihe wurde die Anforderung, eine Möglichkeit für die Visualisierung von Thoraxdaten zu bieten, durch die gemeinsame Darstellung eines CT- und eines PET-Volumendatensatzes erfüllt. Es wurde gezeigt, dass eine multimodale Volumenvisualisierung der Daten besser für die Kommunikation von Diagnoseergebnissen geeignet ist als eine 2D-Schichtbilddarstellung.

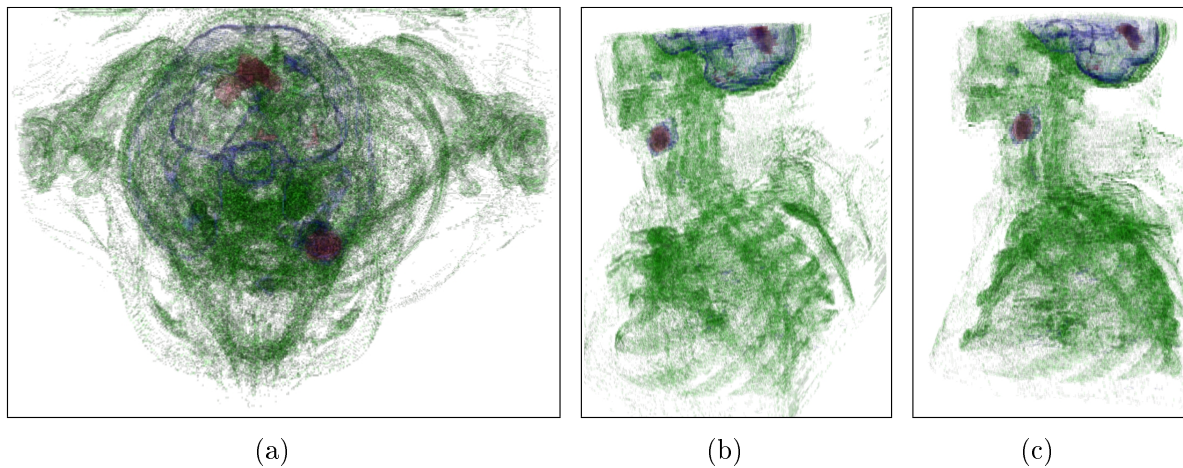


Abbildung 6.7.: Die dargestellten drei Blickrichtungen auf das Volumen (von Oben, Hinten links oben und von links) ermöglichen eine Lagezuordnung und verdeutlichen den Effekt einer interaktiven Exploration der multimodalen 3D-Visualisierung.

6.3. Darstellung von Strukturen mit und ohne anatomischen Kontext

In diesem Abschnitt werden die multiintensitätsbasierten Visualisierungen zweier Schädel Datensätze genannt „CT MRT Head 3“ vorgestellt. Ziel dieser Visualisierungen ist die Darstellung der Ventrikel aus dem MRT-Datensatz im Zusammenhang mit der Knochenstruktur aus dem CT-Volumen. Ventrikel sind wassergefüllte Hohlräume im Gehirn, die z.B. für den Abtransport von Schadstoffen aus dem Gehirn verantwortlich sind. Die starke Vergrößerung der Ventrikel in diesen Datensätzen stellt eine krankhafte Veränderung dar.

In den verwendeten CT-Daten heben sich die Ventrikel kaum vom umgebenden Weichgewebe ab. Für die Darstellung der Ventrikel muss eine geeignete Weichteilfensterung gewählt werden, bei der die Knochenstrukturen allerdings nicht mehr im Detail unterscheidbar sind, da die hellen Bereiche des Bildes auf den maximalen Grauwert abgebildet werden. Soll die Knochenstruktur detailliert dargestellt werden, sind die Ventrikel nicht mehr sichtbar (s. Abb. 6.8(a) und 6.8(c)).

Die vorliegenden MRT-Daten wurden mit einer T2 Wichtung aufgenommen, wodurch sich die wasserhaltigen Ventrikel als helle Strukturen gut vom umgebenden Gewebe abheben. Ebenfalls gut erkennbar ist im MRT das Gehirn, welches die Ventrikel umgibt. Bei der alleinigen Betrachtung der MRT-Daten wäre eine Einordnung der Lage und Größe im Verhältnis zum Schädelknochen des Patienten allerdings nicht möglich, da Knochenstrukturen nicht abgebildet werden (s. Abb. 6.8(b) und 6.8(d)).

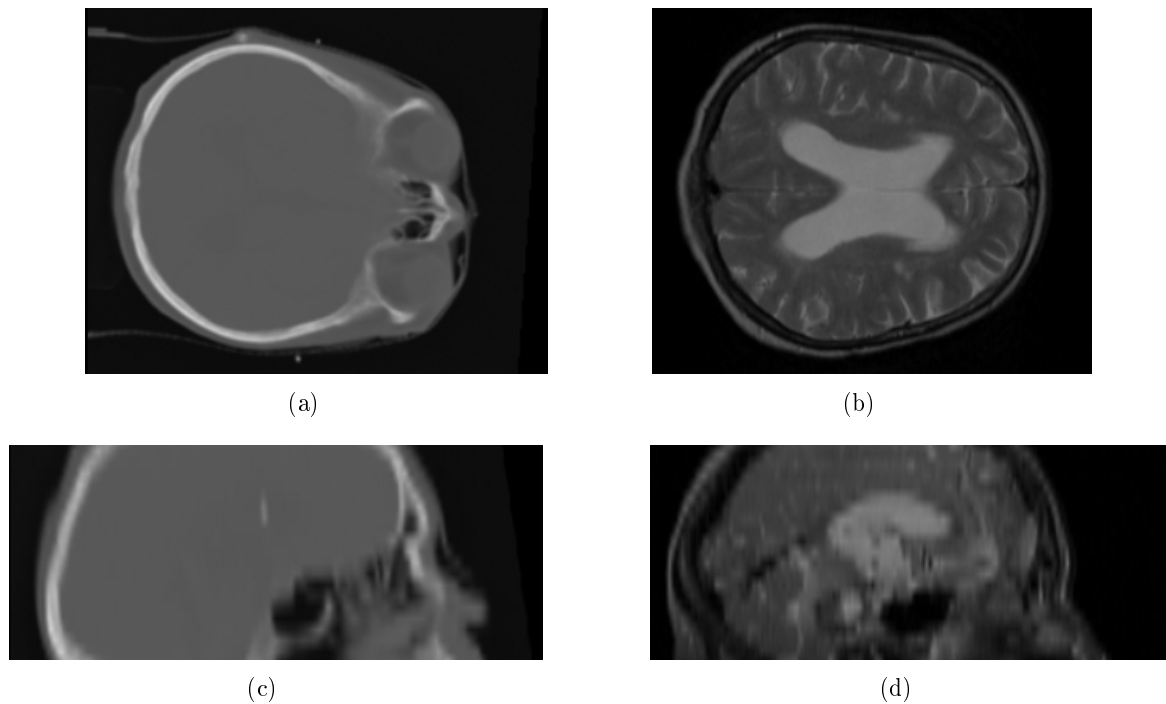


Abbildung 6.8.: Axiale und sagittale Schichtbilder der visualisierten CT- und MRT-Schädel Datensätze.

Um die Ventrikel mit den Knochenstrukturen gemeinsam darstellen zu können, wurde eine Visualisierung mit dem multiintensitätsbasiertem Ansatz unter Verwendung einer STF erstellt. Das verwendete Intensitäts-/ Intensitäts-Histogramm wurde mit einem Radius von 200 mm klassifiziert. Dieser Radius eignet sich sehr gut für die Darstellung von Schädel Datensätzen, da er ungefähr der Ausdehnung des Schädels eines Erwachsenen entspricht.

In Abbildung 6.9 werden die Ventrikel in einer axialen (a) und sagittalen (b) 3D-Ansicht darstellt. Da für die Generierung der Visualisierung eine TF über die Intensitätswerte der beiden Datensätze spezifiziert wurde, stellt die Abbildung die Ventrikel so dar, wie sie in beiden Datensätzen vorhanden sind. Die Ventrikel werden durch 3 von 47 Klassen des Histogramms repräsentiert. In der Abbildung wird ersichtlich, dass es trotz der Verwendung einer 2D-Transferfunktion nicht ohne weiteres möglich ist, interessierende Strukturen allein darzustellen. So besitzen Voxel in der Augenregion in beiden Datensätzen die gleichen Eigenschaften wie die interessierenden Ventrikel und werden daher in der Visualisierung mit dargestellt. Durch die Reduktion des Radius wäre es möglich diese Strukturen getrennt voneinander darzustellen, allerdings müssten dann aus dem Histogramm weitaus mehr Bereiche für die Darstellung der Ventrikel ausgewählt werden.

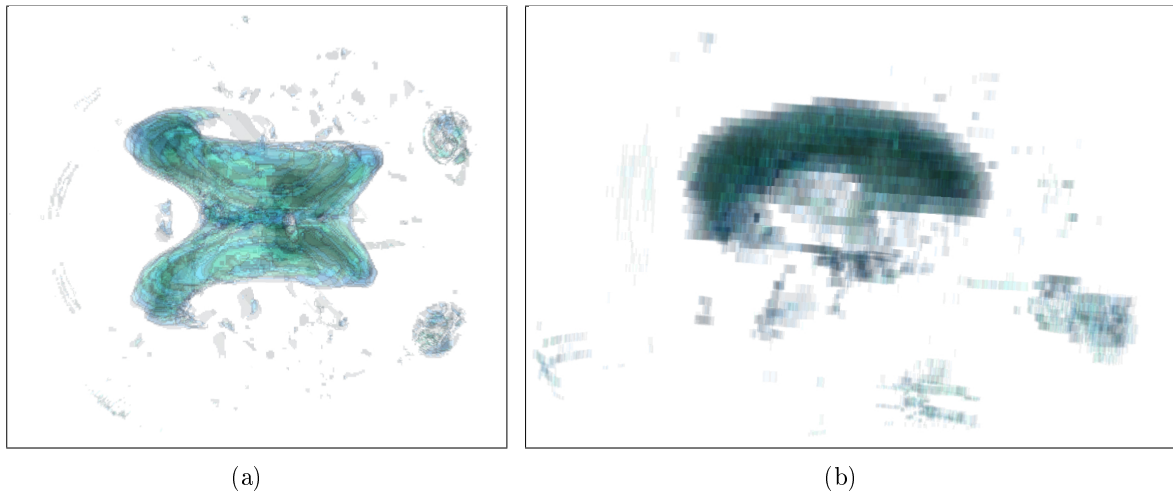


Abbildung 6.9.: Dreidimensionale Darstellung der vergrößerten Ventrikel aus dem CT- und MRT-Datensatz.

Um die Größe und Lage der Ventrikel im Schädel abschätzen zu können, wurden die Visualisierungen aus Abbildung 6.10 um die Knochenstruktur erweitert. Es wurde den Ventrikeln Kontextinformation hinzugefügt und somit das Größenverhältnis der Ventrikel zum Schädel für den Betrachter abschätzbar gemacht, woraus auf die Schwere der krankhaften Veränderung geschlossen werden kann. Insbesondere in der sagittalen Ansicht ist die Größe des Ventrikels gut auszumachen. Die dargestellte Knochenstruktur ist durch 5 Klassen des Histogramms repräsentiert. Somit wurden für die Erstellung dieser Visualisierung 8 Bereiche aus dem Histogramm ausgewählt.

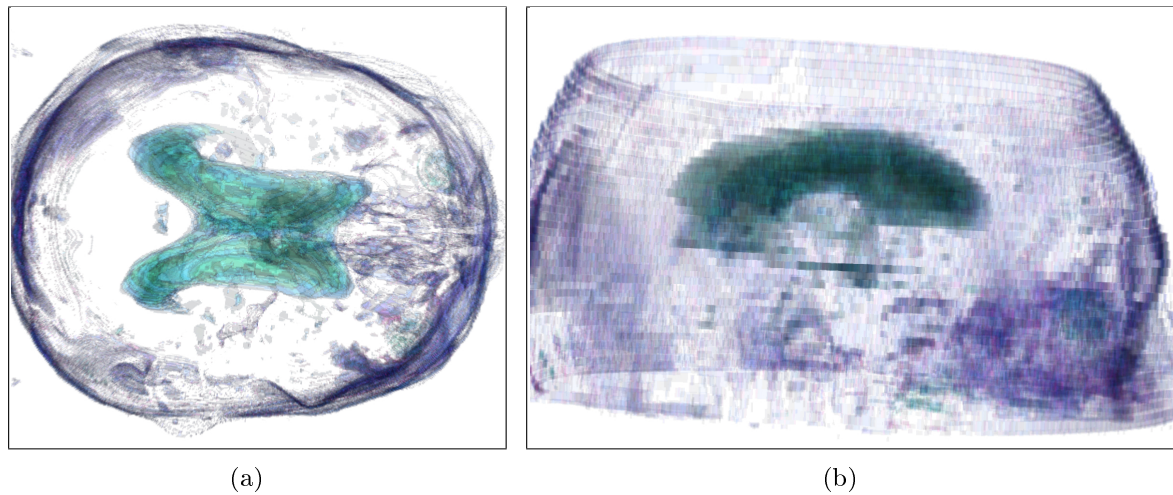


Abbildung 6.10.: Darstellung der vergrößerten Ventrikel mit der Knochenstruktur aus dem CT- und MRT-Datensatz. Durch die hinzugefügte Knochenstruktur wird es möglich die Lage und Größe der Ventrikel abzuschätzen.

In Abbildung 6.11 wurde durch Auswahl eines weiteren Bereiches aus dem klassifizierten Histogramm die Hirnstruktur zur Visualisierung hinzugefügt, sodass die Größe und Lage der Ventrikel zum Gehirn noch besser abschätzbar ist.

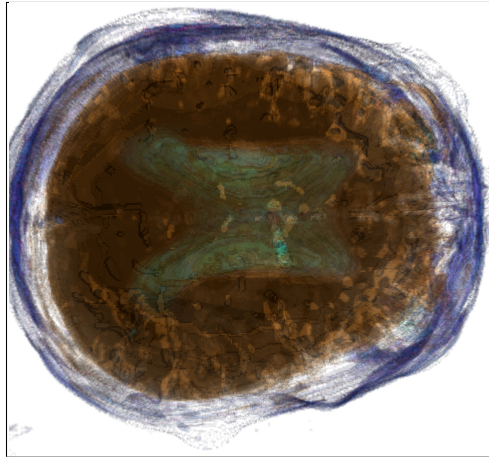


Abbildung 6.11.: Darstellung der Hirnstruktur mit den vergrößerten Ventrikeln und der Knochenstruktur aus dem CT- und MRT-Datensatz.

Die Zuordnung der Bereiche des klassifizierten Histogramms zu den dargestellten Strukturen wird aus Abbildung 6.12 ersichtlich. Diese Abbildung zeigt das 2D-Histogramm, welches als Grundlage für die Spezifikation der verwendeten STF diente. In der Histogrammdarstellung wurden die durch den Nutzer ausgewählten Bereiche heller dargestellt und somit hervorgehoben. Die Bereiche und die dazugehörigen Strukturen in der Visualisierung sind durch Pfeile gekennzeichnet.

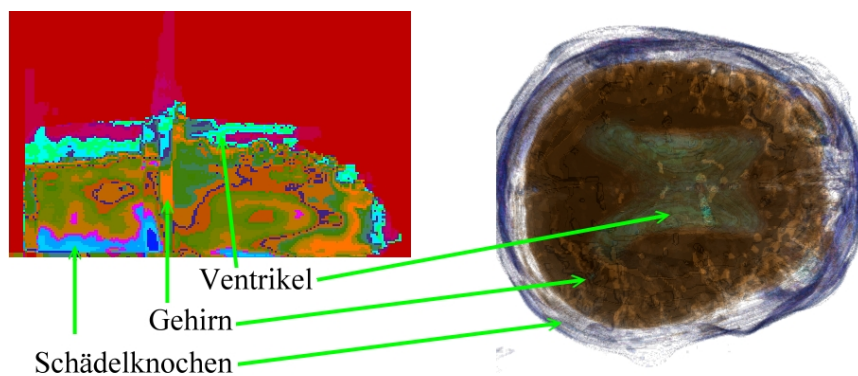


Abbildung 6.12.: Den drei visualisierten Strukturen werden in dieser Abbildung die dazugehörigen Bereiche des zweidimensionalen Intensitäts-/ Intensitäts-Histogramms zugeordnet.

Mit dieser Bildreihe wurden folgende Anforderungen erfüllt: Zum einen erfolgte die Darstellung von CT- und MRT-Datensätzen des Schädels. Weiterhin wurden Strukturen einzeln und mit anatomischer Kontextinformation dargestellt. Auch die Darstellung komplementärer Information wurde durch die Visualisierung der Ventrikel in Verbindung mit der Knochenstruktur erreicht.

6.4. Vergleich von Visualisierungen mit und ohne Überabtastung

Dieser Abschnitt verwendet die zwei miteinander registrierten PET- und MRT-Datensätze aus „CT MRT PET“, um anhand von multiintensitätsbasierten Visualisierungen die Auswirkung der Überabtastung zu untersuchen. Die beiden verwendeten Datensätze bilden den Schädel eines Patienten ab. Das dazugehörige Histogramm wurde mit einem Radius von 200 mm klassifiziert und enthält bei einer Visualisierung ohne Überabtastung 40 Klassen. Durch die Überabtastung wird die Anzahl der Voxel in jede Dimension verdoppelt und das Spacing halbiert, wodurch das resultierende Histogramm in 42 Klassen unterteilt wird.

Für die Visualisierungen in den Abbildungen 6.13(a) und (b) wurden aus dem klassifizierten Histogramm der nicht überabgetasteten Datensätze neun Bereiche ausgewählt, um den schwach aktiven Teil des Gehirns (fünf Klassen) zusammen mit der Haut (vier Klassen) darzustellen. Die dargestellten Datensätze haben eine räumliche Auflösung von 0,78 mm x 0,78 mm x 5,62 mm. Auf die Qualität der abgebildeten Visualisierungen wirkt sich vor allem die große Schichtdicke aus. Während die Struktur der Oberfläche des dargestellten Hirnteils in der axialen Ansicht 6.13(a) gut zu erkennen ist, bilden sich in der sagittalen Ansicht 6.13(b) starke Treppenartefakte heraus.

Die Qualität der Visualisierungen verbessert sich stark, wenn eine Überabtastung der Datensätze durchgeführt wird. Die resultierenden Visualisierungen, für die acht Klassen aus dem erstellten 2D-Histogramm ausgewählt wurden, zeigen die Abbildungen 6.13(c) und 6.13(d). Von den acht gewählten Bereichen entfallen jeweils vier auf Gehirn und Haut. Durch die Überabtastung ergibt sich eine räumliche Auflösung von 0,39 mm x 0,39 mm x 2,81 mm. Die Qualitätsverbesserung wird vor allem in der sagittalen Ansicht aus Abbildung 6.13(d) ersichtlich, in der die Treppenartefakte stark reduziert wurden. Weiterhin ergibt sich aus der geringeren Schichtdicke eine stärkere Transparenz der dargestellten Regionen (s. Abb. 6.13(a) und 6.13(c)). Begründen lässt sich dies mit dem *Ray Casting* und der damit verbundenen Abtastung entlang des Strahls, welcher durch das Volumen gesendet wird. Im Folgenden soll dies an einem Beispiel verdeutlicht werden. Verläuft ein Strahl beim Raycasting senkrecht durch eine einzelne Schicht mit einer Dicke d , so werden die n Transparenzwerte der Abtastpunkte innerhalb der Schicht für die Färbung des betreffenden Pixels herangezogen. Für eine Schicht mit der Dicke $d/2$ werden die Transparenzwerte von $n/2$ Abtastpunkten für die Berechnung des Pixelwertes herangezogen, wodurch sich eine stärkere Transparenz ergibt.

Da sich mit der Überabtastung die zu verarbeitende Datenmenge verachtfacht, erhöhen sich dementsprechend auch die Berechnungszeiten. Ohne Überabtastung dauert es auf dem am Anfang des Kapitels genannten Computer 9,25 Sek. bis das klassifizierte Histogramm berechnet wurde und die Spezifikation der multiintensitätsbasierten STF durch den Nutzer beginnen kann. Wird eine Überabtastung durchgeführt, so verlängert sich die Rechenzeit auf ca. 38 Sek.. Die Überabtastung ist vor allem bei schlecht aufgelösten Datensätzen empfehlenswert, da sich die Qualität der Visualisierungen stark verbessert. Die verlängerten Rechenzeiten liegen dann im Bereich von gut aufgelösten Datensätzen und können somit toleriert werden (vgl. Tabelle A.2).

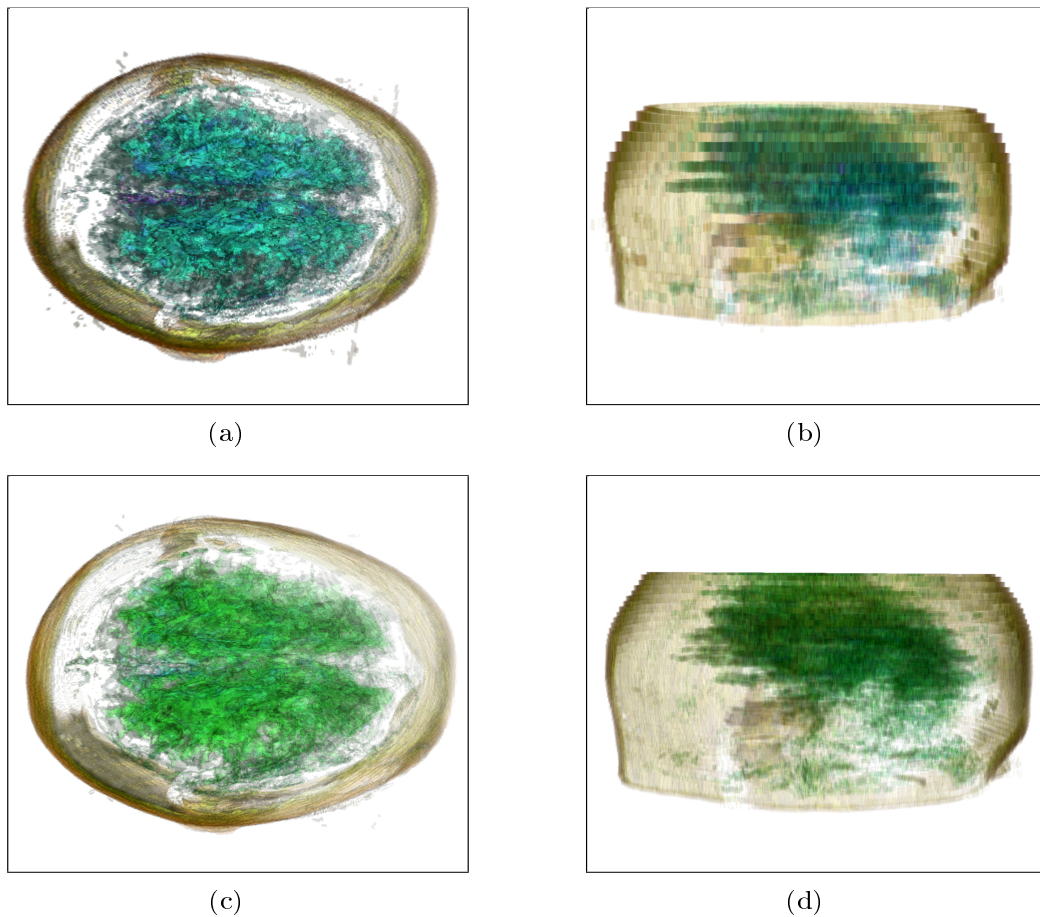


Abbildung 6.13.: Vergleich des Effekts der Überabtastung am schwach aktivierten Bereich des Gehirns anhand einer multiintensitätsbasierten Visualisierung zweier MRT- und PET- Datensätze. Die Überabtastung reduziert die Treppenartefakte, welche sich aus der großen Schichtdicke ergeben (s. (b) und (d)) und verbessert die transparente Darstellung der Regionen (s. (a) und (c)).

Die zuvor beschriebenen Effekte sind auch in den Visualisierungen aus Abbildung 6.14 erkennbar. Diese zeigen den stark aktiven Teil des Gehirns mit der Haut des Patienten. Für die dargestellten Visualisierungen wurden sowohl ohne als auch mit Überabtastung sechs Klassen aus den dazugehörigen Histogrammen ausgewählt. Der stark aktive Bereich des Gehirns wird dabei durch jeweils zwei Bereiche repräsentiert.

Im Folgenden wird näher auf die Auswirkungen der Überabtastung auf die Klassifikation der Histogramme eingegangen. Die 2D-Histogramme, welche die Grundlage für die Visualisierungen in den Abbildungen 6.13 und 6.14 bilden sind in Abbildung 6.15 dargestellt. Den Bereichen werden die dazugehörigen Strukturen aus der Visualisierung zugeordnet. Im Vergleich der beiden Abbildungen fällt auf, dass die Klassifikation des Histogramms durch die Überabtastung homogener wird. Vor allem im oberen Teil des Histogramms, welcher die Voxel mit hohen Gradientenbeträgen repräsentiert ist der Effekt deutlich zu erkennen. Die Bereiche grenzen sich eindeutiger voneinander ab und werden kompakter, da die Bins einer Klasse nicht mehr verstreut über das Histogramm

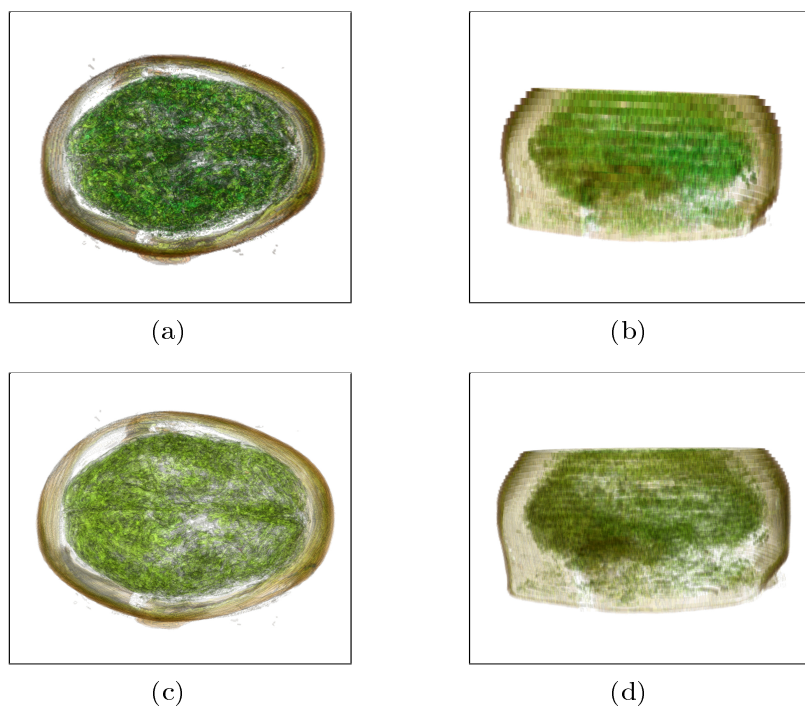


Abbildung 6.14.: Vergleich des Effekts der Überabtastung am stark aktivierten Bereich des Gehirns anhand einer multiintensitätsbasierten Visualisierung der Datensätze aus Abbildung 6.13. Die Überabtastung reduziert die Treppenartefakte, welche sich aus der großen Schichtdicke ergeben (s. (b) und (d)) und verbessert die transparente Darstellung der Regionen.

vorliegen. Dies erleichtert dem Nutzer die Zuordnung von Bereichen zu anatomischen Strukturen, anhand der Intensitätseigenschaften. Die Verbesserung der Klassifikation erklärt sich durch die Reduktion des Rauschens im Histogramm, da wesentlich mehr Voxel in das Histogramm eingetragen werden und so die Ortsinformation der Bins repräsentativer wird.

An diesem Beispiel wird deutlich, wann der multiintensitätsbasierte Ansatz für die Visualisierung genutzt werden kann. Die im PET-Datensatz enthaltene Information über die Aktivität der Regionen wird durch den Weichgewebekontrast des MRT-Datensatzes unterstützt. Die verschiedenen Bereiche, welche Strukturen im Volumen repräsentieren, sind über die gesamte Höhe des Histogramms verteilt. Hierdurch wird bei der Klassifikation die Trennung der aktivierten Regionen vom umgebenden Gewebe möglich. Im Gegensatz dazu steht die multiintensitätsbasierte Visualisierung eines Ganzkörperdatensatzes aus Abschnitt 6.5.2. Im direkten Vergleich der beiden Histogramme in Abbildung 6.16 wird der Unterschied in der Klassifikation deutlich.

Mit dieser Bildreihe wurde die Anforderung, eine TF zu generieren, die auf zwei Datensätze angewendet werden kann und die deren Intensitätswerte als Parameter nutzt, durch die gemeinsame Darstellung eines MRT- und eines PET-Volumendatensatzes erfüllt. Es wurde gezeigt, dass es auch bei dieser Art von Visualisierung möglich ist, Kontextinformation in Form von anderen anatomischen Strukturen zur Visualisierung hinzuzufügen.

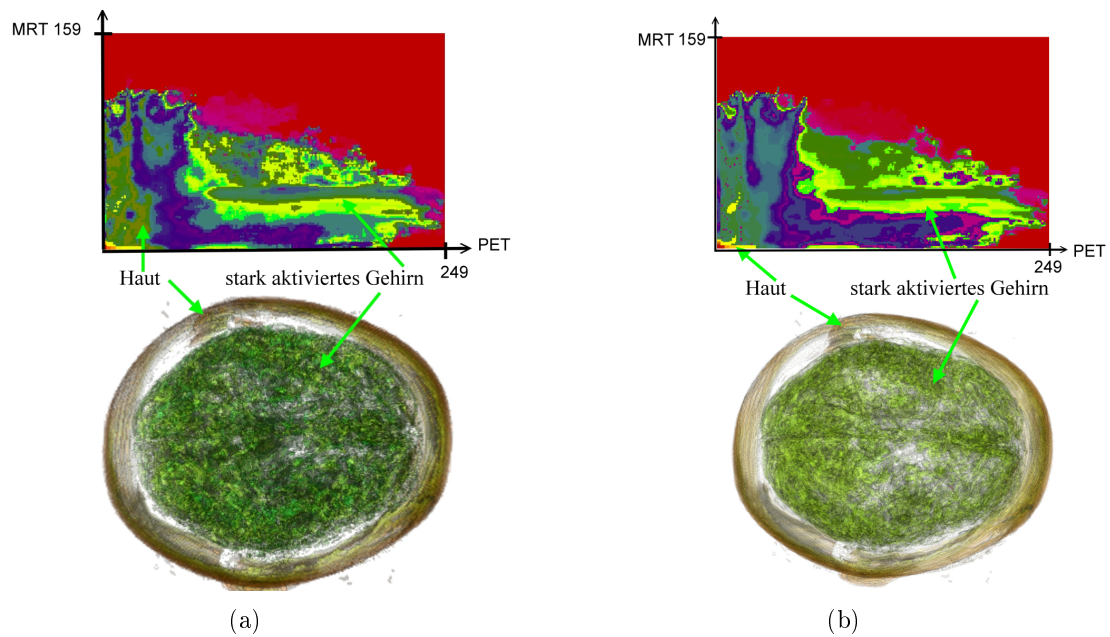


Abbildung 6.15.: Vergleich der klassifizierten Histogramme bei Überabtastung.

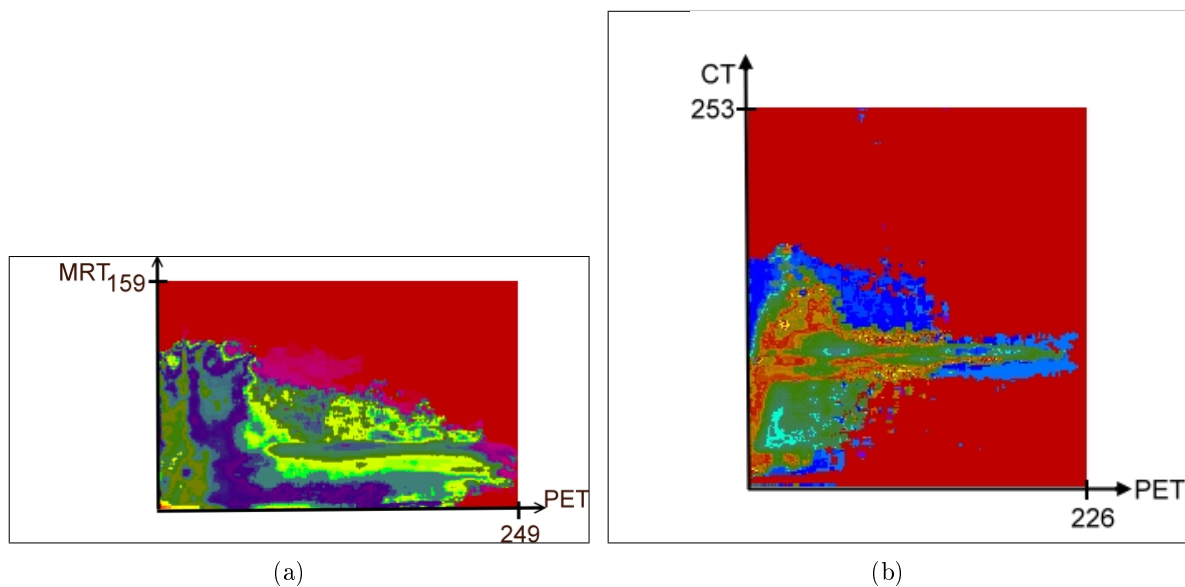


Abbildung 6.16.: (a): Das zweidimensionale Intensitäts-/ Intensitäts-Histogramm des MRT/PET-Schädeldatensatzes. (b): Das zweidimensionale Intensitäts-/ Intensitäts-Histogramm eines CT/PET-Ganzkörperdatensatzes. Der Bereich auf der MRT-Achse, in den die PET-Intensitätswerte sortiert werden, umfasst sehr viel mehr Grauwerte als im CT/PET Histogramm. Hierdurch können die PET-Informationen verschiedenen Geweben im MRT-Datensatz zugeordnet und getrennt dargestellt werden.

6.5. Vergleich multiintensitätsbasierte und gradiententenbasierte Visualisierung

Dieser Abschnitt beschäftigt sich mit dem Vergleich von Visualisierungen, die mit den zwei umgesetzten Arten der STF-Spezifikation erstellt wurden. Als Grundlage dient der Ganzkörperdatensatz „CT PET Body“ bei dem ein CT- und ein PET-Volumen miteinander registriert wurden. Im CT-Datensatz ist eine auffällige Struktur im rechten Lungenflügel erkennbar, die durch die PET-Aufnahme als Tumor identifiziert werden kann.

Die vorliegenden Ausgangsdaten sind in Abbildung 6.17 in Form von coronalen und sagittalen Schichtbildern dargestellt. Diese zeigen jeweils zusammengehörige Schichten in den beiden Datensätzen. Die abgebildeten CT-Schichtbilder wurden so gefenstert, dass das Weichgewebe gut erkennbar ist. Dadurch wird diffuses Gewebe im Inneren der Lunge sichtbar. Die PET-Aufnahme zeigt mehrere aktivierte Bereiche, deren Lage im PET-Volumen allerdings keinen anatomischen Strukturen zugeordnet werden kann. Um diesen Nachteil auszugleichen, wurden die beiden Volumina miteinander registriert und sollen gemeinsam visualisiert werden. Im PET-Datensatz sind neben dem Tumor noch weitere helle und somit aktive Bereiche erkennbar.

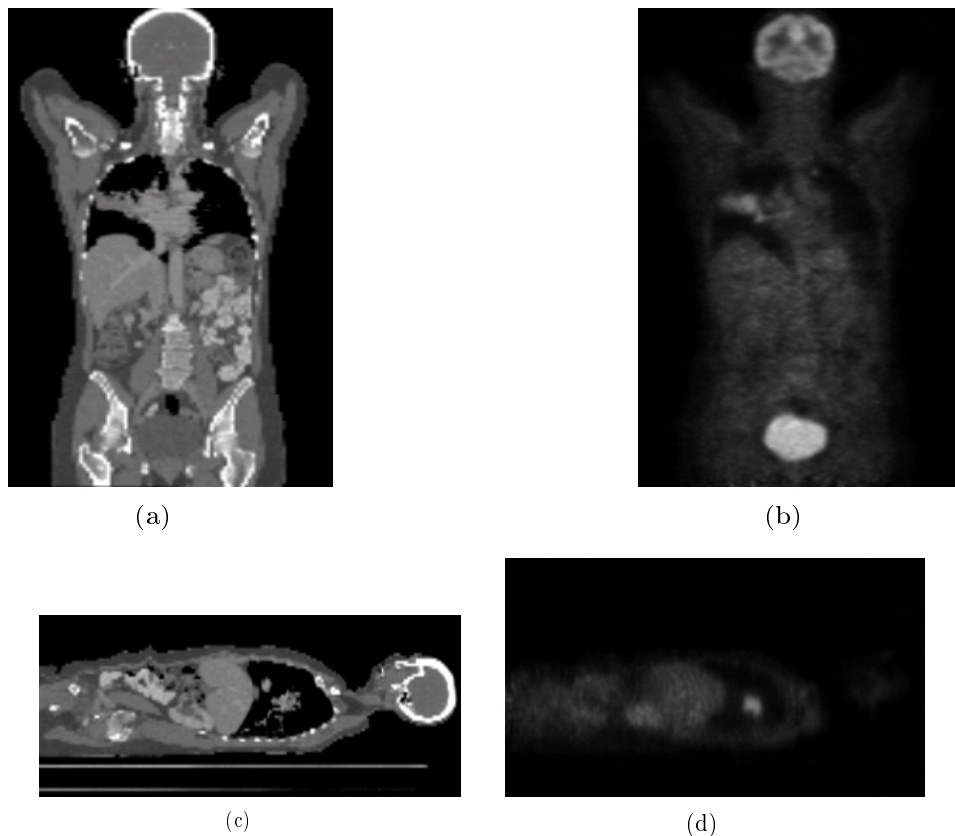


Abbildung 6.17.: Sagittale und coronale Schichtbilder der visualisierten CT- und PET-Ganzkörperdatensätze.

Die im folgenden beschriebenen Volumenvisualisierungen wurden sowohl mit multiintensitätsbasierten als auch gradientenbasierten STFs erzeugt. In beiden Fällen wurde ein Klassifikationsradius von 1155 mm verwendet, welcher an die Größe der Datensätze angepasst ist (Körperlänge vom Scheitel bis zur Hüfte). Aus diesem Radius ergeben sich 95 Klassen für das multiintensitätsbasierte Histogramm und 110 für das Intensitäts-/ Gradientenstärke-Histogramm des CT-Datensatzes bzw. 78 Klassen für das PET-Volumen. Für die gezeigten Visualisierungen wurden die Datensätze in der Vorverarbeitung mit dem Gaußfilter geglättet. Hierdurch war es möglich den Tumor aus dem PET-Datensatz besser zu extrahieren, da die Voxelwerte geglättet wurden und somit die Voxel, welche den Tumor repräsentieren, benachbarten Bins im Histogramm zugeordnet wurden. Die Zeiten für die Generierung des Histogramms mit Gaußfilterung hat sich bei diesen Datensätzen um ca. 2 Sek. erhöht. Ob die Gaußfilterung einen generellen Vorteil für die Darstellung bringt, welcher die längere Bearbeitungszeit rechtfertigt, muss anhand von weiteren Datensätzen untersucht werden. Die Eigenschaften der Datensätze sowie Messzeiten werden in Tabelle [A.2](#) erfasst.

6.5.1. Visualisierung mit gradientenbasierten STFs

In [Abbildung 6.18](#) werden die gradientenbasierten Visualisierungen der CT- und PET-Datensätze dargestellt. [Visualisierung 6.18\(a\)](#) zeigt die aktiven Regionen aus dem PET-Datensatz als violette Strukturen. Für diese Visualisierung mussten 19 Bereiche aus dem klassifizierten Histogramm des PET-Datensatzes zur betreffenden STF hinzugefügt werden. Der dargestellte Teil der Haut wird im Histogramm des CTs durch eine Klasse repräsentiert. Nachdem diese zur STF des CT-Datensatzes hinzugefügt wurde, kann das CT-Volumen entsprechend visualisiert werden. Die gezeigte Visualisierung ergibt sich aus der *Color Fusion* der beiden Datensätze. Nachdem Teile der Haut als komplementäre Information aus dem CT-Datensatz zur [Visualisierung 6.18\(b\)](#) hinzugefügt wurden, können die aktiven Regionen als Gehirn, Herz und Blase sowie ein Tumor identifiziert werden. Eine genauere Lagezuordnung wird möglich, wenn aus dem CT-Datensatz innere Strukturen eingeblendet werden (s. [Abb. 6.18\(c\)](#)). Dabei wird ersichtlich, dass sich der Tumor innerhalb des rechten Lungenflügels befindet. Die dargestellten inneren Strukturen werden im CT-Histogramm durch zwei Klassen repräsentiert. Für die gezeigte Visualisierung wurde die Haut wieder aus der STF des CT-Volumens entfernt und die beiden Klassen hinzugefügt. Durch die unterschiedliche Einfärbung der Strukturen und die Verwendung einer globalen, über die Skalarwerte einheitlichen Transparenz bleibt der Tumor auch bei der Darstellung der Lunge als weitere komplementäre anatomische Struktur sichtbar.

[Abbildung 6.19](#) zeigt eine sagittale Ansicht auf die Visualisierung der beiden Datensätze. Durch die unterschiedlichen Blickrichtungen auf die Datensätze ist es möglich, die Lage der Strukturen innerhalb des Körpers besser abzuschätzen. Dies ist einer der Hauptvorteile einer 3D-Visualisierung, welche interaktiv um die drei Raumachsen rotiert werden kann. Die Darstellung eines Teils der Haut erleichtert die räumliche Zuordnung. Für die erzeugten Visualisierungen wurden die gleichen Klassen aus den Histogrammen gewählt wie für [Abbildung 6.18](#).

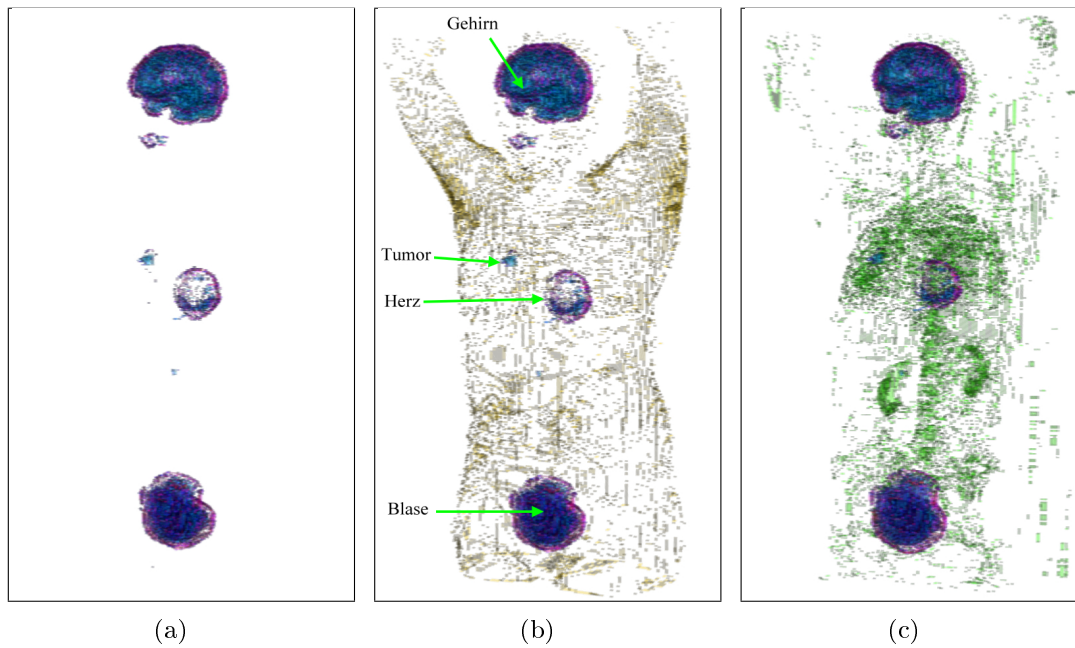


Abbildung 6.18.: Coronale Ansicht auf die gradientenbasierten Visualisierungen der CT- und PET-Ganzkörperdatensätze.

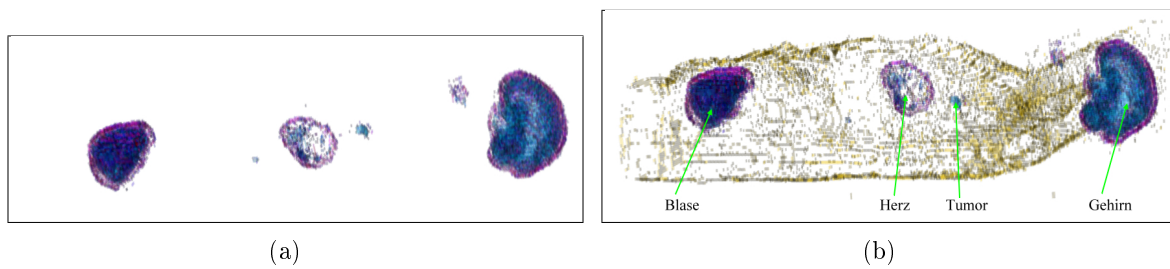


Abbildung 6.19.: Sagittale Ansicht auf die gradientenbasierten Visualisierungen der CT- und PET-Ganzkörperdatensätze.

6.5.2. Visualisierung mit multiintensitätsbasierter STF

Für den multiintensitätsbasierten Ansatz wurde ein gemeinsames Histogramm für den PET- und den CT-Datensatz erstellt. Für die in Abbildung 6.20 gezeigten Visualisierungen wurde der gleiche Klassifikationsradius wie im vorhergehenden Abschnitt für die gradientenbasierte Visualisierung verwendet. In Abbildung 6.20(a) werden Teile der Haut und die aktiven Bereiche aus dem erzeugten Volumen mit den beiden Intensitätswerten der Datensätze dargestellt. Dazu wurden im klassifizierten Histogramm 11 Bereiche ausgewählt. Der Tumor wird in dieser Visualisierung jedoch nicht abgebildet. Auch als weitere Klassen aus dem Histogramm zur Visualisierung hinzugefügt wurden, gelang es nicht, den Tumor getrennt vom umgebenden Gewebe darzustellen. Durch die Auswahl der Klassen, welche die stärksten Intensitäten aus dem PET-Datensatz repräsentieren, wurde in jedem Fall auch Lungengewebe mit dargestellt (s. Abb. 6.20(b)). Ebenso wurde ohne Erfolg versucht die Anzahl der Klassen durch die Wahl eines geringeren Klassifikationsradius zu erhöhen und so ggf. das Tumorgewebe vom Lungengewebe zu trennen.

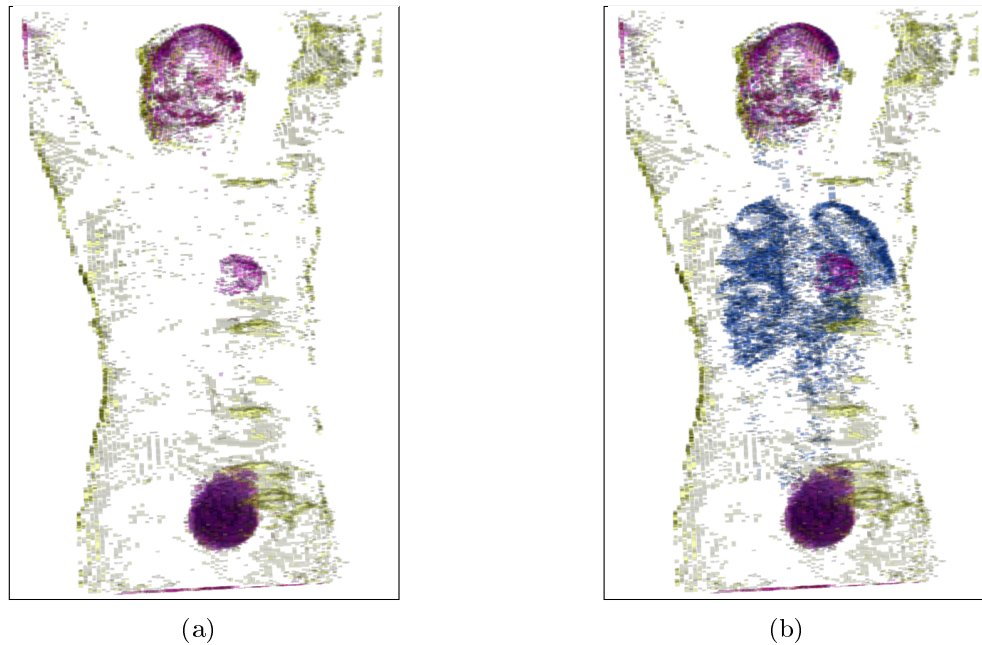


Abbildung 6.20.: Coronale Ansicht auf die multiintensitätsbasierten Visualisierungen der CT- und PET-Ganzkörperdatensätze. Der Tumor in der Lunge ist nicht erkennbar.

Aus dem Vergleich der Visualisierungen aus Abbildung 6.20 und Abbildung 6.18 aus dem vorangegangenen Abschnitt wird die Schwäche des multiintensitätsbasierten Ansatzes erkennbar. Die getrennte Darstellung der aktiven Regionen aus dem PET-Datensatz und des umgebenden Gewebes gelingt aus mehreren Gründen nicht. Zum einen wird ein großer Teil der Gewebe im CT-Datensatz durch ähnliche Grauwerte repräsentiert. Hierdurch wird fast der gesamte Intensitätsbereich des PET-Datensatzes auf einen kleinen Teil in der Mitte des CT-Intensitätsumfangs abgebildet. Dies wird aus Abbildung 6.16(b) ersichtlich. Zum anderen ist der Lungentumor über dazugehöriges Gewebe mit dem Mediastinum verbunden, welches im PET-Datensatz fast die gleiche Helligkeit besitzt wie der Tumor.

In diesem Fall ist der Unterschied zwischen der multiintensitätsbasierten und den gradientenbasierten STF's gut erkennbar. Der Vorteil der gradientenbasierten STF's liegt darin, dass sie durch Verwendung der Gradienteninformation Strukturen entlang ihrer Grenzen ähnlich wie bei den Isooberflächen extrahieren kann. Durch die verwendete Ortsinformation wird eine gemeinsame Klassifikation immer unwahrscheinlicher, je weiter die Schwerpunkte zweier Bins auseinander liegen. Treten nun häufig annähernd gleiche Grauwerte auf, so wird die Ortsinformation ungenauer und so die Trennung der Gewebe erschwert. Es scheint, als ob die gradientenbasierten STF's gegen diesen Effekt weniger anfällig sind als die multiintensitätsbasierten STF's.

Mit dieser Bildreihe wurden folgende Anforderungen erfüllt: Darstellung von PET-Volumendaten. Visualisierung von Ganzkörperaufnahmen im Vergleich multiintensitätsbasierter und gradientenbasierter STF's. Zudem wurde erneut die Anforderung, komplementäre Information darzustellen, erfüllt.

7. Diskussion und Ausblick

Erreichte Ziele

In dieser Arbeit wurde ein Visualisierungssystem für die parallele Darstellung multimodaler medizinischer Volumendaten entworfen. Die implementierte Visualisierungskomponente basiert auf den frei verfügbaren Programmbibliotheken ITK, VTK und wxWidgets und wurde erfolgreich in die am Fraunhofer IGD entwickelte Bildverarbeitungssoftware MIP eingebunden.

Um die Anforderungen an die Visualisierungskomponente zu erfüllen, wurden zwei Ansätze für die direkte Volumenvisualisierung durch Farb- und Transparenztransferfunktionen vorgeschlagen und implementiert. Zunächst wurde, basierend auf den von Roettger et al. [RBS05] entwickelten *Spatialized Transfer Functions*, ein multiintensitätsbasierter Ansatz implementiert und auf die Anwendbarkeit für multimodale Visualisierungen untersucht. In diesem Ansatz wird der Nutzeraufwand für die Spezifikation durch die Verwendung einer gemeinsamen TF für die Darstellung beider Datensätze vermindert. Daneben wurde mit dem gradientenbasierten Ansatz eine Möglichkeit umgesetzt und untersucht, die Datensätze durch die Anwendung jeweils einer TF abzubilden und anschließend die Visualisierungen zu fusionieren. Der Vorteil des gradientenbasierten Ansatzes liegt, neben der besseren Anpassung an den Datensatz durch die Verwendung der Gradienteninformation, in der schnellen Aktualisierung bei Änderungen der Registrierung. In beiden Ansätzen wurde auf die effiziente und intuitiv durchzuführende Spezifikation der STFs zurückgegriffen, welche es dem Nutzer ermöglicht auch mehrere TFs in kurzer Zeit zu generieren.

Es wurden gute Visualisierungsergebnisse mit PET- und MRT-Datensätzen sowohl bei separater als auch multimodaler Darstellung erzielt. In den Visualisierungen können einzelne Strukturen aus den Datensätzen separat dargestellt werden und bei Bedarf besteht die Möglichkeit Kontextinformation oder komplementäre Strukturen aus einem Datensatz anderer Modalität zur Visualisierung hinzuzufügen. Ein weiterer positiver Aspekt ist die Verwendung von trilinearer Interpolation für die Anwendung der Registrierung im multiintensitätsbasierten Fall.

Die Nutzerinteraktion für die Vorverarbeitung reduziert sich auf die Auswahl der auszuführenden Operationen (Gaußfilterung, Überabtastung). Über die Einbeziehung der Ortsinformation in den Generierungsprozess der TF, durch Verwendung von STFs und die damit verbundene Klassifikation des 2D-Histogramms, konnte weiterhin der Spezifikationsaufwand für den Nutzer verringert werden. Daneben ist die Bedienung der Oberfläche zur Definition der STF leicht zu erlernen. In der Darstellung des klassifizierten

Histogramms wird dem Nutzer über die Hervorhebung der ausgewählten Bereiche eine Rückmeldung über das Aussehen der STF gegeben. Die verwendete globale Transparenz kann durch den Nutzer modifiziert werden wodurch ihm eine weitere Möglichkeit gegeben wird die Visualisierung an die eigenen Bedürfnisse anzupassen.

Für die Untersuchung auf die Übertragbarkeit von Klassifikationsergebnissen zur Definition von *Presets* für die Transferfunktionen wurde bereits ein großer Teil der Vorarbeit geleistet, indem die Verarbeitung der Daten standardisiert und weitestgehend normiert erfolgt.

Offene Ziele und Lösungsvorschläge

Die notwendigste Erweiterung des Visualisierungssystems ist die direkte Verwendung der generierten 2D-FTF durch den Renderer. Dies soll in der VTK Version 5.0 umgesetzt werden. Werden 2D-FTFs möglich, so muss die Klassifikation dahingehend geändert werden, dass nicht mehr Klassennummern sondern, wie in [RBS05] beschrieben, Farbwerte an die Bins vergeben werden. Das klassifizierte Histogramm aus dem der Nutzer ihn interessierende Bereiche ausgewählt hat, kann dann direkt als FTF genutzt und zusammen mit dem Volumendatensatz an den Renderer übergeben werden. Hierdurch würde die Aktualisierung der Visualisierung beschleunigt werden. Die Datenstruktur für den effizienten Zugriff auf die Volumenelemente könnte dann wegfallen, wodurch der Speicherbedarf gesenkt wird.

Bisher erfolgt im gradientenbasierten Ansatz keine trilineare Interpolation bei der Transformation des Datensatzes, da in diesem bereits die Klassennummern gespeichert sind. Nach Umstellung auf VTK 5.0 liegen an dieser Stelle in der Pipeline die Grauwerte des Datensatzes vor und können trilinear interpoliert werden.

In der implementierten Visualisierungskomponente konnten keine TTFs basierend auf der Gradientenstärke (*Gradient Opacity*) verwendet werden, da nur die ausgewählten Klassen mit einer 1D-FTF visualisiert werden und die anderen Voxel den Wert 0 besitzen. Somit konnte die Anforderung der Hervorhebung von Objektgrenzen und dem Ausblenden homogener Bereiche bisher nicht erfüllt werden. Dies wird jedoch mit den zuvor genannten Veränderungen möglich.

Weiterhin erfolgt bisher nur die Darstellung der Voxel, welche in die ausgewählten Bereiche des Histogramms fallen. Denkbar wäre in der monomodalen Visualisierung die Darstellung des gesamten Datensatzes in Form von Grauwerten unter Verwendung einer gradientenbasierten TTF um nur die Kanten der Strukturen anzuzeigen. Dieser Grauwertvisualisierung werden dann die farbigen interessierenden Strukturen hinzugefügt. Auf diese Weise würde zu den eingefärbten Bereichen der gesamte Datensatz als Kontextinformation angezeigt werden. In der multimodalen Visualisierung ist dies nicht ohne weiteres möglich, da dort mehrere Grauwerte für einen Voxel vorliegen.

Für die Auswahl der Farbwerte ist neben der zufälligen Farbvergabe aus [RBS05] zum einen die Verwendung verschiedener auswählbarer Farbtabelle denkbar. Zum anderen wäre auch die manuelle Farbzuhnung zu einem Bereich eine interessante Möglichkeit die Visualisierung an die eigenen Vorstellungen anzupassen. Ebenso könnten Bereichen eigene Transparenzen zugewiesen werden.

Um dem Nutzer die Spezifikation der STF weiter zu erleichtern, sollte die Interaktivität der Oberfläche um die Auswahl von Voxeln im Schichtbild und die Hervorhebung der betreffenden Positionen im Histogramm erweitert werden (vgl. [Stö04] und [KKH01]). Nach der Auswahl eines Bereiches sollte auch das dargestellte Schichtbild mit der 2D-FTF eingefärbt werden, um dem Nutzer ein weiteres Feedback über die ausgeführte Aktion zu geben. Auch die Erleichterung der Bereichsauswahl z.B. über die Auswahl aller Bereiche innerhalb einer angegebenen rechteckigen Region ist denkbar.

Die entwickelte Visualisierungskomponente könnte unter Verwendung der neuen VTK Version um ein 3D-*Texture Mapping* als Alternative zum *Ray Casting* erweitert werden, wodurch die interaktive Betrachtung der Visualisierung in Echtzeit möglich wird.

Weiterhin sollte untersucht werden, ob die verwendete Klassifikation verbessert werden kann. Denkbar wäre z.B. die Aktualisierung von Schwerpunkt und Varianz eines Bereiches nachdem ein Bin zu diesem Bereich hinzugefügt wurde. In diesem Fall würde für die Normberechnung auch die Ortsinformation des Bereiches und nicht mehr eines einzelnen Referenzbins herangezogen werden. Ansätze dieser Art werden bereits erfolgreich in *Clustering*-Algorithmen angewendet.

Um die Möglichkeiten und Grenzen des vorgestellten Visualisierungssystems näher zu spezifizieren, sollte die Auswirkung der Vorverarbeitung und vor allem der Gaußfilterung auf die beiden implementierten Ansätze untersucht werden. Weiterhin ist die Analyse der klassifizierten 2D-Histogramme vor allem im Hinblick auf die Zuordnung von Klassen zu Gewebetypen ein sinnvoller Ansatz. Dazu müssen die Untersuchungen auf eine größere Menge von Datensätzen ausgeweitet werden. Vor allem sollte die Darstellung von CT-Datensätzen näher betrachtet werden, da hier Probleme bei der Extraktion von Strukturen auftraten.

Die weiterführenden Untersuchungen des 2D-Histogramms könnten auf die Frage der Anwendbarkeit von Referenz-Transferfunktionen ausgeweitet werden. Dabei gilt es herauszufinden, inwieweit eine erzeugte Klassifikation des Histogramms auf Datensätze der gleichen Modalität bei gleicher Fragestellung an das Histogramm eines anderen Datensatzes angepasst werden kann. Weiterhin sollte untersucht werden, ob die Klassenauswahl auf diese Weise automatisiert und somit nach der Klassifikation des Histogramms eine automatische Vorgabe einer Visualisierung erzeugt werden kann.

A. Anhang

Tabellen

	Schichtbilder	Indirekte Volumen-visualisierung	Direkte Volumen-visualisierung
Art und Struktur der Daten	<ul style="list-style-type: none"> aus Volumendaten extrahierte Schichten; 2D-Felder von Skalarwerten alle tomographischen Verfahren Blick auf originale Daten schnelle Übersicht ohne aufwändige Vorverarbeitung 	<ul style="list-style-type: none"> extrahierte Oberflächen als geometrische Repräsentationen: Punkte, Linien, Flächen, Dreiecksnetze, Punktwolken alle tomographischen Verfahren extrahierte Organe darstellen Lage, Größe und Form von Veränderungen feststellen 	<ul style="list-style-type: none"> Volumendaten 3D-Felder von Skalarwerten fast nur CT-Daten Übersichtsdarstellung Exploration des Datensatzes Blick ins Innere mit Kontextinformation
Ziel der Visualisierung	<ul style="list-style-type: none"> Hintergrundwissen zur Bildentstehung 	<ul style="list-style-type: none"> versteht Extraktion der Oberflächen → Genauigkeit kann Qualität der Visualisierung realistisch einschätzen 	<ul style="list-style-type: none"> versteht und beherrscht Definition von Transferfunktionen kennt Hounsfield-Werte bzw. MRT-Eigenschaften
Vorwissen des Betrachters	<ul style="list-style-type: none"> Präferiert Blick auf die Originaldaten (wie beim Röntgenbild) kann Volumen aus 2D-Ansichten rekonstruieren 	<ul style="list-style-type: none"> kennt den nicht dargestellten Kontext will nur Oberfläche von Objekten sehen, nicht deren Inneres wünscht räumliche Darstellung/Interaktion anatomische Modelle für die Lehre 	<ul style="list-style-type: none"> will Organe und Kontext sehen wünscht räumliche Darstellung/ Interaktion durchsichtiger (gläserner) Mensch
Metaphern im Anwendungsgebiet Eigenschaften des Darstellungsmediums	<ul style="list-style-type: none"> auf Film entwickelte Röntgenbilder Film: hoher Grauwert- und Kontrastumfang Monitor: geringer Grauwert- und Kontrastumfang → macht Wahl von Window/Level nötig 	<ul style="list-style-type: none"> Monitor, Grafikkarte: schnelles Rendering mit Beleuchtungseffekten → ermöglicht räumliche Darstellung 	<ul style="list-style-type: none"> Monitor, Grafikkarte: Rendering langsamer als bei indirekter Volumen Visualisierung Skalarwerte in Farb- und Opazitätswerte übersetzen

Tabelle A.1.: Übersicht über die Visualisierungs Methoden

Eigenschaften der Datensätze			
Gliederungspunkt	Monomodale Visualisierung		Vergleich 3D- und 2D- Visualisierung
Name des Ausgangsdatsatz	CT MRT PET		CT PET Thorax
Modalitätenreihenfolge	MRT	PET	CT
Dargestellte Körperregion	Schädel		Thorax
Inhalt der Filterpipeline	Median	Med., Überabtastung	Median
Size = Anzahl der Voxel	256 x 256 x 21	512 x 512 x 42	369 x 267 x 131
Spacing = Größe eines Voxels	0,78 x 0,78 x 5,62	0,39 x 0,39 x 2,81	0,98 x 0,98 x 2,4
Art der Visualisierung	gradient MRT	gradient PET	gradient PET
Größe des Histogramms	159 x 139	249 x 40	243 x 160
Radius		200	234 x 55
Anzahl der Klassen	37	38	1130
Dauer der einzelnen Arbeitsschritte in Sekunden			
Zeit für die Generierung des Histogramms	59,73 Sek.	70,91 Sek.	67,89 Sek.
Zeit für die Generierung des Histogramms inkl. Registrierungszeit	0,06 Sek.	0,02 Sek.	0,08 Sek.
Zeit für die Klassifizierung des Histogramms	1,02 Sek.	0,08 Sek.	0,14 Sek.
Zeit pro Klassenauswahl (picken)	8,63 Sek.	3,86 Sek.	8,40 Sek.
Zeit für Neuklassifizierung nach Änderung des Radius		0,81 Sek.	1,05 Sek.
Zeit für das Rendering			0,95 Sek.
			200
			43
			6,55 Sek.
			9,86 Sek.
			0,09 Sek.
			0,33 Sek.
			2,75 Sek.
			0,20 Sek.

Eigenschaften der Datensätze			
Gliederungspunkt	Vorteile der Überabtastung		Vergleich multiintensitäts- und gradientenbasiert
Name des Ausgangsdatsatz	CT MRT PET		CT PET Body
Modalitätenreihenfolge	MRT	PET	CT, PET
Dargestellte Körperregion	Schädel		Ganzkörper
Inhalt der Filterpipeline	Median	Med., Überabtastung	Median, Gauß
Size = Anzahl der Voxel	256 x 256 x 21	512 x 512 x 42	97 x 97 x 351
Spacing = Größe eines Voxels	0,78 x 0,78 x 5,62	0,39 x 0,39 x 2,81	5,15 x 5,15 x 2,42
Art der Visualisierung	multiintensitätsbasiert		gradient CT
Größe des Histogramms	249 x 159		253 x 39
Radius	200		1155
Anzahl der Klassen	40	42	95
Dauer der einzelnen Arbeitsschritte in Sekunden			
Zeit für die Generierung des Histogramms	9,17 Sek.	80,81 Sek.	17,17 Sek.
Zeit für die Generierung des Histogramms inkl. Registrierungszeit	16,92 Sek.	87,86 Sek.	13,28 Sek.
Zeit für die Klassifizierung des Histogramms	0,16 Sek.	0,19 Sek.	0,28 Sek.
Zeit pro Klassenauswahl (picken)	0,06 Sek.	1,24 Sek.	0,14 Sek.
Zeit für Neuklassifizierung nach Änderung des Radius	1,90 Sek.	9,92 Sek.	1,63 Sek.
Zeit für das Rendering	0,20 Sek.	0,78 Sek.	0,52 Sek.
			0,53 Sek.
			78
			14,27 Sek.
			0,02 Sek.
			0,06 Sek.
			0,98 Sek.

Tabelle A.2.: Messdaten zu den visualisierten Datensätzen.

Programmcode

Listing A.1: In der Funktion `MIP2DHistGeneratorPanel::ComputeOverlappingVolumes()` wird die überschneidende Region der beiden Volumendatensätze berechnet. Diese Funktion ist hier ausschnittsweise gezeigt.

```

1 // Definition des itk::Image Typs mit drei Dimensionen
typedef itk::Image<unsigned short, 3> ImageType;

// die Ausdehnung der Volumen holen
// Bildkoordinate der linken, unteren, vorderen Ecke
6 // der zu Verarbeitenden Region der Volumen bestimmen
// ist nach dem Resampling immer [0,0,0]
// m_pRotatedVolume ist das Transformierte Volumen
// m_pSecondImage ist das zweite eingeladene Volumen
ImageType::IndexType firstMinimumCoords =
11   m_pRotatedVolume->GetRequestedRegion().GetIndex();
ImageType::IndexType secondMinimumCoords =
    m_pSecondImage->GetRequestedRegion().GetIndex();

// Anzahl der Voxel bis zur rechten, oberen, hinteren Ecke
16 // der zu Verarbeitenden Region der Volumen bestimmen
ImageType::SizeType firstMaximumCoords =
    m_pRotatedVolume->GetRequestedRegion().GetSize();
ImageType::SizeType secondMaximumCoords =
    m_pSecondImage->GetRequestedRegion().GetSize();
21

// den Unterschied in der Ausdehnung der beiden Volumen
// für jede Dimension bestimmen
int difference[3];
difference[0] = firstMaximumCoords[0] - secondMaximumCoords[0];
26 difference[1] = firstMaximumCoords[1] - secondMaximumCoords[1];
difference[2] = firstMaximumCoords[2] - secondMaximumCoords[2];

// die zu verarbeitenden Regionen der beiden Volumen beschneiden
for(int i = 0; i < 3; i++) // für alle drei Raumdimensionen
31 {
    // wenn das erste Volumen in x-Dimension größer ist als das zweite
    if(difference[i] >= 0)
    {
        // wenn der Unterschied eine grade Anzahl von Pixeln ist
        // kann die Region vom ersten Volumen direkt angepasst werden
36         if((difference[i] % 2) == 0)
        {
            firstMinimumCoords[i] += difference[i] / 2;
            firstMaximumCoords[i] -= difference[i] / 2;
41         }
        // bei ungeradem Unterschied wird die minimale Schicht
        // (links, vorn, unten) weggelassen

```

```

else
{
46     firstMinimumCoords[i] += (difference[i] / 2)+1;
        firstMaximumCoords[i] -= difference[i] / 2;
}
}
// wie oben nur für das zweite Volumen
51 else
{
    if(difference[i] % 2 == 0)
    {
        secondMinimumCoords[i] -= difference[i] / 2;
56         secondMaximumCoords[i] += difference[i] / 2;
    }
    else
    {
        secondMinimumCoords[i] -= (difference[i] / 2)-1;
61         secondMaximumCoords[i] += difference[i] / 2;
    }
}
}

```

Listing A.2: Im itkListSampleToSTFHistogramGenerator werden das Histogramm generiert und Schwerpunkt und Varianz berechnet. Daneben wird die `std::multimap` für den effizienten Zugriff auf die Voxel eines Bins erzeugt.

```

1 // Das Histogramm mit der angepassten Größe und dem
// oberen und unteren Grenzen der abgetragenen Werte erzeugen
m_Histogram->Initialize(m_Sizes, h_lower, h_upper) ;
// einen Iterator über das Histogramm generieren
HistogramType::Iterator ithisto = m_Histogram->Begin();
6 HistogramType::Iterator end = m_Histogram->End();
// Alle Bins auf 0 setzen
while(ithisto != end)
{
    ithisto.SetFrequency(0);
11 ++ithisto;
}
// einen Iterator über die Eingabedaten generieren
typename TListSample::ConstIterator imageIter = m_List->Begin() ;
typename TListSample::ConstIterator imageLast = m_List->End() ;
16
typename HistogramType::IndexType index ;
typename TListSample::MeasurementVectorType lvector ;
typename HistogramType::MeasurementVectorType hvector ;
unsigned int i;
21 unsigned int imageIndexX = 0;
unsigned int imageIndexY = 0;
unsigned int imageIndexZ = 0;

```

```
itk::Index<2> indexBaryAndVary;
double worldCoordX = 0;
26 double worldCoordY = 0;
double worldCoordZ = 0;
unsigned int mem, div;

// die Felder für Schwerpunkt (barycenter) und
31 // Varianz initialisieren
double init[3] = {0.0, 0.0, 0.0};
ArrayType baryCoordinates = ArrayType(init);
m_Barycenters->SetRegions(m_Sizes);
m_Barycenters->Allocate();
36 m_Barycenters->FillBuffer(baryCoordinates);
double varianceCoordinate = 0.0;
m_Variances->SetRegions(m_Sizes);
m_Variances->Allocate();
m_Variances->FillBuffer(varianceCoordinate);
41 // die std::multimap für die Speicherung der Voxelkoordinaten
m_VolumeIndexMultiMap.clear();

// das Histogramm auffüllen und gleichzeitig Schwerpunkt und
// Varianz der Bins berechnen
46 // dazu über die Eingabedaten iterieren
while (imageIter != imageLast)
{
    // aus dem Identifier des Iterators über die Liste der
    // Eingabedaten die Bildkoordinaten berechnen
51 imageIndexX = imageIter.GetInstanceIdentifier() % m_ImageSizeX;
mem = imageIter.GetInstanceIdentifier() - imageIndexX;
div = mem / m_ImageSizeX;
imageIndexY = div % m_ImageSizeY;
mem = div - imageIndexY;
56 div = mem / m_ImageSizeY;
imageIndexZ = div % m_ImageSizeZ;
// die Bildkoordinaten in Weltkoordinaten umrechnen
worldCoordX = imageIndexX * m_ImageSpacingX;
worldCoordY = imageIndexY * m_ImageSpacingY;
61 worldCoordZ = imageIndexZ * m_ImageSpacingZ;

// die Messwerte des Bildes in die des Histogramms umwandeln
lvector = imageIter.GetMeasurementVector() ;
for ( i = 0 ; i < HistogramType::MeasurementVectorSize ; i++)
66 {
    hvector[i] = (THistogramMeasurement) lvector[i] ;
}
// die Position des Bins in die der Messwert sortiert werden muss holen
// wird ITK intern durch binäre Suche ermittelt
71 m_Histogram->GetIndex(hvector,index);

// liegt der Messwert ausserhalb der Histogrammgrenzen?
```



```

if (!m_Histogram->IsIndexOutOfBounds(index))
{
76 // Die k-Nachbarschaft aus ~\cite{Spatialized_TF} bei der
// Histogrammgenerierung anwenden
for (int k = -2; k <= 2; k++)
{
81 for (int l = -2; l <= 2; l++)
{
indexBaryAndVary[0] = index[0];
indexBaryAndVary[1] = index[1];
indexBaryAndVary[0] = indexBaryAndVary[0] + k;
indexBaryAndVary[1] = indexBaryAndVary[1] + l;

86 // liegt der Nachbarpunkt auch noch im Histogramm?
if (!m_Histogram->IsIndexOutOfBounds(indexBaryAndVary) &&
indexBaryAndVary[0] >= 0 && indexBaryAndVary[0] < m_Sizes[0] &&
indexBaryAndVary[1] >= 0 && indexBaryAndVary[1] < m_Sizes[1])
91 {
// das Bin im Histogramm um eins erhöhen
m_Histogram->IncreaseFrequency(indexBaryAndVary, 1);
// die Koordinaten des Voxels zum Schwerpunkt hinzurechnen
baryCoordinates = m_Barycenters->GetPixel(indexBaryAndVary);
96 baryCoordinates[0] += worldCoordX;
baryCoordinates[1] += worldCoordY;
baryCoordinates[2] += worldCoordZ;
m_Barycenters->SetPixel(indexBaryAndVary, baryCoordinates);
// die Koordinaten des Voxels zur Varianz hinzurechnen
101 varianceCoordinate = m_Variances->GetPixel(indexBaryAndVary);
varianceCoordinate += (worldCoordX * worldCoordX) +
(worldCoordY * worldCoordY) + (worldCoordZ * worldCoordZ);
m_Variances->SetPixel(indexBaryAndVary, varianceCoordinate);

106 // den Voxel nur für das jeweilige Bin speichern, nicht auch für seine
// k Nachbarn
if (k == 0 && l == 0)
{
unsigned int id = indexBaryAndVary[1] * m_Sizes[0] +
111 indexBaryAndVary[0];
unsigned int volumeid = imageIter.GetInstanceIdentifier();
// das Wertepaar in die multimap einbinden
m_VolumeIndexMultiMap.insert(
std::pair<unsigned int const, unsigned int const>(id, volumeid));
116 }
}
}
}
}
121 ++imageIter ;
}
// wieder an den Anfang des Histogramms gehen

```

```
ithisto = m_Histogram->Begin();
end = m_Histogram->End();
126
// über das Histogramm iterieren, um die Schwerpunkte zu normieren
// und die Varianz zu berechnen
while(ithisto != end)
{
131 // aus dem Identifier des Iterators die Position im Histogramm berechnen
indexBaryAndVary[0] = ithisto.GetInstanceIdentifier() % m_Sizes[0];
mem = ithisto.GetInstanceIdentifier() - indexBaryAndVary[0];
div = mem / m_Sizes[0];
indexBaryAndVary[1] = div % m_Sizes[1];
136 // die Anzahl der Voxel im Bin für die Normierung holen
double voxelAnzahl = ithisto.GetFrequency();
// den Schwerpunkt normieren
baryCoordinates = m_Barycenters->GetPixel(indexBaryAndVary);
baryCoordinates[0] /= voxelAnzahl;
141 baryCoordinates[1] /= voxelAnzahl;
baryCoordinates[2] /= voxelAnzahl;
m_Barycenters->SetPixel(indexBaryAndVary, baryCoordinates);
// die Varianz mit dem Schwerpunkt berechnen und normieren
varianceCoordinate = m_Variances->GetPixel(indexBaryAndVary);
146 varianceCoordinate = (varianceCoordinate / voxelAnzahl) -
(baryCoordinates[0] * baryCoordinates[0]) -
(baryCoordinates[1] * baryCoordinates[1]) -
(baryCoordinates[2] * baryCoordinates[2]);
m_Variances->SetPixel(indexBaryAndVary, varianceCoordinate);
151
++ithisto;
}
```

Verzeichnis verwendeter Abkürzungen

- 1D** - eindimensional
- 2D** - zweidimensional
- 3D** - dreidimensional
- CT** - Computertomographie, Computertomograph
- et al.** - et alii <lat.> (und andere)
- FTF** - Farbtransferfunktion
- IGD** - Institut für Graphische Datenverarbeitung der Fraunhofer Gesellschaft
- ITK** - *Insight Toolkit*; freie Programmbibliothek für die Bildverarbeitung, entwickelt von der Kitware Inc.
- MI** - *Mutual Information*
- MIP** - Medical Imaging Platform; zu erweiternde Software für die medizinische Bildverarbeitung
- MIR** - *Mutual Information Registration*
- MPP** - *Multiple Properties per Point*
- MRT** - Magnetresonanztomographie, Magnetresonanztomograph
- OPP** - *One Property per Point*
- PET** - Positronen Emissions Tomographie, Positronen Emissions Tomograph
- STF** - *Spatialized Transfer Function*
- STFs** - *Spatialized Transfer Functions*
- TF** - Transferfunktion
- TFs** - Transferfunktionen
- TTF** - Transparenz-Transferfunktion
- TTFs** - Transparenz-Transferfunktionen
- VTK** - *Visualization Toolkit*; freie Programmbibliothek für die Visualisierung, entwickelt von der Kitware Inc.

Abbildungsverzeichnis

2.1. Spiral-Computertomograph.	5
2.2. Rückprojektion und gefilterte Rückprojektion.	6
2.3. Hounsfield Einheiten.	6
2.4. Einteilung der verwendeten CT-Datensätze.	8
2.5. Funktionsweise eines PET-Scanners.	11
2.6. Einteilung der verwendeten PET-Datensätze.	12
2.7. Schichtlage im MRT.	14
2.8. T1 und T2 gewichtete MRT-Aufnahmen.	14
2.9. Einteilung der verwendeten MRT-Datensätze.	16
2.10. Rekonstruktionsfilter.	20
2.11. Partialvolumeneffekt.	22
2.12. Transformationsarten der Registrierung.	25
3.1. Weichteil- und Knochenfenster im CT.	28
3.2. Multimodale Visualisierung durch Schichtbilder.	29
3.3. Allgemeine Volumenvisualisierungspipeline.	31
3.4. Visualisierungspipeline für indirekte Volumenvisualisierung.	31
3.5. Multimodale indirekte Volumenvisualisierung.	33
3.6. Visualisierungspipeline für direkte Volumenvisualisierung.	35
3.7. <i>Property Fusion</i> Pipeline.	37
3.8. <i>Property & Gradient Fusion</i> Pipeline.	38
3.9. <i>Material Fusion</i> Pipeline.	38
3.10. <i>Shading Fusion</i> Pipeline.	39
3.11. <i>Color</i> Pipeline.	40
3.12. <i>Ray Casting</i>	42
3.13. <i>3D-Texture Mapping</i>	43
3.14. <i>Shear Warp</i>	45
4.1. Zweidimensionale Transparenz-Transferfunktion.	50
4.2. Verwendetes Kantenmodell.	52
4.3. Gradientenrichtung entlang einer Kante.	52
4.4. Verlauf der Messwerte und Ableitungen senkrecht zu einer Kante.	53
4.5. Dreidimensionaler Plot der Messwerte senkrecht zu einer Kante.	54
4.6. Abtastung der Datenwerte und resultierende Bogenstruktur im Histogramm.	55

4.7.	3D-Sobeloperatoren.	57
4.8.	Generierung des 2D-Histogramms.	58
4.9.	Grauwertdarstellung eines 2D-Histogramms.	58
4.10.	2D-Plot der ersten Ableitung in Abhängigkeit von der Bildintensität für zwei Gewebe.	59
4.11.	Auswirkung der Überabtastung und der k-Nachbarschaft auf das 2D-Histogramm.	63
4.12.	Darstellung eines klassifizierten Histogramms.	65
5.1.	Darstellung der ausgewählten Visualisierungsmöglichkeiten.	72
5.2.	Vergleich der Histogramme des gradientenbasierten und des multiintensitätsbasierten Ansatzes.	73
5.3.	Vergleich der klassifizierten Histogramme des gradientenbasierten und des multiintensitätsbasierten Ansatzes.	74
5.4.	Benutzeroberfläche der implementierten Visualisierungskomponente.	78
5.5.	Darstellung des Programmablaufs unter Verwendung von zwei Transferfunktionen (gradientenbasierter Ansatz).	80
5.6.	Darstellung des Programmablaufs unter Verwendung einer Transferfunktion (multiintensitätsbasierter Ansatz).	81
5.7.	Eingeschränkte Separierbarkeit mehrdimensionaler Transferfunktionen.	82
5.8.	Auswirkung der Überabtastung auf das klassifizierte Histogramm.	87
5.9.	Auswirkung der Anpassung der Histogrammgröße.	89
5.10.	Auswirkung der k-Nachbarschaft auf das 2D-Histogramm.	90
5.11.	Klassifikation eines 2D-Histogramms mit unterschiedlichen Radien.	95
5.12.	Auswirkung der Transformation auf die Bildinformation.	98
5.13.	Beispielhafte Berechnung der sich überschneidenden Bereiche.	99
6.1.	Monomodale gradientenbasierte Visualisierungen eines MRT-Datensatzes.	102
6.2.	Zusammenhang zwischen klassifiziertem Intensitäts-/ Gradientenstärke-Histogramm und der MRT-Visualisierung.	102
6.3.	Gradientenbasierte Visualisierungen eines PET-Datensatzes.	103
6.4.	Schichtbilddarstellung mit Schachbrettmuster der CT- und PET-Thoraxdatensätze.	104
6.5.	Coronale Ansicht auf die 3D-Visualisierung der CT- und PET-Thoraxdatensätze.	105
6.6.	Ansicht von hinten links auf die 3D-Visualisierung der CT- und PET-Thoraxdatensätze.	106
6.7.	Drei Ansichten auf die 3D-Visualisierung der CT- und PET-Thoraxdatensätze.	107
6.8.	Schichtbilder „CT MRT Head 3“.	108
6.9.	Visualisierung der Ventrikel aus „CT MRT Head 3“.	109
6.10.	Visualisierung der Ventrikel und Knochenstruktur aus „CT MRT Head 3“.	109
6.11.	Visualisierung des Gehirns mit Ventrikeln und Knochenstruktur aus „CT MRT Head 3“.	110
6.12.	Zusammenhang zwischen klassifiziertem Histogramm und Visualisierung von „CT MRT Head 3“.	110

6.13. Vergleich des Effekts der Überabtastung am schwach aktivierten Teil des Gehirns.	112
6.14. Vergleich des Effekts der Überabtastung am stark aktivierten Teil des Gehirns.	113
6.15. Vergleich der klassifizierten Histogramme bei Überabtastung.	114
6.16. Vergleich der klassifizierten Histogramme von MRT/PET und CT/PET im multiintensitätsbasierten Ansatz.	114
6.17. Schichtbilder „CT PET Body“.	115
6.18. Gradientenbasierte Visualisierung des Ganzkörperdatensatzes „CT PET Body“ - coronal.	117
6.19. Gradientenbasierte Visualisierung des Ganzkörperdatensatzes „CT PET Body“ - sagittal.	117
6.20. Multiintensitätsbasierte Visualisierung des Ganzkörperdatensatzes „CT PET Body“ - coronal.	118

Literaturverzeichnis

- [Ang00] Edward Angel. *Interactive Computer Graphics - A Top-Down Approach with OpenGLTM*. Addison-Wesley Publishing Company, 2. edition, 2000.
- [CKLG98] Silvia Castro, Andreas König, Helwig Löffelmann, and Meister Eduard Gröller. Transfer Function Specification for the Visualization of Medical Data. Technical Report TR-186-2-98-12, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria, March 1998. human contact: technical-report@cg.tuwien.ac.at.
- [CTA] Vorlesung Radiologische bildgebende Verfahren WS 2004/05 - Bildgebende Verfahren, Röntgen-Computertomografie.
Link: http://www.med.uni-magdeburg.de/fme/khno/audio/studium/cv/VL_RoeBildgebung.pdf.
zuletzt besucht am 02.09.2005.
- [CTH] *Computed Tomography - Its History and Technology*. Siemens AG, Medical Solutions, Henkestr. 127, D-91052 Erlangen, Germany, Telefon: ++49 9131 84-0 www.SiemensMedical.com.
Link: http://www.medical.siemens.com/siemens/en_INT/gg_ct_FBAs/files/brochures/CT_History_and_Technology.pdf
zuletzt besucht am 23.06.2005.
- [FBP] Rückprojektion und gefilterte Rückprojektion.
Link: http://people.na.infn.it/~cavalier/Download/SICSI_ASC/RappresentazioneImmagini/2_Immagini%20acquisizione%20ed%20elaborazione.htm.
zuletzt besucht am 23.06.2005.
- [FPT02] Maria Ferré, Anna Puig, and Dani Tost. Rendering techniques for multi-modal data. Technical Report Report LSI-64-R, 2002.
Link: <http://wai.maia.ub.es/basilic/Publications/2002/FPT02b/r02-13.pdf>
zuletzt besucht am: 23.06.2005.

- [FPT04] Maria Ferre, Anna Puig, and Dani Tost. A framework for fusion methods and rendering techniques of multimodal volume data. *Journal of Visualization and Computer Animation*, 15(2):63–77, 2004.
- [FvDFH90] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics. Principles and Practice*. Addison-Wesley Publishing Company, 2. edition, 1990.
- [GCT] Computertomographie aus: Die Geschichte der Radiologie.
Link: <http://www.radiologie.de>
bzw. direkt: <http://www.radiologienetz.de/article/archive/82/>.
zuletzt besucht am 23.06.2005.
- [GK96] Allen Van Gelder and Kwansik Kim. Direct Volume Rendering with Shading via Three-Dimensional Textures. In *VVS '96: Proceedings of the 1996 symposium on Volume Visualization*, pages 23–ff., Piscataway, NJ, USA, 1996. IEEE Press.
- [GMR] Magnetresonanztomographie aus: Die Geschichte der Radiologie.
Link: <http://www.radiologie.de>
bzw. direkt: <http://www.radiologienetz.de/article/archive/83/>.
zuletzt besucht am 23.06.2005.
- [HB] K.H. Höhne and R. Bernstein. Shading 3d-images from CT using gray level gradients.
- [HK03] Alexander Hendrix and Jacqueline Krempe. *Magnete, Spins und Resonanzen - Eine Einführung in die Grundlagen der Magnetresonanztomographie*. Siemens AG, Medical Solutions, Magnet Resonanz Henkestr. 127, D-91052 Erlangen, Deutschland Telefon: ++49 9131 84-0 www.SiemensMedical.com DokuMR@med.siemens.de, 2003.
- [HRH03] Dieter Hönigmann, Johannes Ruisz, and Christoph Haider. Adaptive Design of a Global Opacity Transfer Function for Direct Volume Rendering of Ultrasound Data. *IEEE Visualization 2003*, pages 489–496, October 2003.
- [HST⁺03] F. Vega Higuera, N. Sauber, B. Tomandl, C. Nimsy, G. Greiner, and P. Hastreiter. Enhanced 3D-Visualization of Intracranial Aneurysms Involving the Skull Base. *miccai*, 2003.
- [HST⁺04] Fernando Vega Higuera, Natascha Sauber, Bernd Tomandl, Christopher Nimsy, Günther Greiner, and Peter Hastreiter. Automatic Adjustment of Bidimensional Transfer Functions for Direct Volume Visualization of Intracranial Aneurysms. *Proc. SPIE*, 5367:275–284, May 2004.
- [Kau] T. Kaulisch. Skript - Einführung in die Magnetresonanztomographie (MRT, MRI, NMRI).
- [KD98] Gordon Kindlmann and James W. Durkin. Semi-Automatic Generation of Transfer Functions for Direct Volume Rendering. *IEEE*, 1998.

- [Kin99] Gordon Lothar Kindlman. Semi-Automatic Generation of Transfer Functions for Direct Volume Rendering. Master's thesis, Cornell University, Januar 1999.
- [KKH01] Joe Kniss, Gordon Kindlmann, and Charles Hansen. Multidimensional Transfer Functions for Interactive Volume Rendering. *IEEE Transactions on Visualization and Computer Graphics*, 8(3):270–285, July-September 2001.
- [KPI⁺03] Joe Kniss, Simon Premože, Milan Ikits, Aaron Lefohn, Charles Hansen, and Emil Praun. Gaussian Transfer Functions for Multi-Field Volume Visualization. *IEEE Visualization 2003*, pages 497–504, October 2003.
- [Lau99] Theodor Laubenberger. *Technik der Medizinischen Radiologie*. 7., überarb. Aufl. Dt. Ärzte Verlag, 1999.
- [LC87] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, pages 163–169, New York, NY, USA, 1987. ACM Press.
- [Lev88] Marc Levoy. Display of Surfaces From Volume Data. *IEEE Computer Graphics and Applications*, 8(3):29–37, 1988.
- [Lev90] Marc Levoy. Efficient Ray Tracing of Volume Data. *ACM Trans. Graph.*, 9(3):245–261, 1990.
- [LFB98] David H. Laidlaw, Kurt W. Fleischer, and Alan H. Barr. Partial-Volume Bayesian Classification of Material Mixtures in MR Volume Data using Voxel Histograms. *IEEE Transactions on Medical Imaging*, 17(1):74–86, Feb. 1998.
- [LL94] Philippe Lacroute and Marc Levoy. Fast Volume Rendering Using a Shear-Warp Factorization of the Viewing Transformation. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 451–458, New York, NY, USA, 1994. ACM Press.
- [LLVT03] Thomas Lewiner, Hélio Lopes, Antônio Wilson Vieira, and Geovan Tavares. Efficient implementation of Marching Cubes' cases with topological guarantees. *Journal of Graphics Tools*, 8:1–15, 2003.
- [LLY05] Claes Lundstöm, Patric Ljung, and Anders Ynnerman. Extending and Simplifying Transfer Function Design in Medical Volume Rendering Using Local Histograms. *EUROGRAPHICS*, 2005.
- [LOPR97] Th. Lehmann, W. Oberschelp, E. Pelikan, and R. Repges. *Bildverarbeitung für die Medizin*. Springer Verlag, 1997.
- [MC] Medical Imaging - Isosurface Rendering.
Link: <http://www.wissrech.iam.uni-bonn.de/research/projects/gerstner/medvis/medvis.html>
bzw. direkt: <http://www.wissrech.iam.uni-bonn.de/research/>

- projects/gerstner/medvis/tooth.jpg.
zuletzt besucht am 30.06.2005.
- [MCV⁺97] Frederik Maes, André Collignon, Dirk Vandermeulen, Guy Marchal, and Paul Suetens. Multimodality Image Registration by Maximization of Mutual Information. *IEEE Transactions on Medical Imaging*, 16(2):187–198, April 1997.
- [MED] MedlinePlus: Drug Information: Magnetig Resonance Imaging Contrast Agents (Diagnostic).
Link: <http://www.nlm.nih.gov/medlineplus/druginfo/uspdi/202770.html>.
zuletzt besucht am 14.10.2005.
- [MHS99] M. Meissner, U. Hoffmann, and W. Straber. Enabling classification and shading for 3d texture mapping based volume rendering using opengl and extensions. In *Proceedings of IEEE Visualization 1999*, pages 207–214, 1999.
- [MR1a] History of MRI aus der Magnetic Resonance Imaging Database.
Link: <http://www.mr-tip.com>
bzw. direkt: <http://www.mr-tip.com/serv1.php?type=db&search=M&set=20>.
zuletzt besucht am 23.06.2005.
- [MR1b] Contrast Agents from Magnetic Resonance - Technology Information Portal.
Link: <http://www.mr-tip.com>
bzw. direkt: <http://www.mr-tip.com/serv1.php?type=coa>.
zuletzt besucht am 14.10.2005.
- [Mul] Sgi homepage datentypen - multi maps.
Link: http://www.sgi.com/tech/stl/table_of_contents.html.
zuletzt besucht am 11.10.2005.
- [MV96] J. Maintz and M. Viergever. An Overview of Medical Image Registration Methods. *Imaging Science Department, Imaging Center Utrecht*, 1996.
- [MV98] J. Maintz and M. Viergever. A Survey of Medical Image Registration. *Medical Image Analysis*, 2(1):1–36, 1998.
- [Pen99] Yun Peng. Multimodale Visualisierung von PET und CT Volumendaten zur Zielvolumendefinition in der Radioonkologie. Master's thesis, Technische Universität Darmstadt, August 1999.
- [PET] A History of Positron Imaging.
Link: <http://www.mit.edu/~glb/alb.html>.
zuletzt besucht am 23.06.2005.
- [Pho75] B. T. Phong. Illumination for Computer Generated Scenes. *Communications of the ACM*, 18(6):311–317, June 1975.

- [PLB⁺01] Hanspeter Pfister, Bill Lorensen, Chandrajit Bajaj, Gordon Kindlmann, Will Schroeder, Lisa Sobierajski Avila, Ken Martin, Raghu Machiraju, and Jinho Lee. The Transfer Function Bake-Off. *IEEE Comput. Graph. Appl.*, 21(3):16–22, 2001.
- [PM04] Simeon Potts and Torsten Möller. Transfer functions on a logarithmic scale for volume rendering. In *GI '04: Proceedings of the 2004 conference on Graphics interface*, pages 57–63, School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada, 2004. Canadian Human-Computer Communications Society.
- [PMD] Pet in more Detail.
Link: <http://www.petnm.unimelb.edu.au/pet/detail/index.html>.
zuletzt besucht am 23.06.2005.
- [PMV03] Josien P. W. Pluim, J.B. Antoine Maintz, and Max A. Viergever. Mutual-Information-Based Registration of Medical Images: A Survey. *IEEE Transactions on Medical Imaging*, 22(8):986–1004, August 2003.
- [Poh] Regina Pohle. Skript Nuklearmedizin - erarbeitet nach der Vorlesung von Prof. Otto und Dr. Coenen.
Link: http://isgnw.cs.uni-magdeburg.de/~regina/SKRIPT_NUKLEAR.HTML.
zuletzt besucht am 07.09.2005.
- [PTF02] Anna Puig, Dani Tost, and M. Ferré. Design of a Multimodal Rendering System. In *Proc. 7th International Fall Workshop Vision, Modeling and Visualization 2002*, pages 488–496, 2002.
- [RBS05] Stefan Roettger, Michael Bauer, and Marc Stamminger. Spatialized Transfer Functions. *EUROGRAPHICS - IEEE VGTC Symposium on Visualization*, 2005.
- [RS01] Christof Rezk-Salama. *Volume Rendering Techniques for General Purpose Graphics Hardware*. PhD thesis, Universität Erlangen Nürnberg, 2001.
- [RSHSG00] Christof Rezk-Salama, Peter Hastreiter, Jörg Scherer, and Günther Greiner. Automatic Adjustment of Transfer Functions for 3D Volume Visualization. In *Proceedings Vision, Modeling and Visualization (VMV) 2000*, pages 357–364, November 2000.
- [SHE] Homepage zum shell sort sortieralgorithmus.
Link: <http://www.iti.fh-flensburg.de/lang/algorithmen/sortieren/shell/shell.htm>.
zuletzt besucht am 20.10.2005.
- [She59] D. L. Shell. A high-speed sorting procedure. *Communications of the ACM*, 2(7):30–32, 1959.
- [SM00] Heidrun Schumann and Wolfgang Müller. *Visualisierung - Grundlagen und allgemeine Methoden*. Springer Verlag, 2000.

- [SML02] Will Schroeder, Ken Martin, and Bill Lorensen. *The Visualization Toolkit - An Object-Oriented Approach To 3D Graphics*. Kitware, Inc., 3. edition, 2002.
- [ST90] Peter Shirley and Allan Tuchman. A polygonal Approximation to Direct Scalar Volume Rendering. In *VVS '90: Proceedings of the 1990 workshop on Volume visualization*, pages 63–70, New York, NY, USA, 1990. ACM Press.
- [Stö04] Diana Stölzel. Entwurf gradientenabhängiger 2D-Transferfunktionen für die medizinische Volumenvisualisierung. Master's thesis, Otto-von-Guericke Universität Magdeburg, September 2004.
- [SZL92] William J. Schroeder, Jonathan A. Zarge, and William E. Lorensen. Decimation of Triangle Meshes. *SIGGRAPH Comput. Graph.*, 26(2):65–70, 1992.
- [Tap04] Andreas Tappenbeck. Entwurf distanzabhängiger 2D-Transferfunktionen für die medizinische Volumenvisualisierung. Master's thesis, Otto-von-Guericke Universität Magdeburg, September 2004.
- [TLM03] Fan-Yin Tzeng, Eric B. Lum, and Kwan-Liu Ma. A Novel Interface for Higher-Dimensional classification of Volume Data. In *Proceedings of IEEE Visualization 2003*, pages 505–512, October 2003.
- [Wes90] Lee Westover. Footprint Evaluation for Volume Rendering. *SIGGRAPH Comput. Graph.*, 24(4):367–376, 1990.

Index

- Ableitung, 52–57
- Abtastung, 19–21, 44, 53, 55, 98
- Adaptation, 91–92
- Anforderungen
 - an die Datensätze, 4
 - an die Transferfunktionen, 48
 - an die Visualisierung, 26
- Artefakte, 9, 13, 17
- Ausrichtung, 14, 46

- bildgebende Verfahren, 3–18, 22
- Bildrekonstruktion, 5, 19–21
- Bin, 54, 58, 61, 63, 88–93

- Computertomographie, 3–9, 27
- CT, *siehe* Computertomographie

- Datenfusion, *siehe* Fusion
- Decision Function, *siehe* Entscheidungsfunktion
- Diskretisierung, 19, 22, 42
- Distanznorm, 92

- Entscheidungsfunktion, 29
- Entwurf, 70–73

- Filter
 - kernel, 56
 - maske, 56–57, 85, 86
 - Bandpass-, 19, 52–53
 - Dreieck-, 19–20
 - Gauß-, 19, 44, 51, 85, 116
 - Median-, 85
 - Puls-, 19
 - Sobel-, 56

- Filtering, 31, 34
- Fixed Image, 23, 98
- Floating Image, 23, 97
- Fusion
 - Art der, 35
 - Color-, 39–41, 47, 71, 116
 - Material-, 38
 - Property & Gradient-, 37
 - Property-, 36–37, 41, 47, 70
 - Shading-, 39
 - Zeitpunkt der, 36, 41, 70
- Fusionspipeline, *siehe* Fusion

- Generierung, 57–58, 88–91
- gradientenbasiert, 50–60, 80, 97, 101–104, 115–118
- Gradientenstärke, -information (Gradient Magnitude), 34, 49, 52–57, 87

- Histogramm
 - 1D-, 68
 - 2D-, 57–60, 62–64, 88–94, 110
 - als Hilfsmittel, 68
 - Intensitäts-/ Gradientenstärke-, 54–55, 60–62, 88
 - Intensitäts-/ Intensität-, 74, 79, 88, 114
- Hounsfield Einheit, Unit (HU), 5–6, 9

- Implementierung, 70–100
- Intensität, 54–55
- Isooberfläche, 31–32
- ITK, 27, 76–77, 84–99

- Kantenmodell (Kantendetektion), 51
- Klassifikation, 63–64, 92–94, 119
- Konzept, 71–73

- Magnetresonanztomographie, 3, 13–18
- Mapping, 31, 34
- Marching Cube Algorithmus, 31
- Modalität, *siehe* bildgebende Verfahren
- monomodal, 25, 34, 102–104
- MRT, *siehe* Magnetresonanztomographie
- multiintensitätsbasiert, 73, 79, 81, 97, 107–110, 115–118
- multimodal, 1, 22, 28, 33, 35, 66, 119
- Multiple Property per Point (MPP), 35, 39
- multivolumen, 1, 23
- Mutual Information Registration, 23

- Nachbarschaft
 - 8-er, 85
 - k, 62, 88
- Norm, 63, 92
- Normierung, 56, 91
- Numerische
 - Ausgaben, 67
 - Eingaben, 69

- Oberflächenextraktion, 31–32
- One Property per Point (OPP), 35, 37
- Opacity, *siehe* Transparenz
- Ortsinformation, 61–62, 74, 88–93, 118, 119
- Oversampling, *siehe* Überabtastung

- Partialvolumeneffekt, 21–22, 86
- PET, *siehe* Positronen Emissions Tomographie
- Pick-Funktionalität, 69
- Pipeline, *siehe* Visualisierungspipeline
- Positronen Emissions Tomographie, 3, 10–13

- Programmierwerkzeug, 76
 - ITK, 27, 76–77, 84–93, 99
 - VTK, 27, 77, 82, 93, 94, 96, 120
 - wxWidgets, 27, 76–77

- Quantisierung, 19–21

- Radius (r), 63, 66, 69

- Radius(r), 77, 95
- Ray Casting, 42–43, 46–47, 100, 111
- Registrierung, 3, 11, 22–25, 36
- Rendering, 34, 41–47, 100
 - bildbasiert, 41–43
 - front to back, 41–43
- Rotation, *siehe* Transformation

- Schichtbild, 5, 19, 27–30, 68, 78, 104, 108, 115
- Schwerpunkt, 61, 90–91
- Shading, 31, 39
- sinc-Funktion, 19–20
- Skalarwert, 34, 49, 50, 116
- Spatialized Transfer Functions, 60–66, 71, 73–75, 94–96
- Standardisierung, 9, 13, 18
- Stufenfunktion, 51–53

- Texture Mapping, 43–44, 121
- Tranfer Function, *siehe* Transferfunktion
- Transferfunktion, 48–69
 - eindimensionale, 49
 - Farb-, 93, 96, 120
 - gradientenbasierte, 50–60, 73, 80, 101
 - mehrdimensionale, 50–60, 82
 - multiintensitätsbasierte, 73, 81
 - Spatialized, 60–66, 71, 73–75, 94–96
 - Spezifikation von, 67–69, 94, 121
 - Transparenz- (Opacity), 49, 65, 96, 120
- Transformation, 23–25, 97
- Translation, *siehe* Transformation
- Transparenz, 49, 50, 65, 96, 120
- Tupel, 62–64

- Überabtastung, 62, 86, 111–113

- Varianz, 61, 90–91
- Visualisierungsparameter, 27, 35, 36, 40
- Visualisierungspipeline
 - direkt, 34–40
 - indirekt, 30–31
- Volumendatensatz, 4, 19–21, 46
- Volumenvisualisierung
 - direkt, 34–41, 48
 - indirekt, 30–34

- Zerlegungssatz, 90

Selbstständigkeitserklärung

Hiermit erkläre ich, Matthias Keil (Matrikel-Nummer 160490), dass ich die vorliegende Diplomarbeit selbstständig angefertigt habe. Es wurden nur die in der Arbeit ausdrücklich genannten Quellen und Hilfsmittel benutzt. Wörtlich oder sinngemäß übernommenes Gedankengut habe ich als solches kenntlich gemacht.

Magdeburg, den 4. November 2005