

Masterarbeit

Level-Set-Segmentierung von Rattenhirn-MRTs

Björn Eiben

Prof. Dr. Dietmar Kunz

Dr. Christoph Palm
Prof. Dr. Uwe Pietrzyk

Zusammenfassung

In dieser Arbeit wird die Segmentierung von Gehirngewebe aus magnet-resonanz-tomographischen Kopfaufnahmen von Ratten mittels Level-Set-Methoden vorgestellt. Dieses so genannte Skull-Stripping stellt einen wichtigen Vorverarbeitungsschritt für quantitative, morphometrische Untersuchungen oder aber Visualisierungsaufgaben dar.

Ein kontrastbasierter Segmentierungsalgorithmus wird von einem Pseudo-3D-Ansatz in einen echt-dreidimensionalen Segmentierer überführt. Die durch die Level-Set-Funktion beschriebene Kontur wird mittels einer partiellen Differentialgleichung iterativ deformiert und den Grenzen des zu segmentierenden Objektes angenähert. Die Geschwindigkeitsfunktion, welche lokale Kontraste auf der Konturnormalen auswertet und so die Oberflächenentwicklung bestimmt, wird untersucht und das lokale Signal adaptiert. Hierzu wird eine Glättung des Signals eingeführt, die sowohl in Richtung der Konturnormalen als auch parallel dazu wirkt. Zusätzlich wird eine varianzbasierte Kontrastverstärkung des lokalen Signals entwickelt. Daraus resultieren insbesondere in Bildbereichen mit geringem Signal-zu-Rausch-Verhältnis erheblich robustere und exaktere Segmentierungsergebnisse. Diese Leistungsfähigkeit wird an vorliegenden Rattenhirn-MRTs demonstriert.

Schlüsselwörter: Level-Set-Methoden, Segmentierung, Skull-Stripping, Magnet-Resonanz-Tomographie

Inhaltsverzeichnis

1	Einleitung	1
2	Stand der Forschung	3
2.1	Das Segmentierungsproblem	3
2.1.1	Segmentierungsverfahren	3
2.1.2	Hybride Verfahren	4
2.2	Aktive Konturmodelle	5
2.2.1	Snakes	5
2.2.2	Implizite Konturen und Oberflächen	7
2.3	Skull-Stripping	9
2.3.1	Geschwindigkeitsfunktionen	10
2.3.2	Formbeschränkungen	11
2.3.3	Initialisierungsansätze	12
3	Material und Methoden	15
3.1	Level-Set-Formulierung	15
3.1.1	Werkzeuge	17
3.1.2	Distanzfunktion	19
3.2	Level-Set Methoden	21
3.2.1	Entwicklung durch Advektion	21
3.2.2	Entwicklung durch Krümmung	21
3.2.3	Entwicklung durch Propagation	23
3.2.4	Superposition von Entwicklungstermen	25
3.3	Diskretisierung	25
3.3.1	Upwind-Schema	26
3.3.2	Narrow-Band-Level-Sets	27
3.3.3	Sparse-Field-Level-Sets	27
3.3.4	Allgemeine Level-Set-Parameter	29
3.4	Insight Segmentation and Registration Toolkit (ITK)	30
4	Algorithmenanpassung	31
4.1	Der modellbasierte Level-Set-Segmentierer	31
4.1.1	Pseudo 3D-Ansatz	31
4.1.2	Geschwindigkeitsfunktion	32
4.1.3	Parameter	33
4.2	Modifikation des modellbasierten Level-Set-Segmentierers	33
4.2.1	Portierung in den 3-D Raum	35
4.2.2	Vorverarbeitung	35

4.2.2.1	Interpolation zu isotropen Daten	35
4.2.2.2	Padding	37
4.2.3	Modifikation der Geschwindigkeitsfunktion	37
4.2.4	Signaladaption	38
4.2.4.1	Konturnormalen-parallele Glättung	39
4.2.4.2	Konturnormalen-orthogonale Glättung	39
4.2.5	Parameter	43
5	Ergebnisse	45
5.1	Bildmaterial	45
5.2	Qualitätsbewertung	46
5.3	Auswertung lokaler Kontraste	47
5.3.1	Lokaler und globaler Median	47
5.3.2	Parametereinfluss	48
5.3.2.1	Kontrastabhängiger Zielwert	48
5.3.2.2	Kontrastlängen	50
5.4	Signaladaption	52
5.4.1	Konturnormalen-parallele Glättung	52
5.4.2	Konturnormalen-orthogonale und parallele Glättung	52
5.4.3	Kontrasterhöhung	53
5.5	Verfahren im Vergleich	54
5.5.1	Allgemeine Level-Set-Parameter	55
5.5.1.1	Krümmung	55
5.5.1.2	Konvergenzkriterium	56
5.5.2	Wahl der Startkontur	57
6	Zusammenfassung	61
6.1	Diskussion	62
6.2	Ausblick	64
A	Herleitungen	67
A.1	Krümmung 3D	67
B	Zusatzmaterial	69
B.1	Ergänzungen zum Upwind-Schema	69
C	Symbole	71
	Literaturverzeichnis	73

Kapitel 1

Einleitung

Die *Segmentierung* stellt eines der großen Aufgabengebiete der medizinischen Bildverarbeitung dar. Dabei verfolgt sie das Ziel, Regionen in Bildern zu finden, welche inhaltlich zum gleichen Objekt gehören. Somit beschreibt sie eine pixel- bzw. voxelweise Klassifikation eines Bildes.

Die Segmentierung von Gehirngewebe aus Magnet-Resonanz-Tomographien (MRTs) des Kopfes im Speziellen wird als *Skull-Stripping* bezeichnet. In magnet-resonanz-tomographischen Aufnahmen stehen dreidimensionale in vivo Strukturinformationen des Gehirngewebes zur Verfügung. Durch Anwendung des Skull-Stripping wird direkt die Oberfläche des Gehirns extrahiert. Diese kann wiederum für morphometrische Analysen herangezogen werden oder in nachgeschaltete Verarbeitungsschritte – beispielsweise die Registrierung – als zusätzliche Information einfließen.

Das Skull-Stripping stellt damit insbesondere in der multimodalen Bildverarbeitung einen wichtigen Vorverarbeitungsschritt dar. Dieser Teilbereich der Bildverarbeitung befasst sich mit dem Vergleich und der Fusionierung von Bilddaten aus unterschiedlichen Quellen (Modalitäten) mit dem Ziel, durch diese Kombination die Aussagefähigkeit im Vergleich zur monomodalen Datenakquirierung zu erhöhen.

Diese Arbeit wurde in Kooperation mit der Arbeitsgruppe *Multimodale Bildverarbeitung* des *Instituts für Neurowissenschaften und Biophysik (INB-3)* im Rahmen des von der *Deutschen Forschungsgemeinschaft (DFG)* geförderten Projektes PA1595/1-1 am Forschungszentrum Jülich erstellt. Dieses Projekt trägt den Titel: „*In vitro vs. in vivo – Softwarebasierte Informationsfusion aus multimodaler Bildgebung bei Rattenhirnen*“. Es verfolgt das Ziel, eine softwarebasierte Fusion von histologischen Schnittbildern mit MRT Daten von Rattenhirnen zu realisieren.

Die vorliegende Arbeit beschäftigt sich als Teilaspekt des genannten Projektes mit dem Skull-Stripping von Rattenhirn MRTs. Bei der Erstellung von histologischen Schnitten verliert man Positionsinformationen orthogonal zur Schnittebene vollständig. Die affine und nicht-lineare Registrierung von Schnittmodalitäten zurück zu einem dreidimensionalen Datensatz wird durch Einbindung des Skull-Stripping-Segmentierungsergebnisses stark vereinfacht.

Auf Grundlage der Veröffentlichung von Zhuang [Zhu06] sollte das Skull-Stripping der Rattenhirnaufnahmen mit Level-Set-Methoden erfolgen. Level-Set-Methoden als mathematisches Konturpropagationsmodell besitzen ein breites Anwendungsspektrum in allen Bereichen, in denen Konturen oder Oberflächen dynamisch deformiert werden. Dazu gehören die Computergrafik, Anwendungen in der Geometrie oder die hier behandelten Segmentierungsaufgaben.

Kapitel 2

Stand der Forschung

Dieses Kapitel gibt zunächst einen kurzen Überblick über Segmentierungsverfahren im allgemeinen und stellt darauf die speziell für das *Skull-Stripping* verwendeten Algorithmen zusammenfassend dar. Ein besonderes Augenmerk wird dabei auf Zusammenhänge und Unterschiede zwischen Aktive-Kontur-Modellen und den verwandten Level-Set-Methoden gelegt.

2.1 Das Segmentierungsproblem

Die Bildsegmentierung kann als pixel- bzw. voxelweise Klassifikation von Objekten in Bildern aufgefasst werden. Vereinfacht ausgedrückt muss für jeden Bildpunkt eine Entscheidung getroffen werden, ob dieser zu dem zu segmentierenden Objekt gehört oder nicht. Diese binäre Entscheidung kann beispielsweise in einem *Maskenbild* abgespeichert werden. Die Ränder des Objektes werden durch die im Binärbild vorhandenen Diskontinuitäten beschrieben [Jä02]. Eine exakte Definition der Segmentierung findet sich beispielsweise in [Ser06].

2.1.1 Segmentierungsverfahren

Zur Lösung des Segmentierungsproblems sind verschiedene Verfahren entwickelt worden. Diese lassen sich grob in die folgenden Kategorien untergliedern.

- *Schwellenwertbasiert*: Hierbei entscheidet allein der Grauwert eines jeden Pixels über die Zugehörigkeit zu der jeweiligen Region. Überschreitet dieser eine zuvor festgelegte Schwelle, so gehört er zur entsprechenden Region. Dieses ist mit Abstand das einfachste, wohl aber auch das am häufigsten eingesetzte Segmentierungsverfahren.
- *Kantenbasiert*: Diese Verfahren nutzen die Eigenschaften der Objektgrenzen, wo der Helligkeitsgradient des Bildes ein betragsmäßiges Maximum annimmt. Eine auf diese Weise gefundene Kante muss dann noch mit einem entsprechenden Algorithmus verfolgt werden. Da ein Objekt eine zusammenhängende Region beschreibt, ist auch das verfolgte Maximum des Gradienten eine geschlossene Kurve.

- *Regionenbasiert*: Die zu segmentierenden Bildregionen weisen ähnliche Eigenschaften auf. Genau auf dieser Ähnlichkeit zusammenhängender Pixel basieren Algorithmen wie das Region-Growing, das Split-and-Merge Verfahren oder auch das Pyramid Linking. In diesem Fall ist die Wahl eines Homogenitätskriteriums zur Definition des Ähnlichkeitsbegriffs notwendig.
- *Atlasbasiert*: Für den Fall, dass die zu segmentierenden Objekte immer einen ähnlichen Aufbau besitzen, wie es in begrenztem Maße beim Skull-Stripping der Fall ist, kann man mit Hilfe der Registrierung ebenfalls eine Segmentierung durchführen. Dazu registriert man den zu segmentierenden Datensatz mit einem Atlas – ein Referenzdatensatz, in dem die interessierenden Regionen sorgfältig segmentiert vorliegen. Daraus erwächst implizit der Vorteil, dass im Bild nur schlecht darstellbare Subregionen ebenfalls grob lokalisiert werden können, vorausgesetzt die Daten entsprechen in ihrer Morphologie weitestgehend denen des Atlanten.
- *Morphologiebasiert*: Hierbei wird die Segmentierung mit Hilfe von morphologischen Operatoren durchgeführt. Ein gängiges Beispiel hierfür ist die Wasserscheidentransformation.
- *Formmodellbasiert*: Von dem zu segmentierenden Objekt wird ein statistisches Formmodell trainiert. Mit einer Hauptkomponentenanalyse (PCA) werden diejenigen Parameter bestimmt, welche die größten Variationen dieses Modells beschreiben können. Die eigentliche Segmentierung ist dann ein iterativer Prozess, welcher das Modell durch Translation, Rotation, Skalierung und Deformation mittels der gefundenen Parameter möglichst gut auf den Bildmerkmalen lokalisiert.
- *Konturmodellbasiert*: Hier werden Modelle für Konturbeschreibungen verwendet, welche sich iterativ der Form des zu segmentierenden Objektes annähern. Dazu zählen Snakes und die Level-Set-Methoden. Die Entwicklung der so beschriebenen Objektgrenzen basiert wiederum auf statistischen, regionenbasierten oder kantenbasierten Eigenschaften oder einer Kombination daraus.

Eine Kombination der kategorisierten Verfahren ist ebenfalls möglich – das führt zu den hybriden Verfahren.

2.1.2 Hybride Verfahren

Jedes dargestellte Verfahren weist in der Regel seine Vor- und Nachteile auf. Aus diesem Grund werden *hybride Verfahren* entwickelt, um möglichst die Nachteile eines Verfahrens mit den Vorteilen eines anderen Verfahrens auszugleichen. In der Literatur werden auch die im Abschnitt 2.3.3 dargestellten Initialisierungsansätze, welche die Startkontur für einen Konturevolutionsprozess darstellen, gerne als hybride Verfahren bezeichnet, auch wenn hier eine reine Hintereinanderausführung von Bildverarbeitungsoperationen vorgenommen wird.

Ein Beispiel für ineinander verschränkte Verfahren ist die Kombination von Registrierung und Level-Set-Methoden, welche von Duay et al. vorgeschlagen wird [Dua05]. Dabei handelt es sich um einen dreistufigen Registrierungsprozess eines Atlanten mit dem zu segmentierenden Datensatz, wobei die Freiheitsgrade von einer Stufe zur nächsten stetig erhöht werden. In der ersten Stufe wird zunächst eine affine Registrierung des Atlanten auf den aktuellen Datensatz durchgeführt. Eine affine Registrierung im dreidimensionalen Raum besitzt zwölf Freiheitsgrade. Die zweite Stufe benutzt eine nicht-rigide, globale Registrierung mit B-Splines als Basisfunktionen. Die Anzahl der genutzten Basisfunktionen ist direkt mit der Anzahl der Freiheitsgrade dieser Stufe verbunden. Damit diese Registrierung eine hinreichende Genauigkeit aufweist, sind gemäß den Autoren eine Vielzahl von Tests mit unterschiedlichen Skalen (Anzahl der Basisfunktionen) und Ortsauflösungen notwendig. Um dieses Problem zu umgehen, wird eine geringere Genauigkeit dieser Stufe gefordert und eine dritte ergänzt, welche die Level-Set-Methoden verwendet. Die Objektkontur wird hierdurch verformt und im Laufe dieser Deformation verfolgt, so dass hieraus ein Deformationsfeld berechnet werden kann. Hierfür wird auf das Gradient-Vector-Flow-Feld von Xu und Prince zurückgegriffen [Xu97, Xu98]. Dieses Feld wird wiederum für eine abschließende Registrierung genutzt.

Die Vielzahl der kategorisierten Ansätze, welche in beliebiger Form kombiniert werden können, legt nahe, dass es eine einheitlich richtige Lösung des Segmentierungsproblems nicht gibt [Ser06, Lap02]. Daher ist es eine wesentliche Aufgabe für das jeweilige Problem einen adäquaten Ansatz zu wählen und diesen für die jeweilige Anwendung zu modifizieren.

Eine sehr flexible und mittlerweile etablierte Möglichkeit der Bildsegmentierung stellen die aktiven Konturmodelle, insbesondere aber die Level-Set-Methoden dar. Diese wurden im Rahmen dieser Arbeit eingesetzt, so dass der historische und mathematische Hintergrund hier nun näher betrachtet wird.

2.2 Aktive Konturmodelle

Die aktiven Konturmodelle wurden von Kass et. al. [Kas88] 1988 vorgestellt, welche von den Autoren im zweidimensionalen Raum in einer expliziten Formulierung als so genannte *Snakes* implementiert wurden. Seitdem wurde dieser Ansatz für die Lösung von Segmentierungsproblemen herangezogen (siehe beispielsweise [Atk98]).

2.2.1 Snakes

Bei den Snakes handelt es sich um Splines, welche eine Optimierung ihrer Lage und Form mittels einer Energieminimierung durchführen. Dabei folgen sie der kontrollierten Kontinuitätsbedingung von Terzopoulos, wodurch eine Regularisierung in Form einer Stabilisierungsenergie vorgenommen wird [Ter86]. Die hierfür

genutzten Energieterme werden in *interne* und *bildbasierte* Energien unterteilt, welche je nach Gewichtung unterschiedliche Eigenschaften der Splines selbst oder Bildeigenschaften stärker oder schwächer bewerten. Hierbei handelt es sich um die explizite Formulierung einer aktiven Kontur.

Sei $\mathcal{C}(s) = \begin{pmatrix} x(s) \\ y(s) \end{pmatrix} : [0; 1] \rightarrow \mathbb{R}^2$ eine über s parametrisierte Kurve, dann wird die zugehörige Gesamtenergie E entlang von \mathcal{C} definiert als:

$$\begin{aligned} E &= \int_0^1 E_{\text{snake}}(\mathcal{C}(s)) ds \\ &= \int_0^1 E_{\text{int}}(\mathcal{C}(s)) + E_{\text{image}}(\mathcal{C}(s)) ds \end{aligned} \quad (2.1)$$

Die interne Energie E_{int} wird auch als Spline-Energie bezeichnet und bedingt bei stärkerer Gewichtung eine Glättung der Kurve. Für dessen Berechnung werden die erste und die zweite Ableitung der Kurve \mathcal{C} entlang s gebildet und diese untereinander gewichtet.

$$E_{\text{int}} = \frac{1}{2} (a(s) \|\mathcal{C}_s(s)\|^2 + b(s) \|\mathcal{C}_{ss}(s)\|^2) \quad (2.2)$$

Der Term erster Ordnung in (2.2), welcher mit $a(s)$ gewichtet wird, kontrolliert die innere Spannung der Snake, während der Term zweiter Ordnung die Steifigkeit mit der Gewichtung $b(s)$ bewertet.

Von entscheidender Bedeutung ist der bildbasierte Energieterm, welcher für die jeweilige Anwendung angepasst wird. Es muss ein Energieterm gefunden werden, der die aktive Kontur auf dem Rand der zu segmentierenden Region minimiert. Bereits von Kass et al. wurden mögliche Bildmerkmale genannt, aus denen sich die bildbasierte Energie additiv aufbaut:

$$E_{\text{image}} = w_{\text{line}} E_{\text{line}} + w_{\text{edge}} E_{\text{edge}} \quad (2.3)$$

Die Linienenergie wird beschrieben durch E_{line} und kann im einfachsten Fall gleich der Bildhelligkeit $E_{\text{line}} = I(\vec{x})$ gesetzt werden, wenn beispielsweise nach dunklen Linien auf einem hellen Hintergrund gesucht wird. Linien in Bildern können aber auch besonders gut durch die zweite Ableitung detektiert werden, dazu setzt man $E_{\text{line}} = -\nabla^2 I(\vec{x})$.

Die Kantenenergie kann durch die negative quadrierte Norm des Helligkeitsgradienten des Bildes bestimmt werden: $E_{\text{edge}} = -|\nabla I(\vec{x})|^2$. Kanten mit steilen Gradienten produzieren dann hohe negative Werte.

Durch eine Modifikation der Energieterme in (2.3) oder eine additive Ergänzung von weiteren Termen, kann der Optimierungsprozess auf die jeweilige Fragestellung individuell zugeschnitten werden.

Fast alle auf diesem Ansatz aufbauenden Weiterentwicklungen beschränken sich bei der Glättungsenergie (2.2) auf den Term der ersten Ordnung: $b(s) = 0$. Auch

die Möglichkeit einer ortsabhängigen Steuerung der Glättung wird in der Regel nicht genutzt, so dass a zu einer Konstanten wird: $a'(s) = 0$. Mit diesen Vereinfachungen ergibt sich dann die Snake-Energie mit einem verallgemeinerten Kantendetektor $g(\nabla I(\vec{x}))$ zu:

$$E = \mu \int_0^1 |\mathcal{C}(s)|^2 ds + \lambda \int_0^1 g(\nabla I(\mathcal{C}(s))) ds \quad (2.4)$$

Der klassische Ansatz der Beschreibung von aktiven Konturen mit Hilfe der expliziten Formulierung hat unter anderem den Nachteil, dass Topologiewechsel einer gesonderten Behandlung bedürfen. Die einmal durch die initiale Kontur vorgegebene Topologie wird bei der Energieminimierung nicht verändert. Wenn in einem Bild mit dieser Methode Regionen unbekannter Zahl gefunden werden müssen, dann werden folglich Split- und Merge-Algorithmen benötigt, die diese Topologiewechsel vornehmen (siehe beispielsweise [Sze93]). Diesen Nachteil, der durch die explizite Formulierung der aktiven Konturen begründet ist, umgeht die implizite Formulierung der Konturen mit Hilfe der Level-Set-Formulierung.

2.2.2 Implizite Konturen und Oberflächen

Die wesentliche Idee der Level-Sets besteht darin, dass Konturen oder Oberflächen nicht explizit in ihrer Lage beschrieben werden, wie es bei den Snakes der Fall ist, sondern implizit durch eine Höhenlinie einer höherdimensionalen Funktion $\phi : \mathbb{R}^{\dim(I)} \rightarrow \mathbb{R}$. Diese Funktion ist in der Regel eine vorzeichenbehaftete Distanzfunktion. Wenn die Nullhöhenlinie als Rand eines Objektes definiert wird, dann ist dessen Kontur implizit durch den Nulldurchgang der Funktion ϕ beschrieben. Für eine ausführliche Darstellung sei hier auf Kapitel 3 verwiesen.

Die Level-Set-Methoden befassen sich mit der iterativen, zeitlichen Entwicklung der Distanzfunktion und damit mit der Deformation der beschriebenen Oberflächen in Form von partiellen Differentialgleichungen. Die ursprüngliche Formulierung wurde von Osher und Sethian 1988 veröffentlicht [Osh88], im gleichen Jahr wie die aktiven Konturmodelle von Kass et al. [Kas88]. Begründet auf der Ähnlichkeit der Modelle wurden die Level-Sets zunächst von Caselles in [Cas93] im Bereich Computer Vision für die Kontur- bzw. Oberflächenmodellierung verwendet. Ähnliche Ansätze folgten durch Malladi in [Mal93]. Eine Überführung der Modelle ineinander erfolgte wiederum über die geodätischen aktiven Konturen von Caselles [Cas97].

Bei diesem geodätischen Ansatz wird zunächst eine Metrik im Riemannraum erstellt, der die Länge der Kurve mit dem Ergebnis des allgemeinen Kantendetektors $g(\nabla I)$ gewichtet. Die hieraus resultierende Kurvenentwicklung

$$\frac{\partial \mathcal{C}(t)}{\partial t} = g(\nabla I) \kappa \vec{\mathcal{N}} - (\nabla g \cdot \vec{\mathcal{N}}) \vec{\mathcal{N}} \quad (2.5)$$

wird mit Hilfe der Level-Set-Formulierung von Osher und Sethian [Osh88] ausgedrückt, wobei κ die Euklidische Krümmung und $\vec{\mathcal{N}}$ die nach innen zeigende

Normale der Kurve \mathcal{C} ist. Die hieraus entwickelte intrinsische Formulierung ist nun nicht mehr von einer beliebigen Parametrierung der Krurve \mathcal{C} abhängig:

$$\begin{aligned}\frac{\partial\phi}{\partial t} &= |\nabla\phi|\nabla \cdot \left[g(\nabla I) \cdot \left(\frac{\nabla\phi}{|\nabla\phi|} \right) \right] \\ &= \nabla g(\nabla I) \cdot \nabla\phi + |\nabla\phi|g(\nabla I)\kappa\end{aligned}\quad (2.6)$$

Unter Verwendung der Produktregel und der Kenntnis, dass die Krümmung

$$\kappa = \nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|} \right) \quad (2.7)$$

ist, lässt sich Gleichung (2.6) wie dargestellt umformen.

Die zeitliche Entwicklung von ϕ in (2.6) wird durch den Kantendetektor $g(\nabla I)$ und die Krümmung κ gesteuert. Als Kantendetektor wird häufig eine Funktion der Form

$$g(\nabla I(\vec{x})) = \frac{1}{1 + |G_\sigma * \nabla I(\vec{x})|^p} \quad (2.8)$$

herangezogen [Cas93, Mal93, Mal96], wobei $G_\sigma * \nabla I(\vec{x})$ die mit einem Gaußschen-Filterkern G_σ mit einer Breite von σ geglättete Version des Gradientenbildes von $I(\vec{x})$ und $p \geq 1$ ist. Beispielsweise ist in [Cas93] $p = 2$ und in [Mal93, Zhu03] $p = 1$ gewählt worden. Die Kantendetektionsfunktion (2.8) ist unabhängig von der Wahl von p in homogenen Bereichen des Eingangsbildes Eins und konvergiert an Kanten gegen Null. Hieraus lässt sich für die Funktion g ableiten, dass diese positiv, monoton fallend und für hohe Werte gegen Null konvergieren muss.

Bisher hat der Kantendetektor g eine zentrale Rolle eingenommen, um die Bewegung der Oberfläche an den gewünschten Stellen zu stoppen. Einen grundsätzlich anderen Ansatz verfolgen Chan und Vese, die eine Segmentierung mit Hilfe des Mumford-Shah Funktionals durchführen [Cha01]. Auch hier basiert die Segmentierung auf einer Energieminimierung. Allerdings ist nicht mehr der Gradient das Optimierungskriterium, sondern viel mehr wird hier eine möglichst hohe Homogenität der Grauwerte innerhalb der Region angestrebt.

Die Energie, die hier von Bedeutung ist, lässt sich aus der Level-Set-Funktion ϕ der Heavisidefunktion H_ϵ (siehe Abschnitt 3.1) und dem Deltafunktional δ_ϵ berechnen:

$$\begin{aligned}E_{\text{MS},\epsilon} &= \eta \int_{\Omega} \delta_\epsilon(\phi(\vec{x})|\nabla\phi(\vec{x})|) d\vec{x} \\ &\quad + \nu \int_{\Omega} H_\epsilon(\phi(\vec{x})) d\vec{x} \\ &\quad + \lambda_1 \int_{\Omega} |I(\vec{x}) - c_1|^2 H_\epsilon(\phi(\vec{x})) d\vec{x} \\ &\quad + \lambda_2 \int_{\Omega} |I(\vec{x}) - c_2|^2 (1 - H_\epsilon(\phi(\vec{x}))) d\vec{x}\end{aligned}\quad (2.9)$$

Der erste Summand der angegebenen Energiegleichung beschreibt die Länge der Kontur $\partial\Omega$ und der zweite die Größe der eingeschlossenen Fläche. Damit stellen

diese beiden Terme eine Regularisierung der Optimierung dar. Die letzten beiden Summanden bewerten die Fläche innerhalb und außerhalb der Kontur, indem die quadratische Abweichung zu den vorgegebenen Konstanten c_1 und c_2 ermittelt wird. Der Gradient spielt hier keine Rolle mehr, so dass dieser Ansatz auch als *active contours without edges* bezeichnet wird.

Mit Hilfe der Variationsrechnung und der Euler-Lagrange-Differentialgleichung lässt sich auch die Minimierung von Gleichung (2.9) als zeitliche Variation beschreiben.

$$\frac{\partial \phi}{\partial t} = |\nabla \phi| \left(\eta \nabla \cdot \left(\frac{\nabla \phi}{|\nabla \phi|} \right) - \nu - \lambda_1 (I - c_1)^2 + \lambda_2 (I - c_2)^2 \right) \quad (2.10)$$

Ein Vorteil dieser Formulierung besteht insbesondere darin, dass die Segmentierung auch bei einem geringen Signal zu Rausch-Verhältnis möglich ist, das heißt wenn der Gradient durch ein starkes Rauschen verzerrt ist. Dies wurde in [Cha01] eindrucksvoll dargestellt. Der regularisierende Parameter $\eta > 0$ wird auch als Skalenparameter verwendet.

Die Rolle der Energieterme bei der Deformation der Snakes übernimmt bei den Level-Set-Methoden eine Geschwindigkeitsfunktion, die dafür sorgt, dass die Level-Set-Funktion an den entsprechenden Stellen stoppt. Eine allgemeine Formulierung wird in Kapitel 3 dargestellt.

Mehrere unterschiedliche Objekte zu segmentieren ist mit nur einer Level-Set-Funktion, wie sie bisher beschrieben worden ist, nicht möglich, denn die Zugehörigkeit zum Objekt wird durch das Vorzeichen der Level-Set-Funktion an der jeweiligen Stelle im Bild entschieden. Zhao et al. [Zha96] umgehen dieses Problem, indem sie mehrere solcher Level-Set-Funktionen verwenden und diese zeitlich entwickeln und so untereinander koppeln, dass weder ein Vakuum noch eine Überlappung auftritt. Ein Nachteil entsteht dadurch, dass die Anzahl von zu segmentierenden Regionen im Vorhinein bekannt sein muss. Brox und Weickert haben die Level-Set-Formulierung erweitert, so dass eine beliebige Anzahl von Regionen in einem Bild segmentiert werden kann [Bro04].

2.3 Skull-Stripping

Im vorangegangenen Abschnitt lag der Fokus auf den mathematischen Modellen zur Segmentierung. Im Folgenden werden nun deren Anwendungen mit Bezug auf das so genannte Skull-Stripping dargestellt, bei dem aus einer Kopf-MR Aufnahme nur das Gehirngewebe segmentiert werden soll.

Für eine differenzierte Betrachtung der Forschungsschwerpunkte wird folgende Unterteilung vorgeschlagen:

- Entwicklung einer adäquaten Geschwindigkeitsfunktion,
- formbasierte Beschränkungen der Level-Set-Funktion,

- Automatisierung der Initialisierung.

In den folgenden Abschnitten werden die genannten Punkte an ausgewählten Publikationen exemplarisch dargestellt. In den Fällen, wo eine Zuordnung zu mehreren der oben aufgeführten Schwerpunkte möglich ist, wurde derjenige ausgewählt, der einen bedeutenderen Teil in der Publikation eingenommen hat.

2.3.1 Geschwindigkeitsfunktionen

Die Geschwindigkeitsfunktion bestimmt im Wesentlichen das Verhalten der Entwicklung der Level-Set-Funktion. Hier gilt es, besondere Sorgfalt bei der Auswahl einer adäquaten Funktion walten zu lassen, da diese das spätere Segmentierungsergebnis bestimmt. In der Regel werden aus den originalen und/oder vorverarbeiteten Daten Eigenschaften wie Bildhelligkeit oder Helligkeitsgradient verwendet um eine Geschwindigkeitsfunktion aufzustellen. Interne Glättungskräfte sind abhängig von der Form der Level-Set-Funktion und werden im Laufe der Entwicklung lokal und zeitlich abhängig berechnet. Aber auch statistische Bildeigenschaften werden unter anderem bei der Berechnung der Geschwindigkeitsfunktion berücksichtigt.

Ein früher Einsatz der Snakes für den Segmentierungsprozess von Gehirngewebe aus 3D-MR-Daten findet man bei Atkins et al. [Atk98]. Ein dreistufiger Ansatz entfernt zunächst histogrammbasiert den verrauschten Bildhintergrund, wobei alle anatomischen Strukturen erhalten bleiben. Der zweite Schritt dient zur Erzeugung einer initialen, intrakraniellen Gehirnmaske. Die nichtlineare Diffusion, ursprünglich vorgeschlagen von Perona und Malik [Per90], wird hier als Vorfiltrierung eingesetzt. Auf dieses geglättete Bild wird eine einfache Schwelle angewendet, welche so gewählt wird, dass die hieraus resultierende Kontur möglichst komplett innerhalb des Gehirns liegt. Der dritte Schritt greift auf die aktiven Konturen und das Prinzip der Energieminimierung zurück. Vier Energieterme sorgen für eine korrekt Lokalisierung der Kontur. Dazu gehören eine *Kontinuitätsenergie*, welche die Glattheit der Kurve bestimmt, eine *Ballonenergie*, welche die Kurve nach außen drückt, eine *Intensitätsenergie*, welche die Kurve auf dunkle Bildbereiche schiebt und eine *Gradientenenergie*, welche die Kurve an hohen Gradienten des Diffusionsbildes lokalisiert. Bis auf die Ballonenergie finden sich diese Energieterme, abgesehen von geringfügigen Modifikationen, bereits bei Kass et al. [Kas88].

Cheng et al. haben das im vorherigen Abschnitt dargestellte Modell des Mumford-Shah-Segmentierers [Cha01] verallgemeinert und mittels Level-Set-Methoden für Gehirngewebe-Segmentierung angepasst [Che05]. Die Geschwindigkeitsfunktion ist damit nicht mehr abhängig von im Bild vorhandenen Gradienten. Die wesentliche Neuerung besteht darin, dass hier zwei separate Level-Set-Funktionen verwendet und zeitlich entwickelt werden. Anhand der Vorzeichen dieser beiden Funktionen kann der Raum nun nicht mehr nur in zwei unterschiedliche Bereiche aufgeteilt werden, sondern in vier. Diese Regionen sind weiße und graue Substanz sowie die Zerebrospinalflüssigkeit (CSF) und der Hintergrund. Die Homogenität

bezüglich der Grauwerte wird auch hier innerhalb der einzelnen Regionen maximiert.

Das *Brain Extraction Tool*, auch bekannt unter dem Akronym *BET*, ist ein frei verfügbares Gehirnsegmentierungsprogramm welches am FMRIB-Centre der University of Oxford im Rahmen der FMRIB-Software-Library (FSL) entwickelt und implementiert worden ist [Smi02]. Es basiert auf einem deformierbaren Netz, an dessen Knoten Formkräfte angreifen. Ein ähnlicher, netzbasierter Ansatz wird auch von Dale in [Dal99] verwendet. Das Netz wird beim Brain Extraction Tool im Laufe des Segmentierungsprozesses iterativ an die Gehirnkontur angepasst und sukzessive feinmaschiger bis schließlich ein Konvergenzkriterium erreicht ist. Die Netzoberfläche kann als eine Annäherung an das Zero-Level-Set einer Level-Set-Funktion betrachtet werden. Die auf die Knoten orthogonal zur Oberfläche angreifenden Formkräfte kann man dann direkt als Geschwindigkeitsfunktion der Level-Set-Entwicklung interpretieren – diesen Ansatz verfolgen Zhuang et al. [Zhu06]. Die Neuerung bei Zhuang et al. besteht im Wesentlichen aus dem Transfer von netzbasierten Formkräften in die Geschwindigkeitsfunktion für die Level-Set-Entwicklung auf zweidimensionalen Bilddaten. Diese Geschwindigkeitsfunktion wird basierend auf lokalen Bildeigenschaften in Abhängigkeit von Position und Oberflächennormalen berechnet, woraus sich eine zeitliche Abhängigkeit ergibt. Bis zu diesem Zeitpunkt wurde die Geschwindigkeit einmalig vor der eigentlichen Level-Set-Entwicklung berechnet. Einen ähnlichen Weg verfolgen auch Palm und Piertzkyk zur Rattenhirnsegmentierung von histologischen Schnitten [Pal08]. Die lokale Adaptivität macht das Modell flexibler, gleichzeitig steigt aber auch der Rechenaufwand. Die weite Verbreitung und die vielversprechenden Segmentierungsergebnisse des Brain Extraction Tools, sowie die Flexibilität des Level-Set-Modells waren Motivation für die im Rahmen dieser Arbeit durchgeführte Erweiterung des Segmentierers von Zhunag et al. für dreidimensionale Datensätze.

2.3.2 Formbeschränkungen

In Regionen mit schwachem Kontrast kann es vorkommen, dass die im Bild vorhandene Information für eine hinreichend genaue Segmentierung nicht ausreicht. Aus diesem Grund werden von einigen Autoren zusätzliche Informationen über die gesuchte Form herangezogen. Dies geschieht entweder über statistische Formmodelle oder über Atlanten.

Eine Möglichkeit wird von Ciofalo in [Cio04, Cio05] vorgestellt. Hierzu wird auf Atlasinformationen zurückgegriffen aus denen so genannte Fuzzy-Maps erstellt werden. Eine affine Registrierung bildet die im MNI-Brainweb-Atlas vorliegenden Informationen auf den aktuell zu segmentierenden Datensatz ab. Die erstellten Fuzzy-Maps dienen der Steuerung der Parameter der Geschwindigkeitsfunktion. So wird in Regionen mit erwartungsgemäß stark gekrümmten Oberflächen auch eine hohe Krümmung zugelassen. Gleiches gilt für Parameter, die die Propagation der Kontur in Richtung der gewünschten Region steuern. Auf diesem Ansatz

aufbauend, wird von Ciofalo in [Cio05] eine Methode mit zueinander in Konkurrenz stehenden Level-Set-Funktionen entwickelt, wobei die relative Lage der Funktionen zueinander dem Fuzzy-Controller zusätzliche Variablen für die Geschwindigkeitsfunktion liefert.

Im Gegensatz zu den atlasbasierten Informationen bei Ciofalo stellt Leventon in [Lev00] ein statistisches Modell der Form des zu segmentierenden Objektes auf. Dieses basiert auf den vorzeichenbehafteten Distanzfunktionen von einer Vielzahl manuell segmentierter Referenzdatensätze. Bei einer hinreichend großen Stichprobe bei der Erstellung des Modells ist auch gleichzeitig eine Information über die Variabilität der Form bekannt. Bei der Evolution der Level-Set-Funktion wird ein zusätzlicher Summand eingeführt, der die aktuelle Kontur in die Richtung einer geschätzten Kontur zieht. Hierfür wird eine maximum-a-posteriori Wahrscheinlichkeit für die Position und Form der Kontur in Abhängigkeit der aktuellen Kontur berechnet. Zusätzlich werden Gradienteninformationen genutzt um die Kontur auf bildbasierte Merkmale zu lokalisieren.

Für eine hinreichend hohe, statistische Aussagekraft ist eine hohe Anzahl von manuell segmentierten Datensätzen notwendig. Aus diesem Grund ist eine sinnvolle Einbindung von Forminformationen in dieser Arbeit leider nicht möglich.

2.3.3 Initialisierungsansätze

Die Level-Set-Methoden beschreiben eine iterative Entwicklung der Oberfläche. Die Anzahl der insgesamt benötigten Iterationen lässt sich deutlich reduzieren, indem eine möglichst gute Startkontur gewählt wird. Dies motivierte beispielsweise Ségonne [Seg04], Baillard [Bai01] Zhu [Zhu03] und Zhuang [Zhu06] mittels unterschiedlicher Methoden eine Festlegung der Startkontur zu finden.

Ségonne [Seg04] verwendet einen Wasserscheidenalgorithmus für eine Vorsegmentierung, welche die Startkontur der anschließenden Level-Set-Segmentierung darstellt. Die Wasserscheidensegmentierung in der Variante von Hahn und Peitgen [Hah00] schwächt das Problem der Übersegmentierung ab.

Auch die Registrierung mit einem Atlas, in dem die Objektgrenzen des zu segmentierenden Objektes vermerkt sind, wird für eine grobe Vorsegmentierung herangezogen. Diesen Ansatz wählen Baillard et al. [Bai01]. Die Vorsegmentierung liefert die gewünschte Startkontur der Level-Set-Entwicklung. Dieses Vorgehen unterscheidet sich von dem durch Duay [Dua05] vorgestellten Ansatz insofern, als dass hier eine echte Initialisierung vorliegt und nicht die entwickelte Level-Set-Funktion wiederum genutzt wird, um ein Deformationsfeld zu berechnen.

Eine weitere Möglichkeit die Startkontur zu erstellen, ist die Verwendung der Fast-Marching-Methods, welche eine starke Verwandtschaft zu den Level-Set-Methoden aufweisen. Diese wählen Zhu et al. in [Zhu03].

Zhuang et al. [Zhu06] verwenden für die Initialisierung der Startkontur eine schwellenwertbasierte Abschätzung der Kopfposition und -größe im Bild, welche wiederum zu einer Abschätzung der Lage des Zentrums des Gehirns genutzt wird. In diese Abschätzung fließen empirische Beobachtungen ein. Die Startkontur ergibt sich als Kreis um das geschätzte Zentrum, wobei der Radius so gewählt wird, dass sich die gesamte Kontur innerhalb des zu segmentierenden Gehirns befindet.

In Anlehnung an die Initialisierung von Zhuang wird in dieser Arbeit eine Kugel mit einstellbarem Radius als Startkontur gewählt. Die Positionierung erfolgt jedoch entweder in der Bildmitte, oder an einer manuell gewählten Position. Auf diese Weise werden die empirisch ermittelten Parameter aus der Initialisierung entfernt, da eine Übertragbarkeit der zu Grunde liegenden Beobachtungen auf Rattenhirn-MRTs nicht direkt möglich ist.

Kapitel 3

Material und Methoden

Um eine detaillierte Beschreibung des ausgewählten Segmentierers von Zhuang [Zhu06] zu ermöglichen, erfolgt zunächst eine mathematische Formulierung der Grundlagen. Die hieraus entwickelten Level-Set-Methoden sind weitestgehend im *Insight Segmentation and Registration Toolkit (ITK)*, einer freien Open-Source C++ Programmierbibliothek, verfügbar. Hiermit ist eine Implementation des ausgewählten Verfahrens möglich.

3.1 Level-Set-Formulierung

Bei der Level-Set-Formulierung handelt es sich, bei einem Definitionsbereich $\Omega \in \mathbb{R}^n$, um die Darstellung von impliziten Punkten ($n = 1$), Kurven ($n = 2$) und Oberflächen ($n = 3$). Eine fortschreitende Erhöhung der Dimension n ist möglich, im Rahmen dieser Arbeit werden aber lediglich die interessierenden Fälle für $n = 2$ und $n = 3$ Dimensionen betrachtet.

Die Kurve bzw. Oberfläche, im Folgenden ohne Berücksichtigung der Dimension als Kontur bezeichnet, wird in ihrer Position nicht direkt beschrieben, wie es beispielsweise über eine explizite Formulierung bei den Snakes der Fall ist (siehe Abschnitt 2.2.1), sondern über den k -Durchgang einer höherdimensionalen, stetigen Funktion $\phi : \mathbb{R}^n \rightarrow \mathbb{R}$. Folglich handelt es sich um eine implizite Beschreibung. Die Kontur $\partial\Omega$ ist dann gegeben durch die Menge aller Punkte \vec{x} , an der die Level-Set-Funktion einen festgelegten Wert k annimmt:

$$\partial\Omega = \{\vec{x} \in \mathbb{R}^n \mid \phi(\vec{x}) = k\} \quad (3.1)$$

Die Kontur selbst hat die Dimension $n - 1$.

Damit die Level-Set-Funktion einen k -Durchgang hat und somit eine Kontur definiert, muss sie an den übrigen Punkten von k verschieden sein. Die Wertigkeit der Level-Set-Funktion entscheidet, ob ein Punkt zur Teilmenge $\Omega_+ \subset \Omega$ oder $\Omega_- \subset \Omega$ gehört. Diese Teilmengen sind wie folgt definiert:

$$\begin{aligned} \Omega_+ &= \{\vec{x} \in \mathbb{R}^n \mid \phi(\vec{x}) > k\} \\ \Omega_- &= \{\vec{x} \in \mathbb{R}^n \mid \phi(\vec{x}) < k\} \end{aligned} \quad (3.2)$$

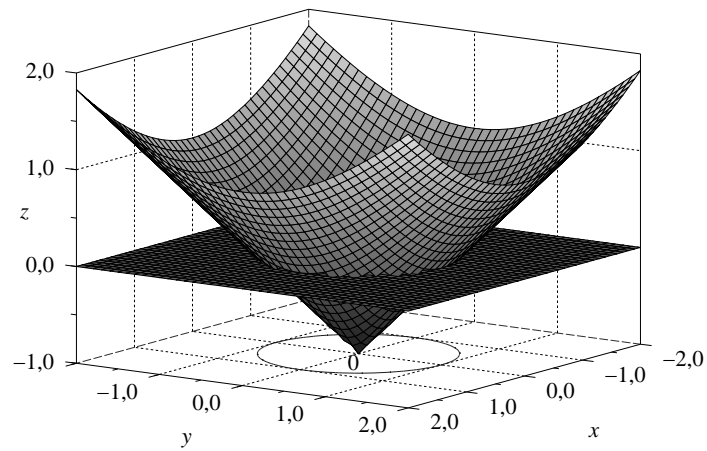
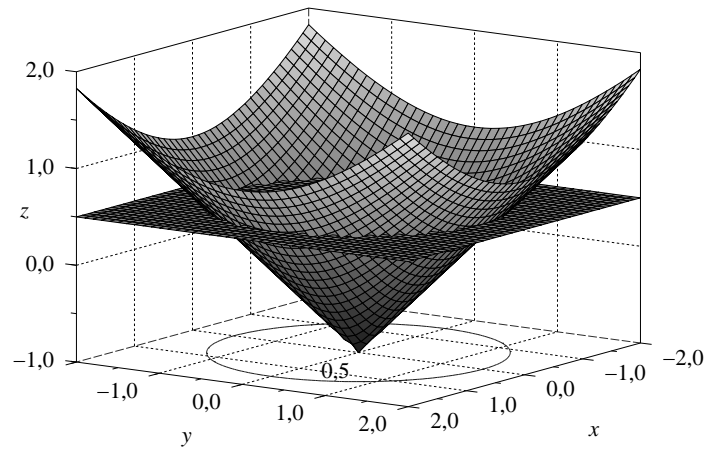
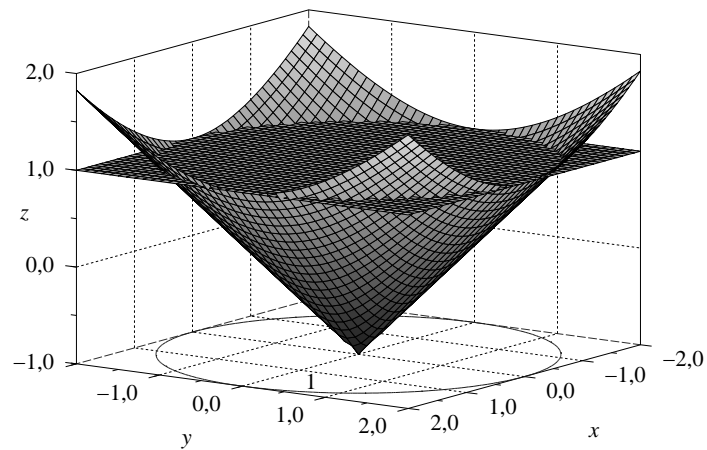
(a) $k = 0,0$ (b) $k = 0,5$ (c) $k = 1,0$

Abbildung 3.1: Iso-Level-Konturen der Funktion $\phi(\vec{x}) = \sqrt{x^2 + y^2} - 1$ für verschiedene Werte von k .

Für den zweidimensionalen Fall ist in Abbildung 3.1 exemplarisch die Funktion

$$\phi(\vec{x}) = \sqrt{x^2 + y^2} - 1 \quad (3.3)$$

mit $\vec{x} = \begin{pmatrix} x \\ y \end{pmatrix}$ für $k = \{0; 0,5; 1\}$ dargestellt. Die zugehörige Kontur ist als Projektion in der xy -Ebene markiert.

Wie man ebenfalls aus Abbildung 3.1 entnehmen kann, wird die Teilmenge Ω_+ als *Außen* und die Teilmenge Ω_- als *Innen* bezeichnet. Diese Konvention findet sich sowohl in der gängigen Literatur [Set99, Osh02], als auch in der Implementation von ITK.

Für den Spezialfall $k = 0$ spricht man auch vom *Zero-Level-Set*. Ist kein expliziter Wert für k angegeben, wird im weiteren Verlauf dieser Arbeit $k = 0$ angenommen – in diesem Fall entscheidet allein das Vorzeichen der Level-Set-Funktion, über die Zugehörigkeit eines Punktes zu Ω_+ oder Ω_- .

3.1.1 Werkzeuge

Gradient

Der *Gradient* der Level-Set-Funktion $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}$ ist definiert als

$$\nabla\phi := \left(\frac{\partial\phi}{\partial x}, \frac{\partial\phi}{\partial y} \right) \quad (3.4)$$

und zeigt in die Richtung des größten Anstiegs der Funktionswerte. Er steht immer senkrecht auf der Iso-Kontur. Wird der Gradient auf Eins normiert, so erhält man die Einheitsnormale \vec{N} , welche in dem dargestellten Fall (Abb. 3.1) nach außen zeigt.

$$\vec{N} = \frac{\nabla\phi}{|\nabla\phi|} \quad (3.5)$$

Äquivalente Formulierungen ergeben sich auch für dreidimensionale Level-Set-Funktionen. Die Normale existiert nur für diejenigen Fälle, in denen $|\nabla\phi| \neq 0$ ist.

Krümmung

Die *Krümmung* κ einer Kontur ist definiert als die Divergenz der Einheitsnormalen und beschreibt den reziproken Wert des Radius eines approximierten Tangentialkreises bzw. einer approximierten Tangentialkugel:

$$\kappa = \operatorname{div}(\vec{N}) = \operatorname{div}\left(\frac{\nabla\phi}{|\nabla\phi|}\right) \quad (3.6)$$

Die Krümmung wird zur Regularisierung der entwickelten Level-Set-Funktion herangezogen, um zu verhindern, dass beliebig stark gekrümmte Oberflächen entstehen.

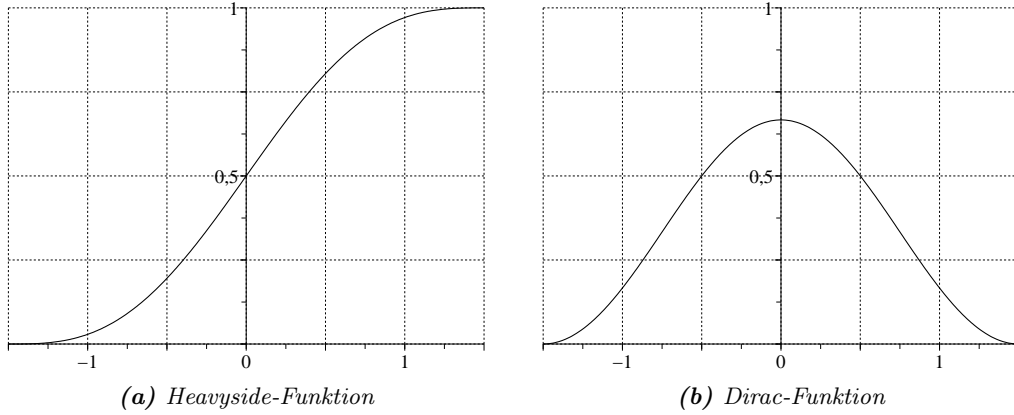


Abbildung 3.2: Die mit dem Parameter $\epsilon = 1,5$ weichgezeichnete Heavyside Funktion und das zugehörige Dirac-Funktional.

Heavyside-Funktion

Die *Heavyside-Funktion* H erfüllt, wie schon in Abschnitt 2.2.2 angesprochen, erfüllt im Rahmen der Level-Set-Methoden eine Reihe von Funktionen. Beispielsweise werden Flächen- und Umfangsberechnungen einer Kontur mit dieser Funktion durchgeführt [Osh02]. Darüber hinaus erlaubt sie die einfache Extraktion des Vorzeichens der Level-Set-Funktion.

$$H(\phi) = \begin{cases} 0 & \text{für } \phi \leq 0 \\ 1 & \text{für } \phi > 0 \end{cases} \quad (3.7)$$

Das Dirac-Funktional δ ist definiert als die erste Ableitung der Heavyside-Funktion.

$$\delta(\phi) = H'(\phi) = \begin{cases} \infty & \text{für } \phi = 0 \\ 0 & \text{sonst} \end{cases} \quad \text{mit} \quad \int_{-\infty}^{\infty} \delta(\phi) \, d\phi = 1 \quad (3.8)$$

Unter Berücksichtigung der Tatsache, dass das Dirac-Funktional fast überall gleich Null ist und dass auf diskreten, abgetasteten Datensätzen diese Punkte selten exakt erreicht werden, wird alternativ eine weichgezeichnete Version des Dirac-Functionals definiert, wobei der Parameter $\epsilon > 0$ den Grad der Weichzeichnung bestimmt. Siehe hierzu Abbildung 3.2.

$$H_\epsilon(\phi) = \begin{cases} 0 & \text{für } \phi < -\epsilon \\ \frac{1}{2} + \frac{\phi}{2\epsilon} + \frac{1}{2\pi} \sin\left(\frac{\pi\phi}{\epsilon}\right) & \text{für } -\epsilon \leq \phi \leq \epsilon \\ 1 & \text{für } \phi > \epsilon \end{cases} \quad (3.9)$$

Auch hier ergibt die erste Ableitung das weichgezeichnete Dirac-Funktional δ_ϵ .

$$\delta_\epsilon(\phi) = \begin{cases} 0 & \text{für } \phi < -\epsilon \\ \frac{1}{2\epsilon} \left(1 + \cos\left(\frac{\pi\phi}{\epsilon}\right) \right) & \text{für } -\epsilon \leq \phi \leq \epsilon \\ 0 & \text{für } \phi > \epsilon \end{cases} \quad (3.10)$$

3.1.2 Distanzfunktion

Für die Level-Set-Funktion wurde bisher nur eine Aussage darüber gemacht, inwiefern das Vorzeichen des Funktionswertes eines Punktes im Definitionsbereich für eine Klassifikation herangezogen werden kann (siehe (3.2)). Der Verlauf einer solchen Funktion kann jedoch für die gleiche Kontur beliebig sein, solange der Nulldurchgang der Funktion unverändert bleibt.

Eine besondere Teilmenge des Satzes der impliziten Funktionen stellen die vorzeichenbehafteten Distanzfunktionen dar. Eine Distanzfunktion $D(\vec{x})$ im Allgemeinen beschreibt für jeden Punkt des Definitionsbereiches den minimalen euklidischen Abstand zur geschlossenen Kontur $\partial\Omega$, welche zunächst als gegeben vorausgesetzt wird.

$$D(\vec{x}) = \min(|\vec{x} - \vec{x}_i|) \quad \forall \vec{x}_i \in \partial\Omega, \forall \vec{x} \in \Omega \quad (3.11)$$

Bedingt durch die Geschlossenheit der Kurve $\partial\Omega$ wird eindeutig ein inneres (Ω_-) und ein äußeres Gebiet (Ω_+) definiert. Die Zugehörigkeit eines Punktes \vec{x} zu Ω_+ oder Ω_- entscheidet dann über das Vorzeichen der Distanzfunktion, welche dann direkt als Level-Set-Funktion genutzt werden kann.

$$\phi(\vec{x}) = \begin{cases} -D(\vec{x}) & \text{für } \vec{x} \in \Omega_- \\ D(\vec{x}) & \text{für } \vec{x} \in \Omega_+ \\ 0 & \text{für } \vec{x} \in \partial\Omega \end{cases} \quad (3.12)$$

Die so definierte Level-Set-Funktion ist über ihre Nullstellen, respektive über die Kontur hinweg monoton. Diese Eigenschaft ist bedingt durch den Vorzeichenwechsel und verhindert somit die Unstetigkeit auf der Kontur.

Die Berechnung der Einheitsnormalen \vec{N} aus (3.5) vereinfacht sich dann zu:

$$\vec{N} = \nabla\phi(\vec{x}) \quad (3.13)$$

Die Normierung kann entfallen, da $|\nabla\phi| = 1$ ist. Das gilt für alle Punkte \vec{x} , zu denen es zur minimalen Distanz nur genau einen Punkt \vec{x}_i auf der Kontur gibt. Der Gradient der Distanzfunktion aus (3.3) ist in Abbildung 3.3 zusätzlich in der xy -Ebene dargestellt.

Die in Abbildung 3.1 dargestellte Funktion ist eine Distanzfunktion zur Kontur $\partial\Omega = \{\vec{x} \in \Omega \mid |\vec{x}| = 1\}$. Die Funktion ist im Punkt $\vec{x} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ nicht differenzierbar, was zu numerischen Instabilitäten beispielsweise bei der Berechnung von Ableitungen führen kann.

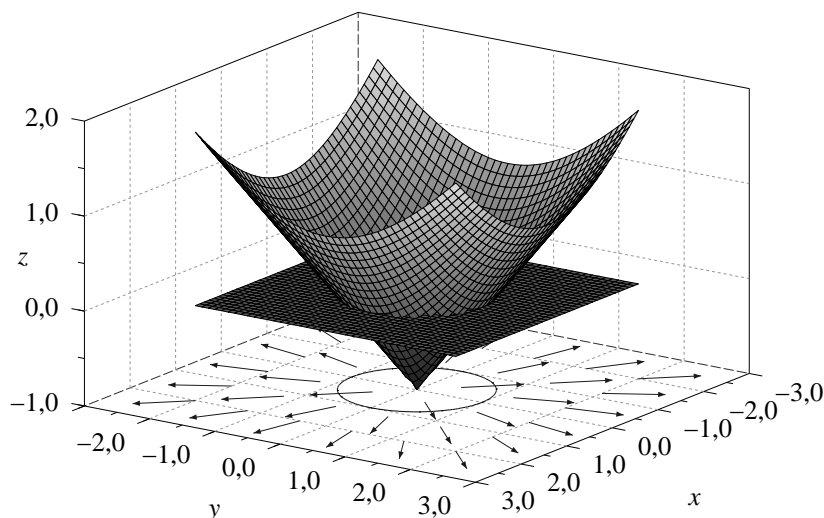


Abbildung 3.3: Distanzfunktion gegeben durch Gleichung (3.3) mit dem Gradienten in der xy -Ebene und der Zero-Level-Set-Kontur.

Ein praktischer Vorteil für die Benutzung von vorzeichenbehafteten Distanzfunktionen liegt auch in der einfachen Berechnung der Krümmung der Kontur. Dadurch, dass die Norm des Gradienten (fast) überall eins ist, vereinfacht sich die Berechnung für den dreidimensionalen Fall zu:

$$\kappa = \operatorname{div} \nabla \phi = \Delta \phi = \phi_{xx} + \phi_{yy} + \phi_{zz} \quad (3.14)$$

Das ist ein enormer Gewinn im Vergleich zur Berechnung der Krümmung einer allgemeinen Level-Set-Funktion, welche im Anhang A.1 für den dreidimensionalen Fall hergeleitet wird:

$$\kappa = \frac{\phi_x^2 \phi_{yy} + \phi_x^2 \phi_{zz} + \phi_y^2 \phi_{xx} + \phi_y^2 \phi_{zz} + \phi_z^2 \phi_{xx} + \phi_z^2 \phi_{yy} - 2\phi_x \phi_y \phi_{xy} - 2\phi_x \phi_z \phi_{xz} - 2\phi_y \phi_z \phi_{yz}}{|\nabla \phi|^3} \quad (3.15)$$

Die Zugehörigkeit eines Punktes \vec{x} zur jeweiligen Teilmenge entspricht einer Klassifikation. Dazu wird für ein Bild der Dimension \mathbb{R}^n eine Level-Set-Funktion ϕ der gleichen Dimension erstellt, welche dazu dient das Bild pixel- bzw. voxelweise zu klassifizieren und damit in zwei Bereiche zu segmentieren.

In der Regel wird man aber keine Funktion wie beispielsweise die in Gleichung (3.3) finden, so dass die Segmentierung eines Bildes als zufriedenstellend gelöst betrachtet werden kann. Daher muss die Level-Set-Funktion entsprechend deformiert werden, was mit Hilfe der partiellen Differentialgleichungen geschieht. Diesen Prozess beschreiben die Level-Set-Methoden.

3.2 Level-Set Methoden

Für die Beschreibung der zeitlichen Entwicklung einer Oberfläche bieten sich partielle Differentialgleichungen an.

3.2.1 Entwicklung durch Advektion

Angenommen, man kennt für jeden Punkt auf der Kontur die Bewegungsrichtung und -geschwindigkeit, welche durch ein Vektorfeld $\vec{V}(\vec{x})$ gegeben ist, so kann für jeden Punkt die Position zum späteren Zeitpunkt mit Hilfe der *Lagrange-Formulierung* bestimmt werden:

$$\frac{\partial \vec{x}}{\partial t} = \vec{V}(\vec{x}) \quad (3.16)$$

Diese Variante der Oberflächen-Deformation wird beim so genannten *Front-Tracking* verwendet. Im Gegensatz zu den Level-Set-Methoden weist es allerdings Nachteile auf, denn für konkrete Berechnungen wird eine Diskretisierung der Kontur vorausgesetzt, die im Laufe der Entwicklung aktualisiert und auf Topologie-Wechsel überwacht werden muss. Diese mitunter aufwändigen und heuristischen Algorithmen kann man umgehen, wenn man die Level-Set-Funktionen aus dem vorangegangenen Abschnitt nutzt. Sie dienen damit nicht nur zur reinen Darstellung der Kontur, sondern gleichzeitig bietet sie die Handhabe für deren zeitliche Entwicklung. Dazu benutzt man die *Eulersche-Formulierung*:

$$\frac{\partial \phi}{\partial t} + \alpha \vec{V} \cdot \nabla \phi = 0 \quad (3.17)$$

Bedingt durch das Skalarprodukt in dieser Gleichung ist nur der zur Kontur orthogonale Anteil von \vec{V} für die Entwicklung relevant. Der konstante, skalare Faktor α wichtet die Stärke des Einflusses des Vektorfeldes.

Differentialgleichung (3.17) beschreibt ein Anfangswertproblem. Im Gegensatz hierzu steht die Beschreibung der Oberflächenentwicklung als Randwertproblem, die zu den so genannten *Fast-Marching-Methods* führt [Set99]. Bei diesen besteht die Idee darin, eine Ankunftszeit der Oberfläche für alle Punkte auf einem diskretisierten Gitter zu berechnen. Hieraus lässt sich direkt ableiten, dass diese Methode nur eine einzige Entwicklungsrichtung der Oberfläche zulässt, da sonst mehr als eine Ankunftszeit zulässig sein müsste. Die Fast-Marching-Methods können die Konturevolution deutlich beschleunigen, werden hier aber nur der Vollständigkeit halber erwähnt.

3.2.2 Entwicklung durch Krümmung

Die Entwicklung der Level-Set-Funktion durch ein externes Vektorfeld kann zu beliebig starken Krümmungen der Kontur führen, die aber nicht immer gewünscht sind. Aus diesem Grund ist eine Regularisierung notwendig. Dazu muss auf die

Eigenschaften der Level-Set-Funktion selbst zurück gegriffen werden, welche dann die Entwicklung der Funktion bestimmen.

Damit die Entwicklung durch Krümmung besser verstanden werden kann, werden hier zunächst die Begriffe der *Konvexität* und der *Konkavität* einer Kontur erklärt. Legt man für die Kontur die Durchlaufrichtung im mathematisch positiven Sinne fest, so ist die Kontur in einer Linkskurve konvex und in einer Rechtskurve konkav.

Bringt man diese Überlegungen nun zusammen mit der Definition der Krümmung, welche ja die Divergenz der Einheitsnormalen der Kontur ist, so laufen bei einer Linkskurve die nach außen gerichteten Normalen auseinander, was der Quelle eines Vektorfeldes entspricht, so dass die Divergenz hier positiv ist. Bei einer Rechtskurve verhält es sich genau umgekehrt, hier laufen die Normalen zusammen. Dies entspricht einer Senke, die Divergenz an diesen Stellen ist negativ. Diese Betrachtungen lassen sich wie folgt zusammenfassen:

- *Konvexität:*
 - Linkskurve bei Durchlauf im mathematisch positiven Sinn,
 - Quelle des Normalenfeldes,
 - $\kappa > 0$.
- *Konkavität:*
 - Rechtskurve bei Durchlauf im mathematisch positiven Sinn,
 - Senke des Normalenfeldes,
 - $\kappa < 0$.

Formal kann man diesen Zusammenhang wie folgt beschreiben:

$$\frac{\partial \phi}{\partial t} = \gamma \kappa |\nabla \phi| \quad (3.18)$$

Dabei ist γ ein konstanter, skalarer Faktor. Bei der Verwendung der vorzeichenbehafteten Distanzfunktion als Level-Set-Funktion gelten die in Abschnitt 3.1.2 beschriebenen Vereinfachungen.

Eine iterative Entwicklung einer Kontur, welche allein durch die Krümmung bestimmt ist, zeigt Abbildung 3.4. Man erkennt deutlich, dass unterschiedliche Bereiche abhängig von ihrer Krümmung geglättet werden, so dass nach etwa $n_I = 1280$ Iterationen die Kontur eine elliptische Form annimmt. Zur Glättung wird diese Kontur in konvexen Bereichen nach innen gedrückt, indem die Level-Set-Funktion hier in ihrer Wertigkeit erhöht wird ($\kappa > 0$). In konkaven Bereichen verhält es sich genau umgekehrt.

Abbildung 3.5 zeigt einen Ausschnitt der in Abbildung 3.4 entwickelten Kontur. Für diese Darstellung wurde die Krümmung der Level-Set-Funktion berechnet und an den Positionen der Startkontur dargestellt.

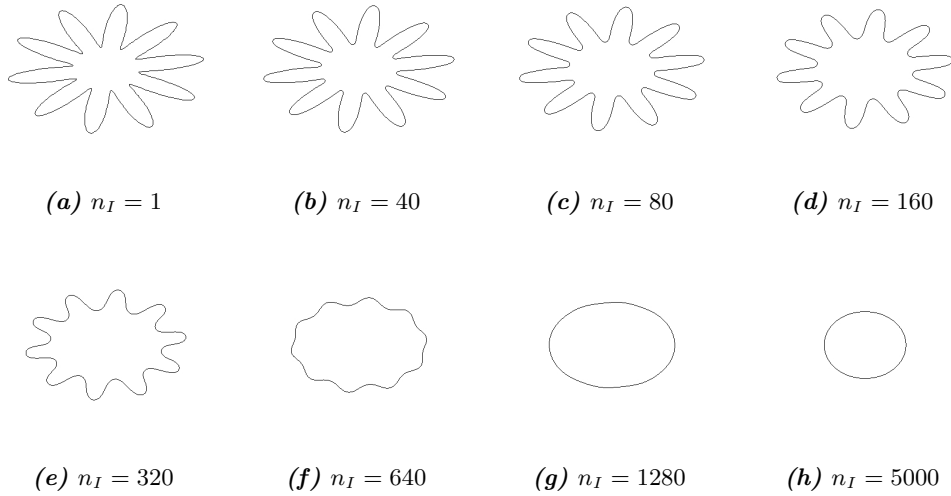


Abbildung 3.4: Iterative Veränderung der Kontur nach n_I Iterationen, wenn ausschließlich die Krümmung die Entwicklung bestimmt.

Eine weitere Konsequenz der Betrachtung der Vorzeichen der Krümmung ist der Übergang von $n_I = 1280$ zu $n_I = 5000$ Iterationen in Abbildung 3.4. Man stellt fest, dass die Kontur deutlich kleiner wird und immer mehr die Form eines Kreises annimmt. Aus dieser Beobachtung lässt sich folgern, dass jede beliebige Kontur zu einem einzigen Punkt zusammen läuft, solange deren Entwicklung lediglich durch die Krümmung bestimmt wird [Set99, Osh02]. Um diesem Effekt entgegen zu wirken, werden so genannte Ballonkräfte eingesetzt, die dann der verkleinernden Tendenz entgegenwirken.

3.2.3 Entwicklung durch Propagation

Die im vorherigen Abschnitt angesprochene Ballonkraft muss die Kontur an jeder Stelle orthogonal zu sich selbst nach außen drücken, was äquivalent zur Verringerung der Wertigkeit der Level-Set-Funktion ist. Dies ist ein Spezialfall, der formal schon in der mathematischen Beschreibung der Entwicklung durch Advektion in Abschnitt 3.2.1 enthalten ist.

Um eine orthogonale Bewegung zu erreichen, dürfen die Elemente des Vektorfeldes $\vec{V}(\vec{x})$ nur in die zur Kontur normale Richtung $\nabla\phi$ zeigen.

$$\vec{V} = \beta \frac{\nabla\phi}{|\nabla\phi|} \quad (3.19)$$

Dabei ist β eine skalare Konstante. Setzt man dieses Vektorfeld nun in (3.17) ein,

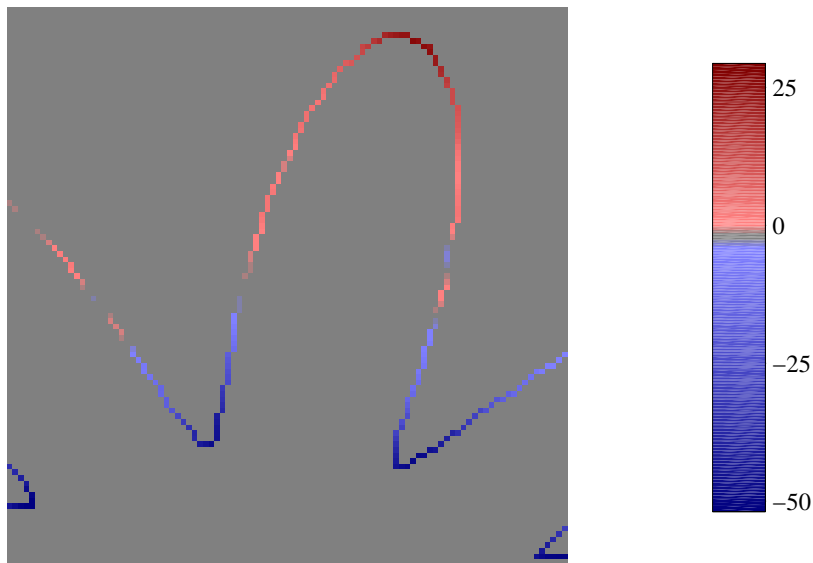


Abbildung 3.5: Visualisierung der Krümmung einer künstlichen Kontur. Negative Werte (blau) beschreiben konkave Kurvenabschnitte, positive Werte (rot) hingegen konvexe.

so erhält man:

$$\begin{aligned}
 \frac{\partial \phi}{\partial t} + \vec{V} \cdot \nabla \phi &= 0 \\
 \frac{\partial \phi}{\partial t} + \beta \frac{\nabla \phi}{|\nabla \phi|} \cdot \nabla \phi &= 0 \\
 \frac{\partial \phi}{\partial t} + \beta |\nabla \phi| &= 0
 \end{aligned} \tag{3.20}$$

Hier wird demnach, genau wie bei der Entwicklung durch Krümmung (Abschnitt 3.2.2), aus den Eigenschaften der Level-Set-Funktion selbst die zeitliche Entwicklung berechnet.

Eine erhöhte Flexibilität erreicht man, wenn man die Entwicklung in Normalenrichtung nicht allein über die Konstante β steuert, sondern zusätzlich einen räumlich abhängigen Faktor hinzufügt, die skalarwertige Geschwindigkeitsfunktion $S(\vec{x})$.

$$\frac{\partial \phi}{\partial t} + \beta S(\vec{x}) |\nabla \phi| = 0 \tag{3.21}$$

Im weiteren Verlauf wird die Geschwindigkeitsfunktion $S(\vec{x})$ noch eine zentrale Rolle einnehmen und genau wie in [Zhu06] und [Pal08] zusätzlich in zeitlicher Richtung variiert werden.

3.2.4 Superposition von Entwicklungstermen

Die bisher dargestellten Entwicklungen der Level-Set-Funktion mit Hilfe der Level-Set-Methoden bedingten die Entwicklung der Kontur aufgrund unterschiedlicher Eigenschaften. Nun werden die jeweiligen Terme aus (3.17), (3.18) und (3.21), welche die Dynamik der sich entwickelnden Kontur beschreiben, durch Superposition additiv in einer partiellen Differentialgleichung zusammengefasst.

$$\frac{\partial \phi}{\partial t} + \alpha \vec{V} \cdot \nabla \phi + \beta S(\vec{x}) |\nabla \phi| = \gamma \kappa |\nabla \phi| \quad (3.22)$$

Diese Gleichung ist eine allgemeine Formulierung, mit der viele Probleme der Oberflächenentwicklung beschrieben werden können. Die Gewichtung der Eigenschaften der Summanden übernehmen die Faktoren α , β und γ .

3.3 Diskretisierung

Bisher hatte die Level-Set-Funktion ϕ einen Definitionsbereich auf den reellen Zahlen $\Omega \in \mathbb{R}^n$. Dies ist für die praktische Anwendung der Segmentierung nicht realisierbar. Aus diesem Grund wird hier eine diskrete Version der Level-Set-Funktion definiert und mit $\Phi(\vec{k})$ bezeichnet, wobei die Elemente des Vektors \vec{k} nur ganze Zahlenwerte annehmen dürfen ($k_i \in \mathbb{Z}$). Der Wertebereich ist analog $\Omega \in \mathbb{Z}^n$.

Dieser Übergang kann als Abtastung der Level-Set-Funktion ϕ mit einem Abtast-raster von h_x , h_y und h_z aufgefasst werden:

$$\Phi(k_x, k_y, k_z) = \phi(k_x h_x, k_y h_y, k_z h_z) \quad (3.23)$$

Auf diskreten Daten muss darüber hinaus eine Approximation der partiellen Ableitungen $\frac{\partial \phi}{\partial x}$ vorgenommen werden. Sei hierzu $\Phi(k)$ zunächst eine eindimensionale Funktion, dann ist die Vorwärtsdifferenz in x -Richtung D^{+x} gegeben durch:

$$D^{+x} \Phi(k_x) = \frac{\Phi(k_x + 1) - \Phi(k_x)}{h_x} \quad (3.24)$$

Diese berechnet die diskrete Ableitung für die Zwischengitterposition $k_x + \frac{1}{2}$. Die Rückwärtsdifferenz D^{-x} folgt analog und berechnet die Ableitung auch für eine Zwischengitterposition $k_x - \frac{1}{2}$:

$$D^{-x} \Phi(k_x) = \frac{\Phi(k_x) - \Phi(k_x - 1)}{h_x} \quad (3.25)$$

Für eine Approximation auf dem Abtastraster wird auf die zentrale Differenz D^{0x} zurückgegriffen:

$$D^{0x} \Phi(k_x) = \frac{\Phi(k_x + 1) - \Phi(k_x - 1)}{2h_x} \quad (3.26)$$

Dabei beschreibt h_x den Abstand zwischen zwei Abtastpunkten in x -Richtung. Für die zweite Ableitung, welche für die Berechnung der Krümmung herangezogen werden muss, wird eine Kombination aus Vorwärts- und Rückwärtsdifferenz berechnet.

3.3.1 Upwind-Schema

Die Lösung der Differentialgleichung (3.22) durch das einfache Vorwärts-Finite-Differenzen-Schema führt für die Lösung der Entwicklung durch Propagation (siehe Abschnitt 3.2.3) zu keiner stabilen Lösung [Whi98], insbesondere wenn die Level-Set-Funktion nicht die Form einer Distanzfunktion besitzt. Aus diesem Grund wurde für die Lösung von Osher et al. [Osh88] das Upwind-Schema vorgeschlagen.

Dieses Schema verfolgt den Ansatz, dass entsprechend der Bewegungsrichtung der Kontur entschieden wird, ob die Vorwärts- oder die Rückwärtsdifferenz für die Berechnung herangezogen wird. Das erhöht die Stabilität dieses numerischen Verfahrens.

Der zweite Term in (3.21) lässt sich dann mit dem Upwind-Schema analog zu [Whi98] anwenden:

$$\begin{aligned} \beta S(\vec{x}) |\nabla \phi| \approx & \left(\left(\max(\beta S(\vec{k}), 0) D^{-x} \Phi(\vec{k}) + \min(\beta S(\vec{k}), 0) D^{+x} \Phi(\vec{k}) \right)^2 \right. \\ & + \left(\max(\beta S(\vec{k}), 0) D^{-y} \Phi(\vec{k}) + \min(\beta S(\vec{k}), 0) D^{+y} \Phi(\vec{k}) \right)^2 \\ & \left. + \left(\max(\beta S(\vec{k}), 0) D^{-z} \Phi(\vec{k}) + \min(\beta S(\vec{k}), 0) D^{+z} \Phi(\vec{k}) \right)^2 \right)^{\frac{1}{2}} \end{aligned} \quad (3.27)$$

Im Anhang B.1 ist exemplarisch dargestellt, inwiefern dieses Schema zur Stabilität der Lösung beitragen kann.

Beim Upwind-Schema ist Limitierung der Zeitschrittgröße aus Stabilitätsgründen dennoch von großer Bedeutung. Der Punkt, welcher sich mit der höchsten Geschwindigkeit voran bewegt, bestimmt die Zeitschrittgröße Δt :

$$\Delta t < \frac{h_x}{\sup_{k \in \Omega} (|\beta S(\vec{k})|)} \quad (3.28)$$

Dieser Ausdruck beschreibt die Courant-Friedrichs-Lewy Bedingung (CFL) [Osh02].

Die Entwicklung durch Krümmung gemäß (3.18) beschreibt einen Diffusionsterm und kann mit einem einfachen Vorwärts-Differenzschema gelöst werden [Whi98], d.h. für eine stabile Entwicklung muss in einem solchen Fall nicht auf das Upwind-Schema zurückgegriffen werden.

3.3.2 Narrow-Band-Level-Sets

Die Level-Set-Methoden, wie sie in den vorangegangenen Abschnitten beschrieben worden sind, definieren mittels einer partiellen Differentialgleichung die zeitliche Veränderung der Level-Set-Funktion und damit die Entwicklung der hierdurch beschriebenen Kontur. Diese Entwicklung wird auf dem gesamten Definitionsbereich durchgeführt, obwohl nur die Werte in direkter Nähe zur Kontur von Interesse sind.

Das stellt einen enormen, überflüssigen Rechenaufwand dar, wenn nur ein einziges Level-Set für die betreffende Applikation von Bedeutung ist. Durch den Einsatz der Narrow-Band-Level-Sets kann der Rechenaufwand deutlich reduziert werden, indem nur die Werte der Funktion in der Nähe des Zero-Level-Sets von einer Iteration zur nächsten erneuert werden. Außerhalb dieser bezüglich ihres Radius definierten Umgebung werden die Werte eingefroren.

Bewegt sich die Kontur zu dicht an den Rand der interessierenden Umgebung muss eine Reinitialisierung vorgenommen werden, bei der die Umgebung neu definiert wird und die Werte der Level-Set-Funktion durch vorzeichenbehaftete Distanzwerte ersetzt werden. Die Größe der Umgebung stellt folglich einen Kompromiss zwischen dem Rechenaufwand bei einem Iterationsschritt und der Häufigkeit der Reinitialisierungen dar.

Ein weiterer vorteilhafter Aspekt bezieht sich auf das Stabilitätskriterium der numerischen Berechnungen (3.28). Wird der gesamte Definitionsbereich in einer Iteration aktualisiert, so muss die CFL-Bedingung (3.28) für alle Werte gelten. Der Bereich, der das potentielle Maximum der Geschwindigkeitsfunktion beinhaltet ist damit deutlich größer, als wenn nur ein schmales Band um die interessierende Kontur betrachtet wird. Weitere Details zu diesem Ansatz finden sich beispielsweise in [Osh02].

3.3.3 Sparse-Field-Level-Sets

Die Sparse-Field-Methode kann als eine Weiterentwicklung der Narrow-Band-Level-Sets betrachtet werden. Während bei der Narrow-Band-Methode ein Band mit einer Breite von etwa sechs bis zwölf Gitterzellen [Ada95] um das Zero-Level-Set zeitlich entwickelt wird, ist bei dem Sparse-Field-Ansatz das Band nur eine Gitterzelle breit. Würde man diese geringe Breite bei der Narrow-Band-Methode wählen, folgte daraus direkt die Notwendigkeit der Reinitialisierung bei jedem Iterationsschritt, da die Kontur bei jeder Bewegung das Band an mindestens einer Stelle verlassen würde. Dieses Problem wird bei dem Sparse-Field-Ansatz umgangen, indem die Distanzfunktion bei jeder Iteration neu approximiert wird.

Die Approximation der vorzeichenbehafteten Distanzfunktion erfolgt nur an den Stellen, die bei der bevorstehenden Iteration auch wieder für weitere Berechnungen (z. B. Krümmung) benötigt werden. Auf diese Weise werden unnötige Berechnungen direkt vermieden.

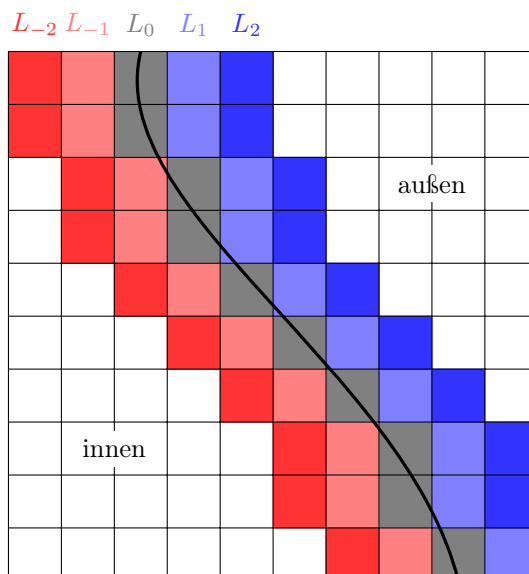


Abbildung 3.6: Die Mengen der Sparse-Field-Methode in Relation zur Kontur. Die aktive Menge L_0 wird auf der Innenseite flankiert von dem L_{-1} und dem L_{-2} Set, auf der Außenseite von den Sets L_1 und L_2 .

Für die Realisierung dieses Vorteils der Sparse-Field-Level-Sets wird eine *aktive Menge* L_0 definiert. Diese ist dadurch gekennzeichnet, dass die Werte der Level-Set-Funktion in dem Intervall $[-\frac{1}{2}, \frac{1}{2}[$ liegen:

$$L_0 = \left\{ \vec{k} \mid -\frac{1}{2} \leq \Phi(\vec{k}) < \frac{1}{2} \right\} \quad (3.29)$$

Eine Kontur mit der zugehörigen aktiven Menge (grau) ist in Abbildung 3.6 dargestellt. Nur die Punkte, die zu der aktiven Menge gehören, werden bei einer Iteration durch die Level-Set-Methode aktualisiert.

An die aktive Menge grenzt nach innen die Mengen L_{-1} , bei der die Werte der Level-Set-Funktion in einem Intervall von $[-\frac{3}{2}, -\frac{1}{2}[$ liegen. Eine allgemeine Definition der Mengen L_i lautet damit:

$$L_i = \left\{ \vec{k} \mid i - \frac{1}{2} \leq \Phi(\vec{k}) < i + \frac{1}{2} \right\} \quad (3.30)$$

mit $-P \leq i \leq P$ und $P, i \in \mathbb{Z}$. P beschreibt hierbei die Anzahl der Mengen um die aktive Menge.

Die Menge L_0 ist schichtweise von so vielen angrenzenden Mengen umgeben, wie für die späteren Berechnungen benötigt werden. Beispielsweise benötigt man für die Berechnung der zweiten Ableitung einen 3×3 Kernel, so dass nach außen und nach innen zwei Schichten benötigt werden. Das trifft insbesondere in konvexen und konkaven Bereichen der Kontur zu.

Die Anzahl der benötigten Mengen ergibt sich aus der 1-Norm, auch bekannt unter dem Namen Manhattan-Distanz, der Länge vom Mittelpunkt eines Kerns zu seinem entferntesten Eckpunkt. Seien \vec{p}_i die Vektoren, welche vom Mittelpunkt auf die Elemente des Kerns zeigen, so gilt für P :

$$P = \max(\|\vec{p}_i\|_1) \quad (3.31)$$

Alle Mengen L_i mit $i \neq 0$ übernehmen die Aufgabe der approximierten, vorzeichenbehafteten Distanzfunktion. Dazu wird von den direkt an die aktive Menge L_0 angrenzenden Mengen ausgehend, zu jedem Element der Nachbar mit dem absolut minimalen Wert herangezogen und dieser um eins verringert bzw. vergrößert. Im Innern der Kontur erfolgt eine Verringerung, außen eine Erhöhung. Dieses Vorgehen wird mit den schichtweise angrenzenden, weiter außen liegenden Mengen wiederholt, bis alle Werte aktualisiert worden sind.

Die Implementation geschieht in der Regel mit Hilfe von verketteten Listen [Whi98], mit denen dynamisch die Fälle behandelt werden können, in denen ein Punkt die aktive Menge verlässt und in eine angrenzende Menge übergeht, oder auch Punkte neu zu einer Menge hinzu genommen werden.

Genau auf die Weise, wie schon die Narrow-Band-Methode die Menge der entwickelten Punkte reduziert hat und damit die Zeitschrittgröße potentiell vergrößert hat, bringt hier die Sparse-Field-Methode nochmals einen zusätzlichen Gewinn.

3.3.4 Allgemeine Level-Set-Parameter

Unabhängig von der Wahl der Funktionen, welche die Advektion bzw. Propagation der Kontur steuern, liegen dem iterativ ablaufenden Prozess gemeinsame Parameter zu Grunde. Die erste Klasse von Parametern gewichtet den jeweiligen Einfluss von Advektion, Propagation und Krümmung relativ zueinander (3.22), die zweite Klasse beschreibt ein Konvergenz- bzw. Abbruchkriterium für den Entwicklungsprozess.

Das Abbruchkriterium ist die Unterschreitung einer festgelegten, mittleren quadratischen Änderung RMS_{\max} (root-mean-square) der Level-Set-Funktion von einer Iteration zur nächsten. Konvergiert der Konturevolutionsprozess gegen eine stabile Lösung, sinkt der so genannte RMS-Wert entsprechend:

$$\text{RMS}(n_I) = \sqrt{\frac{1}{m} \sum_{\vec{k}} \left(\Phi^{n_I+1}(\vec{k}) - \Phi^{n_I}(\vec{k}) \right)^2} \quad (3.32)$$

Da die Konvergenz einer Lösung jedoch nicht immer gegeben ist, wird zusätzlich die maximale Anzahl von durchzuführenden Iterationen n_{\max} festgelegt, bei deren Überschreitung auch ohne das Erreichen des RMS-Schwellenwertes der Konturevolutionsprozess abgebrochen wird.

Folglich stehen bei jeder Level-Set-Entwicklung folgende Parameter zur Verfügung:

- Advektionsgewicht α : Gewichtet die Advektionsfunktion.
- Propagationsgewicht β : Gewichtet die Propagationsfunktion.
- Krümmungsgewicht γ : Die Erhöhung von γ resultiert in einer glatteren Kontur.
- Konvergenzkriterium RMS_{\max} : Entscheidet, wann eine Lösung als konvergiert zu betrachten ist.
- Abbruchkriterium n_{\max} : Bestimmt die maximale Anzahl von durchzuführenden Iterationen – ohne Berücksichtigung des Konvergenzkriteriums.

3.4 Insight Segmentation and Registration Toolkit (ITK)

Das *Insight Segmentation and Registration Toolkit* ist ein frei verfügbares Open-Source Softwaresystem, welches in der Programmiersprache C++ objektorientiert implementiert ist. Es bietet umfangreiche Möglichkeiten zur Bildverarbeitung, insbesondere zur Segmentierung und Registrierung. Nicht nur viele Algorithmen liegen bereits implementiert vor, auch die freie Erweiterbarkeit dank dokumentierter Schnittstellen ist gegeben. Das von C++ unterstützte Template-Konzept wird im Rahmen der generischen Programmierung ausgenutzt.

Ein besonderes Augenmerk bei der Entwicklung von ITK wurde auf Plattformunabhängigkeit gelegt. Hierbei spielt *CMake* eine zentrale Rolle, welches den Kompilervorgang für verschiedenste Compiler aus einheitlichen Konfigurationsdateien regelt.

Die in den vorherigen Abschnitten erläuterte Differentialgleichung (3.22) ist in verallgemeinerter Form im Insight Segmentation and Registration Toolkit (ITK) implementiert. Konsequenterweise ist hier zusätzlich noch die Krümmung mit einer räumlichen Gewichtung versehen worden, ähnlich wie auch die Ballonkraft in Abschnitt 3.2.3 ortsabhängig gemacht wurde. Da die räumlich abhängige Krümmung in der Implementation jedoch keine weitere Rolle spielen wird, ist diese Komponente in (3.22) weggelassen worden.

Für eine effiziente Lösung der partiellen Differentialgleichung wird in ITK die Sparse-Field Methode (siehe Abschnitt 3.3.3) bereitgestellt und im Rahmen dieser Arbeit verwendet. Eine Basisklasse für eine lokal adaptive Behandlung der Geschwindigkeitsfunktion ist in der Arbeitsgruppe für Multimodale Bildverarbeitung des Instituts für Neurowissenschaften und Biophysik (INB-3) des Forschungszentrums Jülich vorhanden und kann genutzt werden.

Kapitel 4

Algorithmenanpassung

Dieses Kapitel beschreibt den modellbasierten Level-Set-Segmentierer, dessen Erweiterung zu einem echt-dreidimensionalen Ansatz sowie Adaptionen der Geschwindigkeitsfunktion und des darin einfließenden Signals.

4.1 Der modellbasierte Level-Set-Segmentierer

Der modellbasierte Level-Set-Segmentierer wurde von Zhuang et al. in [Zhu06] vorgestellt und ist, wie bereits in Abschnitt 2.3.1 angesprochen wurde, eine Portierung der Formkräfte des Brain-Extraction Tools in die Geschwindigkeitsfunktion der Level-Set-Methoden. Da die Level-Set-Methoden jedoch nur für zweidimensionale Daten implementiert worden sind, wurde in [Zhu06] für dreidimensionale Daten ein Pseudo 3D-Ansatz vorgeschlagen.

4.1.1 Pseudo 3D-Ansatz

Obwohl Volumendaten vorliegen, wird eine zweidimensionale, schichtweise Segmentierung vorgenommen. In einer zentralen (bei Menschenhirn-MRTs sagittalen) Schicht wird eine kreisrunde Startkontur gewählt, welche innerhalb des zu segmentierenden Volumens liegt. Von hier ausgehend werden die Level-Set-Methoden verwendet, um die Startkontur zu deformieren. Konvergiert diese Evolution und erfüllt damit ein zuvor definiertes Abbruchkriterium, wird zur benachbarten Schicht übergegangen.

Die Startkontur einer benachbarten Schicht ergibt sich als verkleinerte Version des Segmentierungsergebnisses der direkt zuvor prozessierten Schicht. Diese verkleinerte Kontur ist aus der Level-Set-Funktion leicht zu bestimmen, indem nicht das Zero-Level-Set, sondern eine Höhenlinie mit $k < 0$ gewählt wird (vgl. Abb. 3.1 bzw. (3.1)). Dieser Vorgang kann als *Konturpropagation* bezeichnet werden. Die Größe von k , und damit der Grad der Verkleinerung, ist abhängig von der Voxeldimension der Volumendaten in der propagierten Richtung.

Darüber hinaus wurde eine *Leakage-Detektion* vorgestellt, die ebenfalls auf der Annahme beruht, dass benachbarte Schichten eine hohe Ähnlichkeit aufweisen.

In den Fällen, in denen die überlappenden Bereiche zu klein sind, wird das Segmentierungsergebnis verworfen, der Krümmungsterm um einen zuvor definierten Faktor erhöht und erneut eine Segmentierung durchgeführt. Dieser Vorgang wird wiederholt, bis eine hinreichend starke Überlappung erreicht wird. In den Randschichten wird dieses Vorgehen jedoch nicht angewendet, da hier die Änderung der Größe des zu segmentierenden Gehirngewebes in Richtung der weiter außen liegenden Schichten schon aus geometrischen Gründen größer ausfällt als in zentralen Gehirnregionen.

4.1.2 Geschwindigkeitsfunktion

Für die Segmentierung durch die Level-Set-Methoden werden von der allgemeinen Geschwindigkeitsfunktion (3.22) lediglich der regularisierende Krümmungsterm (Abschnitt 3.2.2) und der Propagationsterm (Abschnitt 3.2.3) benötigt. Beide bewirken die Deformation der Kontur in dessen Normalenrichtung, so dass es eine natürliche Folgerung ist, auch die Bildeigenschaften auf der Normalen für die Bestimmung der Geschwindigkeitsfunktion heranzuziehen.

Die Geschwindigkeitsfunktion $S(\vec{x}, \vec{N}, I)$ an der Stelle $\vec{x} \in \partial\Omega$ ergibt sich lokal und zeitlich adaptiv aus dem minimalen und maximalen Intensitätswert I_{\min} und I_{\max} der Bilddaten auf der nach innen zeigenden Normalen $-\vec{N}$.

$$I_{\min}(\vec{x}, \vec{N}) = \min_{0 \leq d \leq d_m} (I(\vec{x} - d\vec{N})) \quad (4.1)$$

$$I_{\max}(\vec{x}, \vec{N}) = \max_{0 \leq d \leq d_M} (I(\vec{x} - d\vec{N})) \quad (4.2)$$

Die Robustheit dieser lokalen Extremwerte soll im Folgenden erhöht werden. Dazu wurde von Smith in [Smi02] eine Histogrammauswertung der Bilddaten vorgenommen, welche ebenfalls bei Zhuang Anwendung findet [Zhu06]. Sei dazu $h_c(I)$ das kumulierte Histogramm der relativen Häufigkeiten, dann werden hierin die 2%- und die 98%-Schwelle bestimmt und mit $h_{2\%}$ bzw. $h_{98\%}$ bezeichnet:

$$h_{2\%} = \max_{h_c(I) \leq 0,02} (I) \quad (4.3)$$

$$h_{98\%} = \max_{h_c(I) \leq 0,98} (I) \quad (4.4)$$

$$(4.5)$$

Zusätzlich wird innerhalb der Startkontur der Median der Helligkeitswerte ermittelt. Sei dazu $h_{c,0}$ das kumulierte Histogramm der Bildintensitäten innerhalb des Zero-Level-Sets, dann ergibt sich der Medianwert für diese Startkontur gemäß:

$$h_{\text{med}} = \max_{h_{c,0}(I) \leq 0,5} (I) \quad (4.6)$$

Dieser Medianwert h_{med} fließt in ein robustes Intensitätsminimum $I_{r,\min}$ und Intensitätsmaximum $I_{r,\max}$ ein. Dabei darf $I_{r,\min}$ nicht größer werden als h_{med} und

nicht kleiner als $h_{2\%}$ und $I_{r,\max}$ nicht kleiner als h_{med} und nicht größer als $h_{98\%}$:

$$I_{r,\min} = \min(\max(I_{\min}, h_{2\%}), h_{\text{med}}) \quad (4.7)$$

$$I_{r,\max} = \max(\min(I_{\max}, h_{98\%}), h_{\text{med}}) \quad (4.8)$$

$I_{r,\min}$ und $I_{r,\max}$ bestimmen lokal die Wertigkeit der Geschwindigkeitsfunktion:

$$S(\vec{x}, \vec{N}, I) = T - \frac{I_{r,\min}(\vec{x}, \vec{N}) - h_{2\%}}{I_{r,\max}(\vec{x}, \vec{N}) - h_{2\%}} \quad (4.9)$$

Der Quotient in vorstehender Gleichung wird durch das Kontrastverhältnis auf der Normalen bestimmt. Wenn dieser den vorgegebenen Zielwert von T überschreitet, wechselt das Vorzeichen der Geschwindigkeitsfunktion und die Kontur bewegt sich in die entgegengesetzte Richtung. Bei kleinen, lokalen Kontrasten, wie es im Innern des zu segmentierenden Bereiches der Fall ist, ist $I_{\min} \approx I_{\max}$ und damit der rechte Summand der Geschwindigkeitsfunktion etwa gleich Eins. Bei einem typischen Wert für $T = 0,5$ ist die Geschwindigkeitsfunktion dann negativ und die Kontur bewegt sich nach außen.

4.1.3 Parameter

Um das Verhalten des Segmentierers zu bestimmen stehen neben den allgemeinen Level-Set-Parametern (Abschnitt 3.3.4) bei dem modellbasierten Level-Set-Segmentierer folgende Parameter zur Verfügung:

- Kontrastabhängiger Zielwert T : Je kleiner dieser Wert ist, desto höher muss der lokale Kontrast sein, damit die Geschwindigkeitsfunktion ihr Vorzeichen wechselt.
- Längen für Minimums- und Maximumssuche d_m und d_M : Auf diesen Längen wird der lokale Kontrast ermittelt. Je größer diese Werte sind, desto globaler sind die Auswirkungen der Kontrastbestimmung.
- Relative Gewichtung zwischen Glättung γ und Propagation β : Je größer das Verhältnis $\frac{\gamma}{\beta}$ ist, desto glatter wird die Ergebniskontur.
- Position der Startkontur: Der hierin global bestimmte Median fließt durch in die lokale Auswertung der Geschwindigkeitsfunktion ein.

4.2 Modifikation des modellbasierten Level-Set-Segmentierers

Der vorgestellte modellbasierte Segmentierer weist trotz des vielversprechenden Ansatzes potentielle Schwachstellen auf:

- Die Schichtweise Segmentierung zeigt besonderes in Randschichten Ungenauigkeiten.

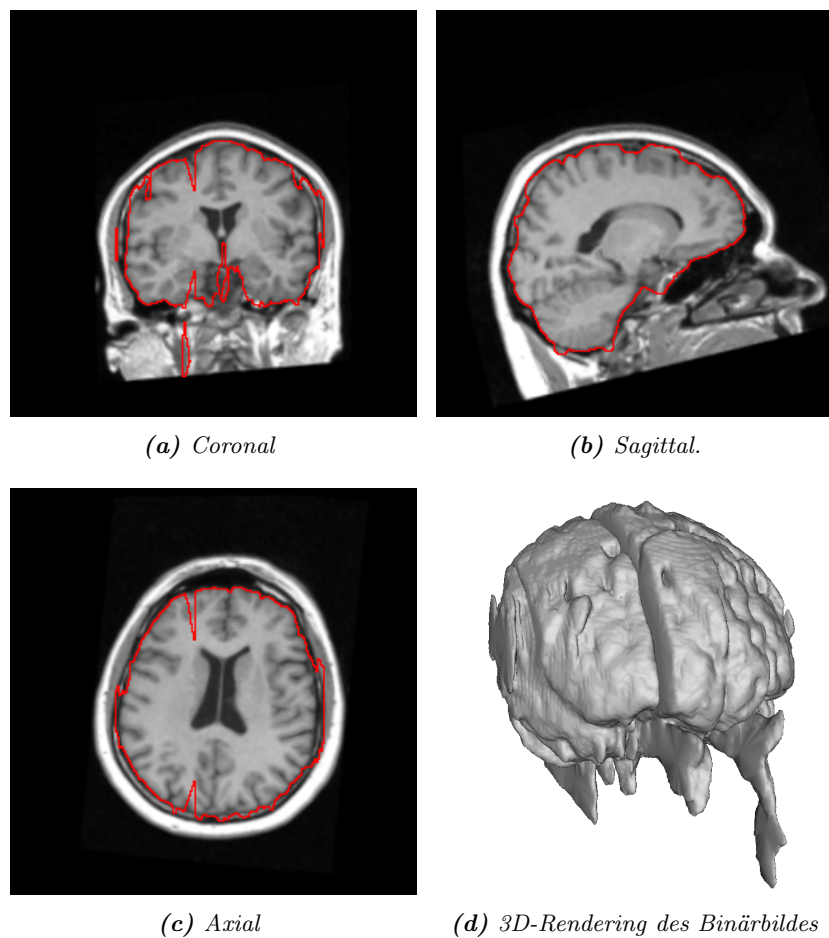


Abbildung 4.1: Segmentierungsergebnis einer Humanhirn Magnet-Resonanztomographie.

- Bei MRT-Daten mit ausgeprägten Helligkeitsgradienten wirkt sich die Histogrammauswertung innerhalb der Startkontur nachteilig aus und führt zu vorzeitigen Abbrüchen des Segmentierungsprozesses.
- Leakage-Vermeidung durch unterschiedliche Krümmungsgewichte in benachbarten Schichten führt zu Artefakten.

Abbildung 4.1 verdeutlicht die oben aufgeführten Probleme, die durch Anwendung des ursprünglichen Algorithmus auf ein Menschenhirn-MRT entstanden sind. Diese Schwachstellen können durch die in den folgenden Abschnitten vorgeschlagenen Modifikationen behoben werden.

4.2.1 Portierung in den 3-D Raum

Die Level-Set-Formulierung bietet direkt die Möglichkeit, eine Portierung in den dreidimensionalen Raum vorzunehmen – aus der zweidimensionalen Konturkurve wird dann eine dreidimensionale Konturoberfläche. Die verwendete Programmierbibliothek ITK unterstützt diese Möglichkeit durch Anwendung des generischen Programmierkonzeptes, bei dem Algorithmen so allgemein formuliert werden, dass sie mit unterschiedlichen Datentypen arbeiten können. Die Programmiersprache C++ unterstützt dieses Konzept in Form der Templates. Bilddaten können daher in ITK beliebige Dimensionen besitzen. Es wurde eine Implementation für zwei- und dreidimensionale Daten erstellt.

Die Geschwindigkeitsfunktion ist ebenfalls bereits so allgemein formuliert, dass ohne Einschränkung ein Übergang in den dreidimensionalen Raum möglich ist. Lediglich die Startkontur wird nicht mehr als kreisrunde Scheibe, sondern als zentral bzw. manuell positionierte Kugel mit frei wählbarem Radius definiert.

Auf die Möglichkeit der automatischen Leakage-Detektion, wie sie im ursprünglichen Verfahren eingesetzt wird, muss verzichtet werden, da Leakage in dreidimensionalen Daten nicht über die Ähnlichkeit zweier getrennt berechneter Segmentierungsergebnisse benachbarter Schichten detektiert werden kann. Aus der Verarbeitung dreidimensionaler Daten resultiert nur ein einziges Segmentierungsergebnis. Darüber hinaus zeigt die vorgeschlagene Leakage-Detektion nicht die gewünschte Leistungsfähigkeit (siehe Abbildung 4.1).

Ein nachteiliger Aspekt, der bei dieser Portierung auftritt, ist der erhöhte Rechenaufwand, der durch die Verwendung echt dreidimensionaler Daten entsteht. Zusätzlich wird jeder Messwert auf der Normalen mittels einer linearen Interpolation approximiert. Dies stellt einen Kompromiss zwischen Genauigkeit und Rechenaufwand dar. Die Sampleweite beträgt eine halbe Voxellänge.

4.2.2 Vorverarbeitung

Die Vorverarbeitung der Daten dient dazu, den Kernalgorithmus mit sinnvollen Daten zu versorgen. Diese werden in allen Raumrichtungen zu möglichst isotropen Daten interpoliert und am Rand fortgesetzt.

4.2.2.1 Interpolation zu isotropen Daten

Bei der Anwendung des Segmentierungsverfahrens auf Rattenhirne besteht das Problem der anisotropen Voxelauflösung. Beispielsweise ist für den Datensatz TN211 die Abtastung in Richtung der z -Achse um den Faktor drei geringer als in Richtung der x - und y -Achse (Abschnitt 5.1). Dieses so genannte *Spacing* ist eine zusätzliche Information, die in medizinischen Daten aber in der Regel verfügbar ist, da nur so das Resultat des bildgebenden Verfahrens mit der aufgenommenen, anatomischen Struktur zur Deckung gebracht werden kann.

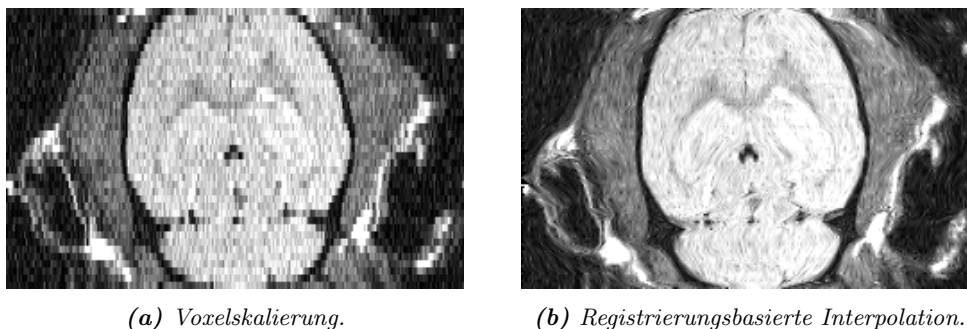


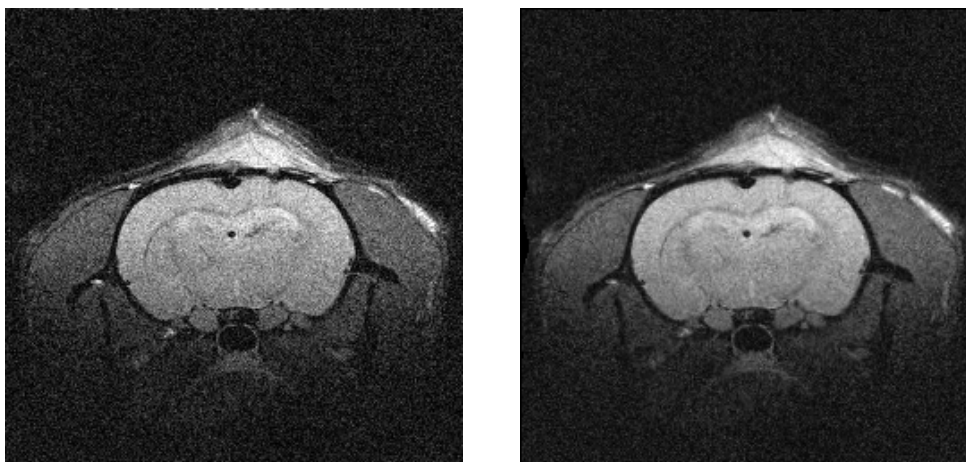
Abbildung 4.2: *Voxelskalierung im Vergleich zum Ergebnis der registrierungsbasierten Interpolation nach Penney [Pen04] in einer Ebene parallel zur Interpolationsrichtung.*

Die aus ITK verwendeten Level-Set-Basis-Algorithmen berücksichtigen das Spacing jedoch nicht [Iba03]. Das ist aber genau dann notwendig, wenn man davon ausgeht, dass die durch die Regularisierung angestrebte Glattheit der erwarteten Oberfläche nicht von der Koordinatenachse abhängig ist. Auch bei der Bestimmung der Geschwindigkeitsfunktion, insbesondere im Hinblick auf die für die lokale Kontrastbestimmung verwendeten Längen d_{\min} und d_{\max} , müsste eine entsprechende richtungsabhängige Samplingdistanz berücksichtigt werden.

Aus diesem Grund wurde eine registrierungsbasierte Interpolation eingesetzt, welche von Penney in [Pen04] vorgestellt wurde, um möglichst isotrope Daten zu erzeugen.

Die Registrierung liefert in zwei benachbarten Schichten des Eingangsbildmaterials zueinander korrespondierende Punkte \vec{p}_a und \vec{p}_b . Die Verbindungslinie zwischen diesen wird als Interpolationslinie \vec{L} bezeichnet. Für die Berechnung des Intensitätswertes eines Bildpunktes \vec{p}_i auf einer zu interpolierenden Schicht wird die Interpolationslinie mit dem geringsten Abstand zu \vec{p}_i gewählt, in diesen Punkt parallel verschoben und die Intensitätswerte der ursprünglichen Schichten an beiden Enden der verschobenen Interpolationslinie bestimmt. Der interpolierte Helligkeitswert an der Stelle \vec{p}_i ergibt sich aus der linearen Interpolation der Endpunktintensitäten. Dieser Schritt ist sehr rechenintensiv, so dass diese Vorverarbeitung komplett losgelöst von dem hier implementierten Algorithmus stattfindet.

Abbildung 4.2 zeigt einen horizontalen Schnitt aus einem interpolierten Datensatz. Zur Verdeutlichung der Leistungsfähigkeit dieser Interpolationsmethode ist sowohl das Ausgangsbildmaterial mit skalierten Voxeln als auch das Ergebnis der registrierungsbasierten Interpolation dargestellt. Ein wesentlicher Vorteil dieses Verfahrens ist der geringe Schärfeverlust auf interpolierten Ebenen (Abb. 4.3). Dies ist insbesondere bei dem kontrastbasierten Ansatz für die Geschwindigkeitsfunktion von Bedeutung.



(a) Schicht des Eingangsdatensatzes.

(b) Interpolierte Schicht.

Abbildung 4.3: Zwei benachbarte Schichten im interpolierten Datensatz. Zu beachten ist der geringe Schärfeverlust in der interpolierten Schicht.

4.2.2.2 Padding

Bilddaten werden natürlicherweise nur in einem räumlich begrenzten Bereich aufgenommen. Insbesondere die im Abschnitt 4.2.4 zustellende Signaladaption betrachtet die Daten auf der Normalen der entwickelten Kontur jedoch nicht nur nach innen, sondern auch nach außen. Damit diese Auswertung möglichst sinnvoll geschehen kann, wurden zwei Fortsetzungsmöglichkeiten implementiert – die Spiegelung am Rand und die Fortsetzung durch Wiederholung der Randpixel bzw. -voxel.

4.2.3 Modifikation der Geschwindigkeitsfunktion

Der kontrastbasierte Ansatz des Segmentierers von Zhuang soll vom Prinzip erhalten bleiben, aber insbesondere im Hinblick auf die genannten Schwachstellen verbessert werden. Einen bedeutenden Ansatzpunkt stellt hierbei die Geschwindigkeitsfunktion dar.

Die globale Histogrammauswertung innerhalb der Startkontur ist insbesondere bei einem dreidimensionalen Ansatz von Nachteil. Für das robuste Normalenmaximum $I_{r,\max}$ (4.8) wurde die Annahme gemacht, dass es nicht kleiner werden darf als der Median der Intensitätswerte innerhalb der Startkontur. Wird die Startkontur nun in einen hellen Bildbereich gelegt, ist auch der Wert des Medians entsprechend hoch.

In (4.9) führt das gemäß (4.7) und (4.8) zu einer Verzerrung der Kontrastverhältnisse in dunklen Bildbereichen – es wird der kontrastabhängige Zielwert auch

dann überschritten, wenn der lokale Kantenkontrast niedriger ist. Eine solche Abhängigkeit von der Wahl der Startkontur ist nicht wünschenswert.

Aus diesem Grund wird (4.9) so modifiziert, dass nur der lokale Kontrast in die Geschwindigkeitsfunktion einfließt. Der ermittelte Median für die Bestimmung der Werte $I_{r,\min}$ und $I_{r,\max}$ wird nun nicht mehr innerhalb der Startkontur bestimmt, sondern ebenfalls auf der nach innen gerichteten Normalen $-\vec{N}$. Damit ist dieser Wert lokal adaptiv und kann globale Helligkeitsgradienten besser berücksichtigen. Einen Einfluss hat der Median erst dann, wenn die Distanz, auf der dieser ermittelt wird, deutlich größer ist als die Längen der Minimums- und Maximumssuche.

Die 2%- und 98%-Histogrammschwellen, welche im kumulierten Histogramm des gesamten Datensatzes ermittelt werden, fließen nur bei der Bestimmung von $I_{l,\min}$ und $I_{l,\max}$ in die Geschwindigkeitsfunktion ein. Hierin werden folglich die ermittelten Intensitätsextrema $I_{r,\min}$ und $I_{r,\max}$ ausgetauscht und die 2% Schwelle entfernt:

$$S(\vec{x}, \vec{N}, I) = T - \frac{I_{l,\min}(\vec{x}, \vec{N})}{I_{l,\max}(\vec{x}, \vec{N})} \quad (4.10)$$

mit der zu (4.7) bzw. (4.8) ähnlichen Bedingung für $I_{l,\min}(\vec{x}, \vec{N})$ und $I_{l,\max}(\vec{x}, \vec{N})$ für robustere Werte. Sei hierzu $m(\vec{x}, \vec{N}, d_M)$ der Intensitätsmedian auf der Konturnormalen

$$m(\vec{x}, \vec{N}, d_M) = \text{Median} \left(I(\vec{x} - d\vec{N}) \right) \quad \text{mit } d = \{0, 1, 2, \dots, d_M\}, \quad (4.11)$$

dann ergibt sich für das robuste, lokale Intensitätsminimum und -maximum:

$$I_{l,\min} = \min \left(\max(I_{\min, h_{2\%}}, m(\vec{x}, \vec{N}, d_M)) \right) \quad (4.12)$$

$$I_{l,\max} = \max \left(\min(I_{\max, h_{98\%}}, m(\vec{x}, \vec{N}, d_M)) \right) \quad (4.13)$$

In Versuchen muss sich zeigen, ob diese lokale Adaption Vorteile gegenüber der globalen Medianauswertung zeigt.

4.2.4 Signaladaption

Basierend auf dem vorgeschlagenen, lokalen Auswertungsansatz der Geschwindigkeitsfunktion, kann das in diese Auswertung einfließende Signal angepasst werden. Eine alternative Herangehensweise wäre die lokale Adaption der Parameter der Geschwindigkeitsfunktion. Diese wurde in der vorliegenden Arbeit jedoch nicht verfolgt.

4.2.4.1 Konturnormalen-parallele Glättung

Bilder die einen globalen Helligkeitsgradienten aufweisen, besitzen in der Regel auch relevante Bildbereiche mit einem schwachen Signal-zu-Rausch-Verhältnis. Die Ermittlung des Normalenkontrastes wird durch das relativ zum Signal erhöhte Bildrauschen deutlich verzerrt. Eine Glättung findet zwar bereits durch die lineare Interpolation beim Sampling entlang der Normalen statt, diese reicht für eine zuverlässige Kantenerkennung innerhalb des Signals aber nicht aus.

Aus diesem Grund wird eine Signaladaption auf der Normalen vorgeschlagen, wodurch die Idee der lokalen Kontrastermittlung für die Bestimmung der Wertigkeit der Geschwindigkeitsfunktion beibehalten werden kann. Die Bausteine, aus denen sich das adaptierte Signal zusammensetzt, sind der gleitende Median und die gleitende, nach innen und nach außen gerichtete Varianz auf der Normalen.

Für eine Position \vec{x} auf der Kontur $\partial\Omega$ mit der nach innen zeigenden Konturnormalen $-\vec{N}$ wird nun ein adaptiertes Signal I' auf der Normalen berechnet.

Das eindimensionale, diskrete Signal $I_{\vec{x},\vec{N}}(k)$ ergibt sich zunächst durch lineare Interpolation der Intensitätswerte entlang der Normalen:

$$I_{\vec{x},\vec{N}}(k) = I\left(\vec{x} - \frac{k}{2}\vec{N}\right) \quad (4.14)$$

Dabei ist $k \in \mathbb{Z}$ und beschreibt das Sampling.

Die nach außen gerichtete Varianz $\text{Var}_a(I_{\vec{x},\vec{N}}(l), d_{\text{Var}})$ und die nach innen gerichtete Varianz $\text{Var}_i(I_{\vec{x},\vec{N}}(l), d_{\text{Var}})$ für eine Stelle l auf dem eindimensionalen Signal werden auf der Varianzdistanz d_{Var} bestimmt.

$$\text{Var}_a(I_{\vec{x},\vec{N}}(l), d_{\text{Var}}) = \text{Var}\left(I_{\vec{x},\vec{N}}(l), I_{\vec{x},\vec{N}}(l+1), \dots, I_{\vec{x},\vec{N}}(l+2d_{\text{Var}})\right) \quad (4.15)$$

$$\text{Var}_i(I_{\vec{x},\vec{N}}(l), d_{\text{Var}}) = \text{Var}\left(I_{\vec{x},\vec{N}}(l), I_{\vec{x},\vec{N}}(l-1), \dots, I_{\vec{x},\vec{N}}(l-2d_{\text{Var}})\right) \quad (4.16)$$

Eine ähnliche Definition ergibt sich für den Median. Dieser wird jedoch nicht einseitig nach innen oder außen ermittelt, sondern zentral um die Stelle l mit dem entsprechenden Radius r_M :

$$\text{Median}\left(I_{\vec{x},\vec{N}}(l), r_M\right) = \text{Median}\left(I_{\vec{x},\vec{N}}(l-2r_M), I_{\vec{x},\vec{N}}(l-2r_M+1), \dots, I_{\vec{x},\vec{N}}(l+2r_M)\right) \quad (4.17)$$

4.2.4.2 Konturnormalen-orthogonale Glättung

Abbildung 4.4 verdeutlicht die geometrischen Verhältnisse bei der Signaladaption entlang einer Normalen \vec{N} .

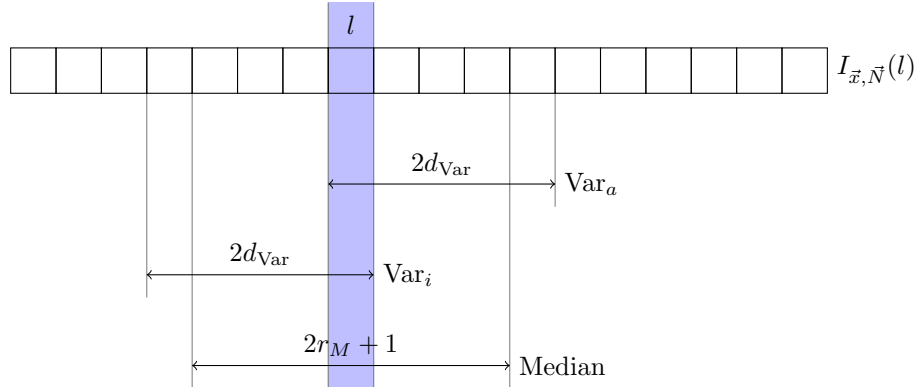


Abbildung 4.4: Gleitende Median- und Varianzbestimmung entlang der Sampling-Werte der Normalen für die Signaladaption gemäß (4.18).

Der adaptierte Signalwert I' ergibt sich nun durch eine Kombination der nach außen und innen gerichteten Varianz und dem gleitenden Medianwert:

$$I'_{\vec{x}, \vec{N}}(l) = \text{Median} \left(I_{\vec{x}, \vec{N}}(l), r_M \right) + w \frac{\text{Var}_a \left(I_{\vec{x}, \vec{N}}(l), d_{\text{Var}} \right) - \text{Var}_i \left(I_{\vec{x}, \vec{N}}(l), d_{\text{Var}} \right)}{\text{Median} \left(I_{\vec{x}, \vec{N}}(l), r_M \right)} \quad (4.18)$$

Die durch (4.18) vorgenommene Signaladaption hat zwei hervorzuhebende Eigenschaften. Erstens glättet der gleitende Median das Signal, ohne jedoch Kanten zu verschmieren, wie es beispielsweise bei der Verwendung des Mittelwertes der Fall gewesen wäre. Zweitens findet eine Kantenaufteilung genau dann statt, wenn sich eine Differenz zwischen der nach außen und nach innen gerichteten Varianz ergibt. Das ist insbesondere dann der Fall, wenn sich die zu findende Kante innerhalb einer der Varianzdistanzen befindet. Die Normierung auf den Median bewirkt hierbei, dass in dunklen Bildbereichen eine stärkere Aufteilung erfolgt. Dieser Einfluss kann durch den skalarwertigen Parameter w gewichtet werden (4.18).

Abbildung 4.5 auf Seite 41 zeigt exemplarisch die Wirkung der Signaladaption an drei teils problematischen Stellen eines Rattenhirn-MRTs. In den Abbildungen 4.5(a), (d) und (g) sind die Normalen markiert, entlang derer die Intensitätssamples genommen wurden. Die korrespondierenden, eindimensionalen Signale sind in den Abbildungen 4.5(b), (e) und (h) als schwarze Kurven, die adaptierten Signale als rote Kurven dargestellt. Die Abbildungen 4.5(c), (f) und (i) zeigen die aus dem Originalsignal und dem adaptierten Signal errechneten Geschwindigkeitsfunktionen. Ist die Geschwindigkeitsfunktion negativ, wird die Kontur nach außen gedrückt (vergleiche Abschnitt 3.2.2), ist sie positiv entsprechend nach innen. Das bedeutet gleichzeitig, dass die Geschwindigkeitsfunktion im Innern der Kontur stets negativ sein und erst auf der Kontur das Vorzeichen wechseln sollte. Die Samplegerade wurde so positioniert, dass der ideale Nulldurchgang von negativen zu positiven Geschwindigkeitsfunktionswerten bei dem Samplewert 30 liegt.

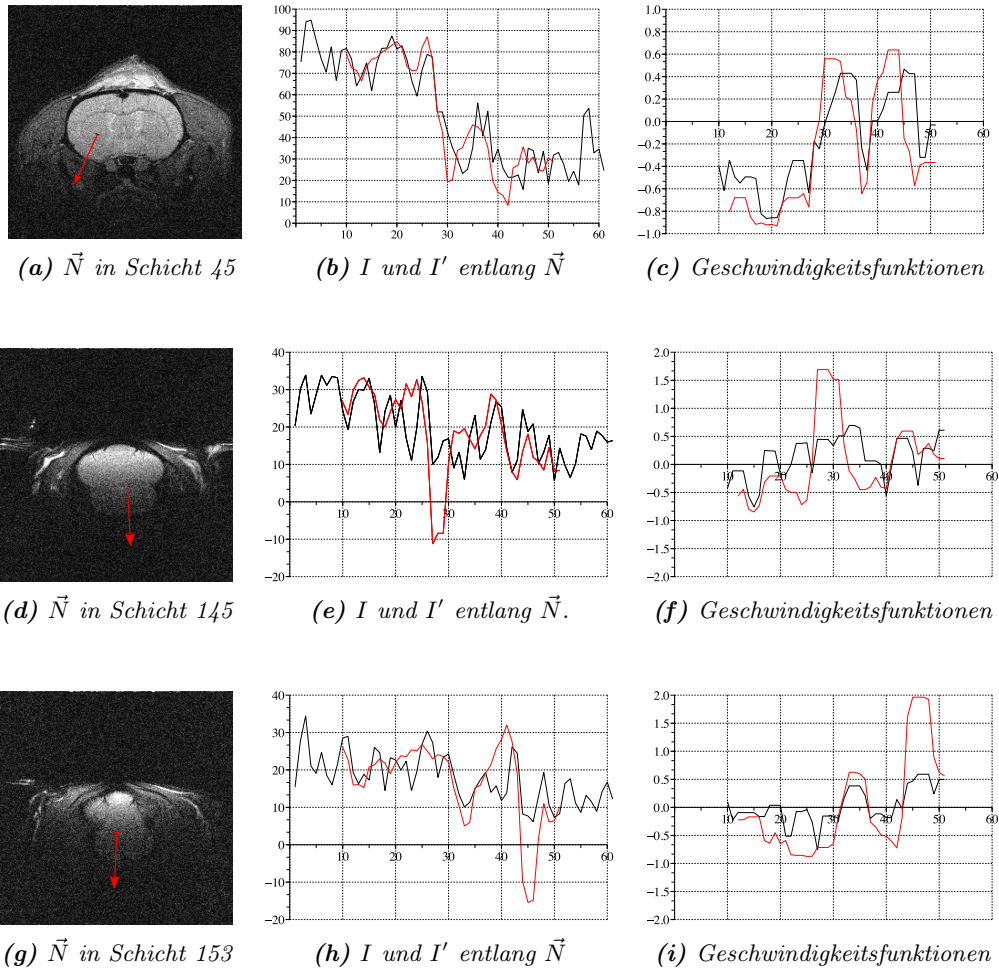


Abbildung 4.5: Die Wirkung der Signaladaption auf die Geschwindigkeitsfunktion an exemplarisch ausgewählten Positionen in teils stark verrauschten Bereichen. Das Signal entlang der Normalen \vec{N} ist von innen nach außen gesampelt worden. Die x -Achse entspricht der Normalen. Das ursprüngliche Signal und die zugehörige Geschwindigkeitsfunktion sind in schwarz dargestellt, das adaptierte Signal und die hierzu gehörende Geschwindigkeitsfunktion in rot. Nulldurchgänge der Geschwindigkeitsfunktion bedeuten einen Richtungswechsel der Konturpropagation. Die eingestellten Parameter für die Signaladaption waren: $r_M = 3$, $d_{\text{Var}} = 5$, $w = 3,5$, $d_m = 3$, $d_M = 10$ und $T = 0,5$.

Zu beachten ist hierbei natürlich, dass nicht allein die Geschwindigkeitsfunktion das Entwicklungsverhalten bestimmt, sondern zusätzlich auch die regularisierend wirkende Krümmung sowie die Geschwindigkeitsfunktion an benachbarten Stellen. Eine direkte Übertragung der exemplarisch dargestellten Geschwindigkeitsfunktionen auf das Segmentierungsergebnis der dreidimensionalen Verarbeitung ist daher nicht möglich. Sie kann nur das tendenzielle Verhalten der Funktion zeigen.

Der Nutzen der Signaladaption wird in Abbildung 4.5(f) sichtbar. Die Geschwindigkeitsfunktion des ursprünglichen Signals (schwarze Kurve) zeigt bereits um den Samplewert 16 und erneut bei 22 einen Vorzeichenwechsel, was zu einem vorzeitigen Segmentierungsabbruch führen würde. Die Geschwindigkeitsfunktion aus den adaptierten Werten wechselt zwar auch schon bei Position 26 das Vorzeichen – und nicht wie im Idealfall erst bei 30 – das ist aber nach einem Vergleich mit dem korrespondierendem Signalverlauf (Abb. 4.5(e)) nachvollziehbar.

Bei deutlichen Kanten (Abb. 4.5(a) bis (c)) erreicht die Geschwindigkeitsfunktion ohne Signaladaption einen Nulldurchgang an der Position 30, mit Signaladaption hingegen etwas weiter innen.

Darüber hinaus wird durch die Signaladaption die Lage des Nulldurchgangs der Geschwindigkeitsfunktion robuster gegenüber einer additiven Verschiebung der Funktion. Eine solche Verschiebung übernimmt aber genau der angestrebte Zielwert T in den Geschwindigkeitsfunktionen (4.9) und (4.10). Folglich sollten sich bezüglich der Wahl von T genauere und robustere Segmentierungsergebnisse durch die Signaladaption einstellen.

Der gleitenden Median entlang der Normalen führt zu einer Glättung des Signals in eben dieser Richtung. Problematisch kann dies an solche Stellen werden, wenn nur auf einer kurzen Distanz der Intensitätswert signifikant abfällt. Ist die Länge, auf der der Median ermittelt wird, groß gegenüber dieser Distanz, wird dieser lokal vorhandene Kontrast als Ausreißer gewertet und damit vom Algorithmus ignoriert. Solche Fälle können insbesondere in Bereichen auftreten, in denen Gehirngewebe dicht an die Schädeldecke angrenzt. Die unumgängliche Folge ist auftretendes Leakage.

Um dieses Problem zu vermeiden soll untersucht werden, inwiefern eine Glättung orthogonal zur Normalen zu besseren Ergebnissen führen kann. Dazu werden die Intensitätswerte nicht nur entlang einer Normalen ermittelt, sondern im dreidimensionalen Fall auf fünf und im zweidimensionalen Fall auf drei parallelen Geraden parallel zur Normalen. Die Samplewerte der zentralen Geraden werden dabei zweifach gewichtet.

Im dreidimensionalen Fall werden zu \vec{N} die Vektoren \vec{P}_1 und \vec{P}_2 so gewählt, dass diese ein rechtshändiges System bilden. Zusätzlich ist die Länge dieser Vektoren auf eins normiert: $|\vec{P}_1| = 1$ und $|\vec{P}_2| = 1$. Daraus ergibt sich das orthogonal

geglättete Signal:

$$I_{\vec{x}, \vec{N}}^o(k) = \frac{1}{6} \left(2I \left(\vec{x} - \frac{k}{2} \vec{N} \right) + \sum_{i=-1}^{+1} I \left(\vec{x} - \frac{k}{2} \vec{N} + i \vec{P}_1 \right) + I \left(\vec{x} - \frac{k}{2} \vec{N} + i \vec{P}_2 \right) \right) \quad (4.19)$$

Dieses Signal I^o kann zur Berechnung der Geschwindigkeitsfunktion genutzt werden, wobei die im vorherigen Abschnitt dargestellten Möglichkeiten der Signaladaption nach der orthogonalen Glättung ebenfalls gegeben sind. Abbildung 4.6 zeigt das orthogonal zur Normalen geglättete Signal an den bereits in Abbildung 4.5 exemplarisch ausgewählten Positionen mit den zugehörigen Geschwindigkeitsfunktionen.

4.2.5 Parameter

Basierend auf dem ausgewählten Algorithmus von Zhuang wurde die Geschwindigkeitsfunktion selbst vereinfacht und auf eine rein lokale Kontrastauswertung ausgerichtet. Darüber hinaus kann eine Signaladaption das in der Geschwindigkeitsfunktion ausgewertete Signal, wie im vorherigen Abschnitt beschrieben, verändern. Dafür stehen folgende Parameter zur Verfügung

- d_{\min} und d_{\max} : Die Längen auf denen die lokalen Intensitätsextrama ausgewertet werden.
- T : Der kontrastabhängige Zielwert.
- d_M : Die nach innen gerichtete Distanz auf der der lokale Median bestimmt wird.

Für die zu \vec{N} parallele Signaladaption stehen folgende Parameter zur Verfügung:

- r_M : Der Medianradius, welcher das Signal entlang der Normalen glättet.
- d_{var} : Die nach innen bzw. nach außen gerichtete Länge zur lokalen Varianzermittlung.
- w : Der Gewichtungsfaktor für die Kontrastaufteilung.

Die orthogonale Glättung kann zusätzlich zu den oben aufgeführten Parametern durchgeführt werden.

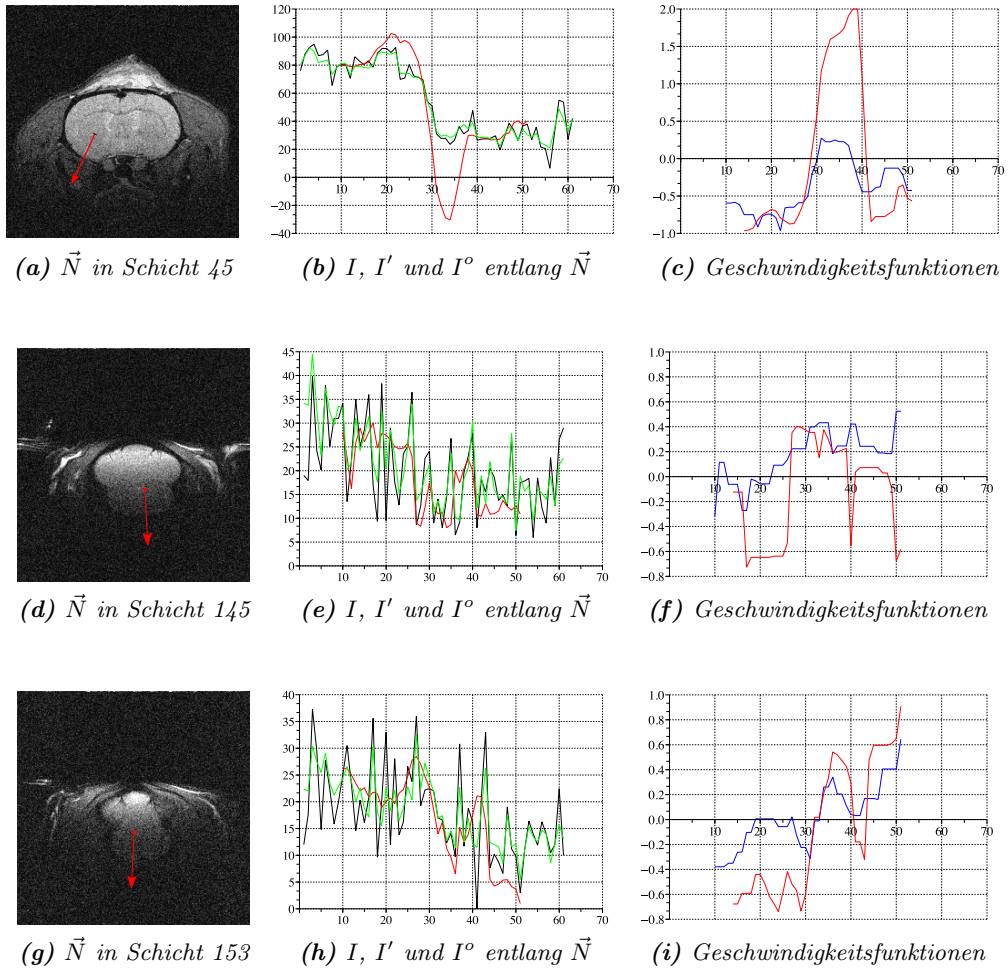


Abbildung 4.6: Auswirkungen der orthogonalen Glättung (grüne Kurven) auf das Signal entlang der gekennzeichneten Normalen (schwarze Kurven) und das adaptierte Signal (rote Kurven) mit den zugehörigen Geschwindigkeitsfunktionen.

Kapitel 5

Ergebnisse

Der implementierte Algorithmus hat eine Vielzahl von Parametern, welche potentiell das Segmentierungsergebnis bestimmen – eine komplette Abtastung des gesamten, hochdimensionalen Parameterraums ist folglich nicht möglich. Daher wird nach einer einführenden Darstellung des zu Grunde gelegten Qualitätsmaßes das Verhalten bezüglich ausgewählter Parameter in Versuchen analysiert.

5.1 Bildmaterial

Das verwendete Bildmaterial ist Eigentum des Forschungszentrums Jülich und wurde für diese Arbeit zur Verfügung gestellt. Es beinhaltet Rattenhirn Magnet-Resonanz-Tomographien (MRTs) von sieben Versuchstieren.

Der Datensatz TN211 wurden mit einem 9,4T Kleintier Magnet-Resonanz-Tomographen erstellt, mit einem Gradientenfeld der Stärke 270 mT/m und einer Oberflächenspule. Der Innendurchmesser betrug 21 cm. Die Aufnahmen entstanden an einer UnityInova Konsole (Varian, Palo Alto, USA). Das T2 gewichtete Echo ($TE = 24$ ms) ergab 40 Schichtbilder mit einer Auflösung von 256×256 Bildpunkten und einer Ortsauflösung von $0,117 \times 0,117 \times 0,5$ mm.

Die übrigen Datensätze weisen eine örtliche Auflösung von $0,2 \times 0,2 \times 0,23$ mm auf und stammen aus einem modifizierten 9 T Bruker ClinScan Kleintier-Scanner. Hier betrug $TE = 3,4$ ms bzw. $TE = 12$ ms.

Das Bildmaterial liegt in den gängigen Bildformaten *Analyze* beziehungsweise dem hierzu jüngeren Derivat *NIfTI* vor. Beide Dateiformate entstammen der *Neuroimaging Informatics Technology Initiative*, welche durch *The National Institute of Mental Health* und das *National Institute of Neurological Disorders and Stroke* unterstützt wird.

Die Datensätze zeigen bedingt durch die unterschiedlichen Aufnahmeparameter ein deutlich unterschiedliches Aussehen (Abb. 5.1). Dieser Sachverhalt muss bei den Versuchen berücksichtigt werden. Es ist nicht zu erwarten, dass Segmentierungsparameter welche für einen Scanner gute Ergebnisse liefern auf die Datensätze des anderen Scanners übertragbar sind und hier ebenfalls zu guten Ergebnissen führen.

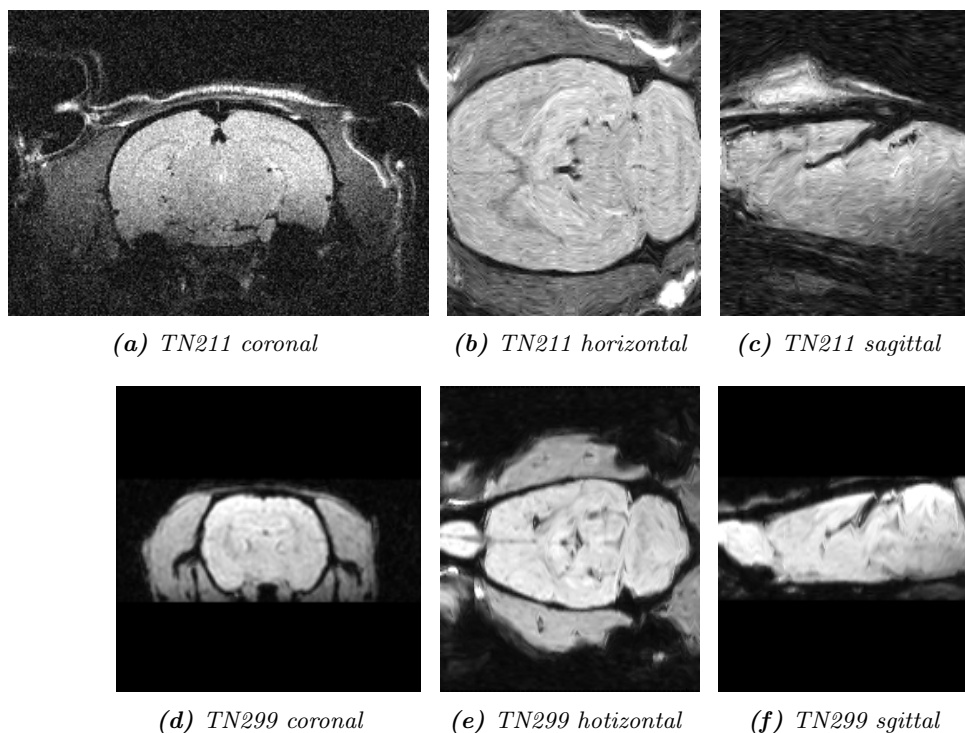


Abbildung 5.1: Gegenüberstellung der Datensätze TN211 und TN299.

5.2 Qualitätsbewertung

Ein objektives Qualitätsmaß ist für die Bewertung der Segmentierungsergebnisse von entscheidender Bedeutung. Das Segmentierungsergebnis des untersuchten Algorithmus muss folglich mit einer Referenzsegmentierung verglichen werden. Diese Referenz wurde manuell in sagittalen Schichten erstellt, wozu als Hilfsmittel der Paxinos Rattenhirnatlas [Pax98] verwendet wurde.

Die Datensätze wurden für die Segmentierung mittels einer registrierungsbasierenden Interpolation (siehe Abschnitt 4.2.2.1) zu annähernd isotropen Daten konvertiert. Die manuellen Segmentierungen wurden der gleichen Interpolation unterzogen. Damit besitzen die Daten eine interpolierte, örtliche Auflösung von $0,117 \times 0,117 \times 0,125$ mm bzw. $0,2 \times 0,2 \times 0,23$ mm.

Für eine quantitative Auswertung wird der *Dice-Koeffizient* herangezogen. Dieser setzt die Menge der vom Algorithmus zum Gehirn gehörig klassifizierten Bildpunkte M_A ins Verhältnis zu den manuell segmentierten Bildpunkten M_R . Der Dice-Koeffizient ist wie folgt definiert:

$$D(M_A, M_R) = \begin{cases} 2 \frac{|M_A \cap M_R|}{|M_A| + |M_R|} & \text{für } |M_A| + |M_R| \neq 0 \\ 0 & \text{sonst} \end{cases} \quad (5.1)$$

Datensatz	D		ΔD
	M_G	M_L	
TN211	0,903	0,929	0,026
TN276	0,917	0,949	0,032
TN283	0,927	0,938	0,011
TN290	0,932	0,936	0,004
TN291	0,610	0,669	0,059
TN299	0,933	0,935	0,003
TN301	0,908	0,916	0,007

Tabelle 5.1: Vergleich zwischen globaler (M_G) und lokaler Medianauswertung (M_L) für die Geschwindigkeitsfunktion.

Bei einer absoluten Übereinstimmung der manuellen und automatischen Segmentierung sind die Mengen M_A und M_R identisch. Der Dice-Koeffizient beträgt in diesem Idealfall $D(M_A, M_A) = D(M_R, M_R) = 1$. Gibt es hingegen keine Übereinstimmung der Ergebnisse, so ist die Schnittmenge leer und der Dice-Koeffizient ist folglich Null.

5.3 Auswertung lokaler Kontraste

In Abschnitt 4.2.3 wurde der modellbasierte Level-Set-Segmentierer so angepasst, dass allein die lokalen Kontraste für die Geschwindigkeitsfunktion ausschlaggebend sind. In einer ersten Versuchsreihe wird zunächst geklärt, ob diese Adaption bessere Ergebnisse liefert. Die zweite Versuchsreihe soll klären, wie die Parameter das Segmentierungsergebnis beeinflussen.

5.3.1 Lokaler und globaler Median

Die lokal adaptive Medianbestimmung lässt sich direkt mit der globalen Medianbestimmung vergleichen, indem die Parameter beider Algorithmen so weit wie möglich gleich gewählt und die Ergebnisse gegenüber gestellt werden. Dies ist für alle zur Verfügung stehenden Datensätze in Tabelle 5.1 geschehen.

Die Parameter der modellbasierten Level-Set-Segmentierung wurden mit $d_{\min} = 8$, $d_{\max} = 16$, $T = 0,5$ in einem Bereich gewählt, wie sie in [Zhu06] – allerdings für Humanhirndaten – vorgeschlagen wurden. Darüberhinaus wurden die allgemeinen Level-Set-Parameter mit $\beta = 1$, $\gamma = 0,25$, $\text{RMS}_{\max} = 0,0125$ und $n_{\max} = 2000$ gewählt (vgl. Abschnitt 3.3.4). Wenn nicht anders angegeben, sind diese allgemeinen Level-Set-Parameter auch bei den im weiteren Verlauf dargestellten Versuchsreihen verwendet worden.

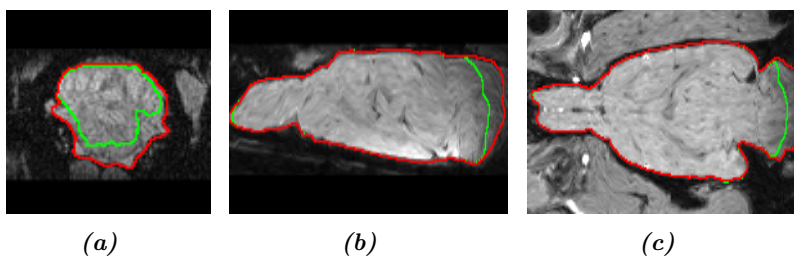


Abbildung 5.2: Vergleich der Segmentierungsergebnisse bei globaler (grün) und lokaler Medianauswertung (rot) und sonst identischen Parametern.

Der Dice-Koeffizient für den Datensatz TN291 ist sehr niedrig, da hier Leakage aufgetreten ist – beide Verfahren konnten dies nicht verhindern. Aus diesem Grund ist der Gewinn hier zwar betragsmäßig am größten, dennoch ist die Segmentierung als fehlgeschlagen zu werten. Der absolute Dice-Koeffizient fällt mit 0,610 bzw. 0,669 deutlich aus dem Rahmen.

Die lokal adaptive Medianbestimmung liefert für alle Datensätze bessere Ergebnisse, wobei der Gewinn bei Datensatz TN276 von 0,917 zu 0,949 am deutlichsten ausfällt. Gleichzeitig ist der resultierende Dice-Koeffizient hier auch betragsmäßig am größten.

Die Ergebniskonturen für den Datensatz TN283 bei lokaler bzw. globaler Medianauswertung sind in Abbildung 5.2 dargestellt. Das Ergebnis mit der globalen Medianauswertung ist als grüne, das der lokalen als rote Kontur gekennzeichnet. Es ist zu erkennen, dass insbesondere in denjenigen Randbereichen, welche einen geringeren Bildkontrast aufweisen, die lokale Kontrastauswertung das gewünschte Verhalten zeigt und ein frühzeitiges Stoppen der Konturevolution reduziert. Die rein lokale Kontrastauswertung ist daher der globalen Medianbestimmung innerhalb der Startkontur vorzuziehen.

5.3.2 Parametereinfluss

Für den Übergang von einer Kontrastbestimmung, in welche der innerhalb der Startkontur ermittelte Median einfließt, zu einer rein lokalen Kontrastbestimmung sind einmal der kontrastabhängige Zielwert T und die Längen, innerhalb derer der Kontrast ermittelt wird, als Parameter interessant.

5.3.2.1 Kontrastabhängiger Zielwert

In einer Versuchsreihe wurde für die lokale Kontrastauswertung der Zielwert T variiert, um herauszufinden, inwiefern die Qualität der Segmentierungsergebnisse hierdurch beeinflusst werden kann. Die Ergebnisse sind für drei Datensätze in

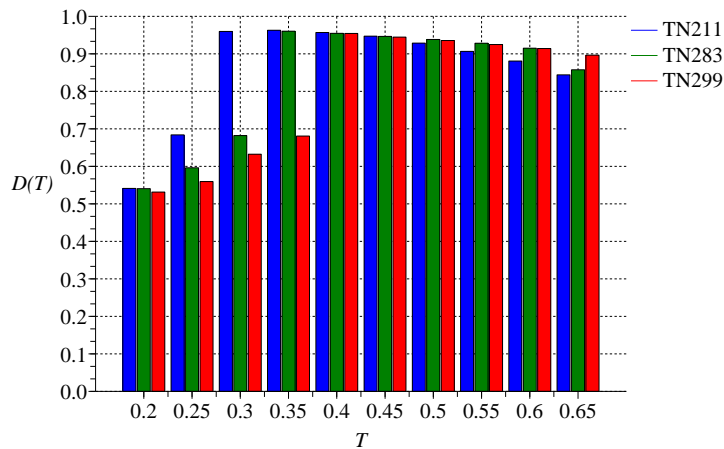


Abbildung 5.3: Der Dice-Koeffizient D für verschiedene Zielwerte T

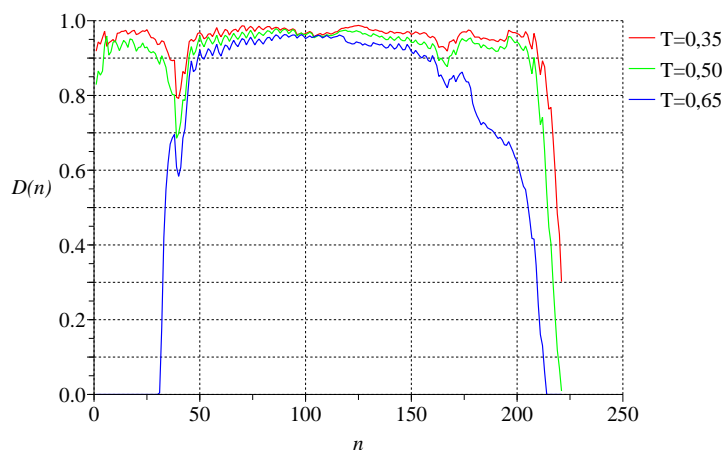


Abbildung 5.4: Der Dice-Koeffizient über die einzelnen Schichten des Datensatzes TN283 für ausgewählte Zielwerte T .

Abbildung 5.3 dargestellt. Es ist deutlich zu erkennen: wenn der kontrastbestimmende Zielwert zu niedrig gewählt wird – das entspricht hohen lokalen Kontrasten – bricht der Dice-Koeffizient bedingt durch Leakage ein.

Das beste Segmentierungsergebnis für die Datensätze TN211 und TN283 gemäß des Dice-Koeffizienten liegt hier bei einem Wert von $T = 0,35$, für den Datensatz TN299 bei $T = 0,4$. Ein zu hoch gewählter Zielwert T führt dazu, dass bereits bei geringen Kontrasten die Segmentierung stoppt. Die Kontur kann sich beispielsweise für $T = 0,65$ nicht weit genug in Randbereiche entwickeln – Abbildung 5.4 verdeutlicht dieses Verhalten. Hier ist für den Datensatz TN283 der Dice-Koeffizient für coronale Schichten aufgetragen.

Die frühe Entstehung von Leakage schon bei $T \leq 0,35$ bei Datensatz TN299 ist auf lokale Bildeigenschaften zurückzuführen. Bei der näheren Betrachtung von

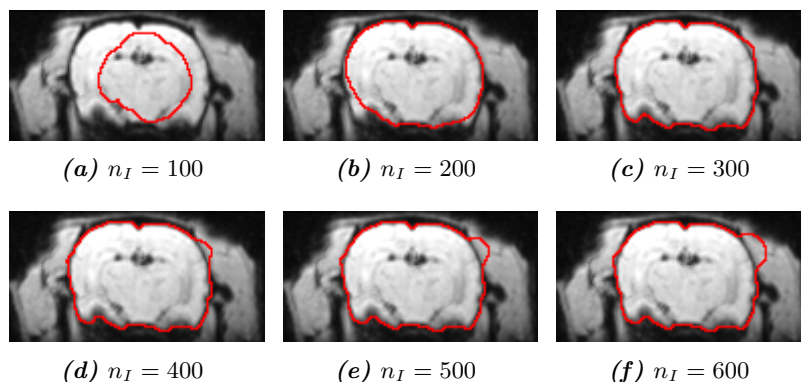


Abbildung 5.5: Leakage bedingt durch geringe lokale Kontraste (Datensatz TN299). $T = 0,35$ ist zu niedrig gewählt, so dass die Kontur nicht stoppt.

Zwischenergebnissen des Segmentierungsprozesses (Abbildung 5.5) lässt sich die Position lokalisieren, an der die Kontur das erste Mal, bedingt durch niedrige lokale Kontraste, aus der eigentlich gesuchten Rattenhirnkontur ausbricht. Dieser niedrige Kontrast hat zwei Ursachen. Zum einen ist die Ortsauflösung des Scanners niedrig gegenüber der Größe der abzubildenden Struktur. Hinzu kommt die Interpolation zu isotropen Daten (Abschnitt 4.2.2.1), die zwar im Vergleich zu anderen Verfahren sehr gute Ergebnisse liefert, aber dennoch eine gewisse Glättung und damit eine Abnahme des Kontrastes der Bilddaten verursacht.

Bei einem Vergleich der Konturpositionen für die Iterationen $n_I = 200$ und $n_I = 300$ in Abbildung 5.5(b) und (c) zeigt sich, dass die Bewegungsgeschwindigkeit an der kritischen Position reduziert ist – insbesondere im Vergleich zur Entwicklungsweite der Kontur zwischen den Iterationen $n_I = 100$ und $n_I = 200$ – dort aber nicht komplett zum Stehen kommt.

Leakage ist in mehreren Versuchen an ähnlichen Positionen aufgetreten, wie sie in Abbildung 5.5 exemplarisch dargestellt ist. Auf der Konturnormalen ist ein hell-dunkel-hell-Kontrast an der Objektgrenze zu beobachten. Obwohl die Geschwindigkeitsfunktion lediglich das absolute Minimum und Maximum auf der Konturnormalen auswertet, scheint das Modell bei einem solchen Kontrast, im Vergleich zu einem reinen hell-dunkel-Kontrast, Schwächen zu haben.

Die Wahl des kontrastabhängigen Zielwertes T ist datenabhängig und entscheidend für die Qualität der Segmentierung. Im Hinblick auf das plötzlich einbrechende Verhalten hin zu niedrigen Werten von T , wird man in der Praxis den Wert tendenziell größer wählen, um nicht die Schwelle des Leakage zu überschreiten.

5.3.2.2 Kontrastlängen

Die Wahl der Längen d_m und d_M , auf welchen der Zielwert T zum Stoppen des Segmentierungsprozesses erreicht werden muss, kann zwar unabhängig von-

d_m	d_M	D		
		TN211	TN283	TN299
2,0	4,0	0,908	0,931	0,947
4,0	8,0	0,906	0,930	0,946
6,0	12,0	0,904	0,930	0,938
8,0	16,0	0,929	0,938	0,935
10,0	20,0	0,902	0,927	0,934
12,0	24,0	0,899	0,929	0,933

Tabelle 5.2: Dice-Koeffizient in Abhängigkeit zur Wahl der Kontrastlängenbestimmung bei rein lokaler Kontrastauswertung.

einander erfolgen, damit aber im weiteren Verlauf dieser Arbeit die Anzahl der Parameter überschaubar bleibt, wurden diese Längen in der Regel so gewählt, dass d_m , auf der das lokale Minimum ermittelt wird, die Hälfte von d_M beträgt, auf der das lokale Maximum bestimmt wird: $2d_m = d_M$. Es ist sinnvoll die Länge der Minimumsbestimmung kürzer zu wählen, da bei dem vorliegenden Bildmaterial die Kanten durch minimale Werte ausgezeichnet sind. Daraus ergibt sich eine exaktere Lokalisierung der Kontur auf dem Minimum selbst.

Für drei Datensätze (TN211, TN283 und TN299), welche in Abschnitt 5.3.1 durchschnittlich gute Segmentierungsergebnisse lieferten, wurde die Länge der Minimumssuche in einem Bereich von $d_m = 2 \dots 12$ respektive die der Maximumssuche in einem Bereich von $d_M = 4 \dots 24$ variiert. Die resultierenden Dice-Koeffizienten sind in Tabelle 5.2 dargestellt. Die Kontrastauswertung erfolgte rein lokal und die Level-Set-Parameter wurden mit dem Propagationsgewicht zu $\beta = 1$, dem Krümmungsgewicht zu $\gamma = 0,25$ und den Abbruchkriterien zu $\text{RMS}_{\max} = 0,125$ und $n_{\max} = 2000$ gewählt.

Die Ergebnisse fallen bei dieser Versuchsreihe nicht so signifikant unterschiedlich aus, wie bei der Variation des Zielwertes T . Der Dice-Koeffizient schwankt nur geringfügig trotz großzügiger Variation der absoluten Längen. Die deutlichsten Auswirkungen sind für den Datensatz TN211 bei einer optimalen Länge von $d_m = 8$ und $d_M = 16$ zu beobachten. Da diese auch für TN283 das beste Resultat lieferte, wurden diese Längen für die weiteren Versuche beibehalten.

Es bleibt festzuhalten, dass die Wahl der optimalen Längen zur Kontrastbestimmung d_m und d_M für TN283 und TN299 unkritischer ist, als die eines korrekten Zielwertes T . Für TN211 lässt sich ein scharfes Maximum des Dice-Koeffizienten in dieser Versuchsreihe für $d_m = 8$ und $d_M = 16$ beobachten.

r_M	D		
	TN211	TN283	TN299
0,5	0,952	0,942	0,939
1,0	0,956	0,942	0,940
1,5	0,966	0,941	0,940
2,0	0,967	0,940	0,940
2,5	0,966	0,759	0,657
3,0	0,846	0,629	0,654
3,5	0,703	0,625	0,651
4,0	0,522	0,623	0,596

Tabelle 5.3: Glättung des Signals durch einen gleitenden Median entlang der Normalenrichtung.

5.4 Signaladaption

In Abschnitt 4.2.4 wurde eine Signaladaption vorgeschlagen, welche aus Glättungen orthogonal und parallel zur Konturnormalen und einer Kontrasterhöhung zusammengesetzt werden kann. Um das Verhalten dieser Modifikationen beurteilen zu können, werden zunächst beide Glättungsarten einzeln getestet und anschließend kombiniert. Es folgt eine Untersuchung, inwiefern die Kontrasterhöhung die Ergebnisse verbessern kann. Es ist zu erwarten, dass bei einer zu starken Glättung – ähnlich wie bei der Wahl eines zu kleinen T – der Dice-Koeffizient D einbricht.

5.4.1 Konturnormalen-parallele Glättung

Diese Glättung verwendet einen gleitenden Median mit dem Radius r_M entlang der Normalen (Abschnitt 4.2.4.1). Für die Datensätze TN211, TN283 und TN299 wurde diese Versuchsreihe mit den allgemeinen Parametern $d_m = 8$, $d_M = 16$, $T = 0,5$ durchgeführt. Die Ergebnisse sind in Tabelle 5.3 dargestellt.

Es zeigt sich, dass die Glättung entlang der Normalen – bis zu einem gewissen Grad – zu einem besseren Segmentierungsergebnis führt. Diese Beobachtung lässt sich insbesondere für den Datensatz TN211 machen. Der maximale Dice-Koeffizient wird hier bei einem Medianradius $r_M = 2$ erreicht. Bei einer Überglättung hingegen wird der Zielkontrast nicht erreicht und es tritt Leakage auf.

5.4.2 Konturnormalen-orthogonale und parallele Glättung

Um die Auswirkungen der zur Konturnormalen orthogonalen Glättung allein und in der Kombination mit der parallelen Glättung zu verdeutlichen, wurde basie-

r_M	D	
	G_{\parallel}	$G_{\parallel,\perp}$
0,5	0,952	0,966
1,0	0,956	0,968
1,5	0,966	0,968
2,0	0,967	0,876
2,5	0,966	0,793
3,0	0,846	0,693
3,5	0,703	0,484
4,0	0,522	0,408

Tabelle 5.4: Abhängigkeit der optimalen Distanz, auf der der gleitende Median ermittelt wird, zur orthogonalen Glättung für den Datensatz TN211.

rend auf der Messreihe, welche im vorangegangenen Abschnitt bereits vorgestellt wurde, die Messung mit zusätzlicher Konturnormalen-paralleler Glättung durchgeführt. Es besteht die Hypothese, dass bei einer zusätzlichen Glättung die ideale Mediandistanz kürzer wird. Die Ergebnisse sind in Tabelle 5.4 dargestellt.

Die rein parallele Glättung ist mit G_{\parallel} und die Kombination aus beiden Glättungen mit $G_{\parallel,\perp}$ bezeichnet. Der optimale Medianradius wird tatsächlich halbiert, gleichzeitig steigt der Dice-Koeffizient geringfügig.

5.4.3 Kontrasterhöhung

Die Kontrasterhöhung gemäß (4.18) sorgt dafür, dass an den Kanten des zu segmentierenden Objektes ein deutlich erhöhter Kontrast erreicht wird. Diesen Effekt ruft die Differenz der nach außen und innen bestimmten Varianz entlang der Normalen hervor. Folglich ist ein gegenläufiger Effekt zwischen dem Zielkontrast T und dem Varianzgewicht w zu beobachten.

Dieses Verhalten wurde in einer Versuchsreihe näher untersucht. Die zugehörigen Messwerte zeigt Tabelle 5.5. Die gegenläufige Tendenz ist deutlich zu erkennen, wenn man die jeweils besten Segmentierungsergebnisse über die Variation der Zielwerte T betrachtet. Ob dieser Zusammenhang allerdings linear ist, kann aus der Versuchsreihe nicht abschließend geklärt werden.

Die Verwendung der Kontrasterhöhung zeigt zwei vorteilhafte Aspekte. Zum einen steigt der absolut erreichte Dice-Koeffizient mit dem gewählten Varianzgewicht w respektive mit dem kontrastabhängigen Zielwert T . Zum anderen wird der Bereich, in dem der Dice-Koeffizient auf einem hohen Niveau bleibt, verbreitert. Dieser Sachverhalt ist in Abbildung 5.6 deutlich zu erkennen.

w	D					
	$T = 0,45$	$T = 0,40$	$T = 0,35$	$T = 0,30$	$T = 0,25$	$T = 0,20$
0,5	0,913	0,678	0,482	0,400	0,327	0,295
1,0	0,963	0,900	0,618	0,488	0,363	0,322
1,5	0,965	0,961	0,897	0,664	0,426	0,344
2,0	0,965	0,962	0,952	0,869	0,584	0,379
2,5	0,953	0,965	0,960	0,902	0,689	0,437
3,0	0,951	0,960	0,961	0,923	0,770	0,495
3,5	0,942	0,958	0,964	0,938	0,827	0,548
4,0	0,940	0,954	0,964	0,947	0,866	0,614
4,5	0,938	0,951	0,962	0,965	0,899	0,690
5,0	0,936	0,949	0,959	0,966	0,921	0,770
5,5	0,934	0,947	0,958	0,964	0,938	0,827
6,0	0,931	0,944	0,957	0,963	0,966	0,871
6,5	0,927	0,943	0,956	0,962	0,966	0,907
7,0	0,919	0,941	0,955	0,961	0,966	0,923
7,5	-	-	-	-	0,966	0,966
8,0	-	-	-	-	0,965	0,966
8,5	-	-	-	-	0,964	0,966
9,0	-	-	-	-	0,963	0,967
9,5	-	-	-	-	-	0,967
10,0	-	-	-	-	-	0,967
10,5	-	-	-	-	-	0,966
11,0	-	-	-	-	-	0,965

Tabelle 5.5: Der Dice-Koeffizient D in Abhängigkeit vom Varianzgewicht w und dem kontrastabhängigen Zielwert T . Zu beachten ist der Zusammenhang zwischen der Reduzierung von T und der Erhöhung von w für ein maximales D .

5.5 Verfahren im Vergleich

Um die Leistungsfähigkeit der Verfahren zu untersuchen, wurden jeweils die optimalen Parameter eines Verfahrens zu dem Datensatz TN211 als Grundlage gewählt und von hier ausgehend der Einfluss der allgemeinen Level-Set-Parameter (Abschnitt 3.3.4) und die Abhängigkeit von der Wahl der Startkontur untersucht. Die übrigen zu Grunde gelegten Verfahrensparameter waren:

- *Globale Medianauswertung* (M_G): $d_{\min} = 8$, $d_{\max} = 16$, $T = 0,25$
- *Rein lokale Kontrastauswertung* (M_L): $d_{\min} = 8$, $d_{\max} = 16$, $T = 0,35$, $d_{\text{Median}} = 20$
- *Konturnormalen-parallele Glättung* (G_{\parallel}): $d_{\min} = 8$, $d_{\max} = 16$, $T = 0,5$, $r_M = 2$

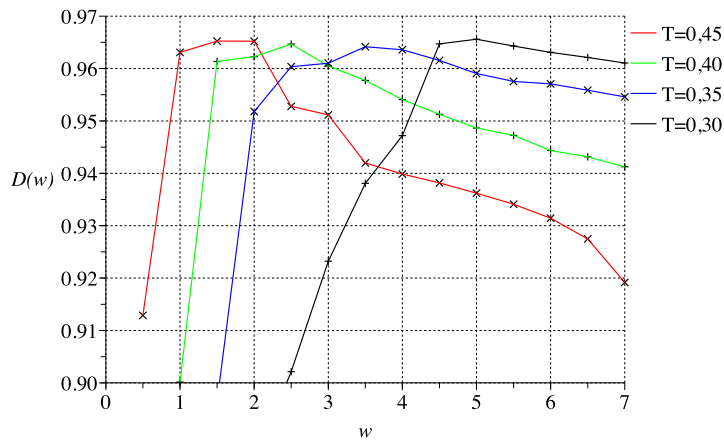


Abbildung 5.6: Der Dice-Koeffizient D in Abhängigkeit vom Varianzgewicht w für unterschiedliche Zielwerte T . Je höher das Varianzgewicht und damit der Effekt der Kontrasterhöhung, desto robuster wird w gegenüber Variationen: Der Dice-Koeffizient fällt zu höheren Varianzgewichten langsamer ab.

- Konturnormalen-orthogonale Glättung (G_{\perp}): $d_{\min} = 8$, $d_{\max} = 16$, $T = 0,45$
- Konturnormalen-orthogonale und parallele Glättung ($G_{\parallel,\perp}$): $d_{\min} = 8$, $d_{\max} = 16$, $T = 0,5$, $r_M = 1$
- Konturnormalen-orthogonale und parallele Glättung mit Kontrasterhöhung ($G_{\parallel,\perp,w}$): $d_{\min} = 8$, $d_{\max} = 16$, $T = 0,45$, $r_M = 1$, $d_{\text{Var}} = 6$, $w = 2$

5.5.1 Allgemeine Level-Set-Parameter

Die allgemeinen Level-Set-Parameter setzen sich zusammen aus dem relativen Verhältnis zwischen Krümmungsgewicht γ und dem Konvergenzkriterium, welches mittels RMS-Wert bestimmt wird. Beide Parameter wurden einzeln variiert und die Segmentierungsergebnisse auf die bekannte Art verglichen. Das Propagationsgewicht wurde konstant auf $\beta = 1$ belassen, da nur die relative Gewichtung zwischen Krümmung und Propagation $\frac{\gamma}{\beta}$ Bedeutung ist.

5.5.1.1 Krümmung

Das Krümmungsgewicht γ bewirkt, dass eine mehr oder weniger glatte Lösung des Segmentierungsergebnisses resultiert. Gibt man durch Herabsetzen dieses Regularisierungsfaktors der Kontur die Möglichkeit, sich in beliebige Richtungen zu entwickeln, so sollten hier insbesondere die Verfahren, welche eine Signalglättung einsetzen, weniger stark durch Rauschen im Bild beeinflusst werden. Diese These wird durch Tabelle 5.6 jedoch nur bedingt gestützt.

γ	D					
	M_G	M_L	G_{\parallel}	G_{\perp}	$G_{\parallel,\perp}$	$G_{\parallel,\perp,w}$
0,30	0,948	0,966	0,968	0,969	0,968	0,961
0,25	0,948	0,964	0,967	0,967	0,968	0,965
0,20	0,947	0,965	0,939	0,965	0,964	0,961
0,15	0,947	0,964	0,917	0,928	0,953	0,964
0,10	0,947	0,881	0,866	0,914	0,943	0,962
0,05	0,947	0,857	0,810	0,873	0,918	0,962
0,00	0,945	0,830	0,683	0,777	0,856	0,960

Tabelle 5.6: Dice-Koeffizienten in Abhängigkeit des Krümmungsgewichtes γ . Die Segmentierungen wurden mit den für die einzelnen Verfahren optimalen Parametern durchgeführt.

Überraschender Weise zeigt das Verfahren mit der globalen Medianauswertung robuste Ergebnisse gegenüber der Variation des Krümmungsgewichtes. Bei den Verfahren, welche eine lokale Signalglättung durchführen, zeichnet sich die Tendenz ab, dass die kombinierte Glättung in orthogonaler und paralleler Richtung im Vergleich zur rein orthogonalen bzw. parallelen Glättung bis zu einem niedrigeren Glättungsfaktor bessere Ergebnisse liefert.

Ähnlich wie M_G robuste Ergebnisse bis zu einem Krümmungsfaktor von $\gamma = 0$ zeigt, sind auch die Ergebnisse des Verfahrens mit kombinierter Glättung und Signalaufteilung $G_{\parallel,\perp,w}$ robust – dies jedoch auf einem höheren Niveau.

5.5.1.2 Konvergenzkriterium

Der RMS-Wert dient für alle Tests als Konvergenzkriterium, sofern eine vorgegebene Anzahl von maximal durchzuführenden Iterationen erreicht wird. Für diese Versuchsreihe wurde diese Anzahl von $n_{\max} = 2000$ auf $n_{\max} = 3000$ heraufgesetzt. Senkt man den RMS-Wert sukzessive ab, so lässt sich feststellen, wie stark eine berechnete Lösung des Segmentierungsproblems von der korrekten Wahl dieses Parameters abhängig ist. Hier ist insbesondere die Fragestellung nach Leakage interessant – ein robustes Verfahren sollte im Idealfall durch Herabsetzen des RMS-Wertes nicht beeinflusst werden. Leakage weist bei einem niedrig gewählten Konvergenzkriterium darauf hin, dass die Kontur nicht komplett auf der Objektkontur angehalten, sondern sich stattdessen mit einer niedrigen Geschwindigkeit weiter entwickelt hat. Die Ergebnisse zu dieser Versuchsreihe sind in Tabelle 5.7 dargestellt.

Es zeigt sich anhand des sehr niedrigen Dice-Koeffizienten, dass bei den Verfahren M_G , M_L , G_{\parallel} , G_{\perp} bei zu geringem RMS_{\max} deutliches Leakage auftritt. Robuste Ergebnisse zeigen hingegen $G_{\parallel,\perp}$ und $G_{\parallel,\perp,w}$.

RMS _{max}	D					
	M_G	M_L	G_{\parallel}	G_{\perp}	$G_{\parallel,\perp}$	$G_{\parallel,\perp,w}$
0,010	0,948	0,964	0,967	0,967	0,968	0,965
0,009	0,833	0,965	0,967	0,966	0,966	0,962
0,008	0,809	0,772	0,967	0,760	0,968	0,960
0,007	0,795	0,771	0,645	0,760	0,967	0,960
0,006	0,793	0,771	0,645	0,760	0,967	0,961
0,005	0,793	0,770	0,645	0,749	0,967	0,961

Tabelle 5.7: Dice-Koeffizienten in Abhängigkeit des Konvergenzkriteriums RMS_{max}. Die Segmentierungen wurden mit den für die einzelnen Verfahren optimalen Parametern durchgeführt.

5.5.2 Wahl der Startkontur

Ein Argument für den Übergang zu der Ermittlung rein lokaler Kontraste, war die Abhängigkeit des Segmentierungsergebnisses von der Wahl der Startkontur. Die Segmentierungsergebnisse sollten folglich robuster werden, wenn diese lokale Abhängigkeit nicht mehr besteht.

Die Versuchsreihe wurde am Datensatz TN211 durchgeführt, da dieser einen ausgeprägten, globalen Helligkeitsgradienten zeigt, so dass die beschriebene Abhängigkeit hier am deutlichsten hervortreten sollte. Der interpolierte Datensatz hat eine Abmessung von $256 \times 256 \times 153$ Voxeln. Die Startkonturen hatten einen Radius von 10 Voxeln und wurden in ihrer horizontalen Position – das entspricht der z -Position – variiert. Die Startpositionen sind:

$$\vec{x}_{S1} = \begin{pmatrix} 128 \\ 128 \\ 25 \end{pmatrix} \quad \vec{x}_{S2} = \begin{pmatrix} 128 \\ 128 \\ 50 \end{pmatrix} \quad \vec{x}_{S3} = \begin{pmatrix} 128 \\ 128 \\ 100 \end{pmatrix} \quad \vec{x}_{S4} = \begin{pmatrix} 128 \\ 128 \\ 125 \end{pmatrix}$$

Die zugehörigen Startkonturen sind in Abbildung 5.7 dargestellt.

Die resultierenden Dice-Koeffizienten sind in Tabelle 5.8 aufgelistet. Niedrige Werte bei den Verfahren ohne Glättung resultieren allein durch Leakage. Offensichtlich wurde für M_G und M_L der Zielwert T zu niedrig gewählt, als dass diese Verfahren gegenüber der Startkonturvariation robust sind. Auch hier wird die Bedeutung der Wahl des richtigen Zielwertes T nochmals deutlich.

Die Startkontur an der Position \vec{x}_{S4} liegt teilweise ausserhalb des zu segmentierenden Rattenhirns. Auch wenn bei dem implementierten Algorithmus von der Voraussetzung ausgegangen worden ist, dass sich die Kontur stets von innerhalb des Objektes hin zu den Objektgrenzen entwickelt, zeigt sich hier doch eine gewisse Stabilität gegenüber der Nichteinhaltung dieser Forderung. In begrenztem Rahmen kann sich die Kontur auch wieder zurück zum Objekt bewegen.

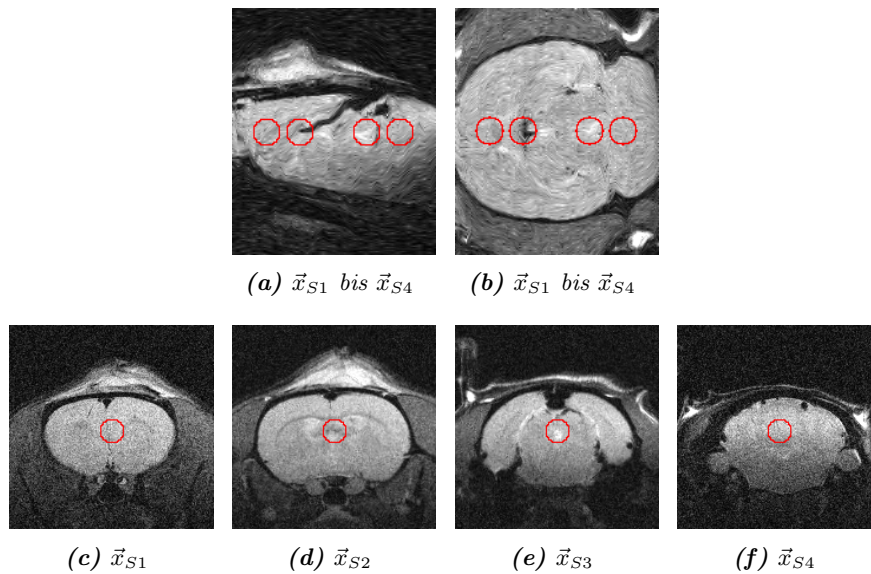


Abbildung 5.7: Startkonturen für die Positionen \vec{x}_{S1} bis \vec{x}_{S4}

Die Position \vec{x}_{S4} liefert bei allen Verfahren die besten Segmentierungsergebnisse. Das lässt sich dadurch begründen, dass insbesondere in dem schwachkontrastigen Bereich um diese Position hin zum Cerebellum, d.h. zum rechten Bildrand in sagittaler Richtung (Abb. 5.7(a)) die Kontur genügend Iterationen zur Verfügung hat, um sich dort hinein auszubreiten. Die Propagation läuft hier entsprechend langsam ab. Dieser Effekt ist bei dem Einsatz der orthogonalen und parallelen Glättung nur sehr schwach ausgeprägt.

Bemerkenswert ist darüber hinaus die Konstanz der Ergebnisse auf hohem Niveau, sobald eine glättende Signaladaption vorgenommen wird. Die Notwendigkeit einer optimalen Wahl der Startkontur wird hierdurch reduziert.

Pos.	D					
	M_G	M_L	G_{\parallel}	G_{\perp}	$G_{\parallel,\perp}$	$G_{\parallel,\perp,w}$
\vec{x}_{S1}	0,834	0,773	0,967	0,965	0,968	0,962
\vec{x}_{S2}	0,835	0,961	0,966	0,966	0,968	0,963
\vec{x}_{S3}	0,952	0,969	0,967	0,970	0,968	0,966
\vec{x}_{S4}	0,956	0,972	0,968	0,970	0,968	0,966
σ^2	$4,8 \cdot 10^{-3}$	$9,5 \cdot 10^{-3}$	$6,7 \cdot 10^{-7}$	$6,9 \cdot 10^{-6}$	0	$4,3 \cdot 10^{-6}$

Tabelle 5.8: Dicekoeffizienten in Abhängigkeit der Positionierung der Startkontur. Die Segmentierungen wurden mit den für die einzelnen Verfahren optimalen Parametern durchgeführt und die Varianz σ^2 der Dice-Koeffizienten für jedes Verfahren berechnet.

Kapitel 6

Zusammenfassung

Die Level-Set-Methoden stellen einen mächtigen, methodischen Rahmen zur Verfügung, um als Lösungsansatz für eine Vielzahl von Konturpropagationsproblemen eingesetzt werden zu können. In dieser Arbeit wurden diese Methoden genutzt, um in Rattenhirn-Magnet-Resonanz-Tomographien das Gehirngewebe aus den abgebildeten, morphologischen Strukturen zu segmentieren.

Ein kontrastbasierter pseudo-3D-Segmentierer, vorgeschlagen von Zhuang [Zhu06], wurde in einen echt-dreidimensionalen Ansatz überführt. Dabei wurde die Flexibilität der Level-Set-Methoden bezüglich einer beliebigen Dimensionalität ausgenutzt.

Für den iterativ ablaufenden Segmentierungsprozess ist die Geschwindigkeitsfunktion von entscheidender Bedeutung – sie bestimmt, basierend auf lokalen Kontrasten, wie sich die Kontur entwickelt und ist somit für die Qualität des Segmentierungsergebnisses verantwortlich.

Eine besondere Herausforderung stellen Datensätze mit einem ausgeprägten, globalen Helligkeitsgradienten dar, da hier eine zuverlässige Detektion von Objektgrenzen in Bildbereichen mit einem schwachen Signal-zu-Rausch Verhältnis problematisch ist. Es wurde gezeigt, dass der Übergang zur Auswertung rein lokaler Kontraste insbesondere einen frühzeitigen Abbruch des Konturevolutionsprozesses reduzieren kann. Die Wahl des kontrastbestimmenden Zielwertes ist für das Segmentierungsergebnis beider Verfahren kritisch.

Um den kontrastbasierten Ansatz beibehalten und gleichzeitig auf die beschriebenen Bildeigenschaften adäquat eingehen zu können, wurde eine Signaladaption entwickelt, welche statistische Bildeigenschaften gleitend auf der Konturnormalen vorwärts- und rückwärtsgerichtet vergleicht. Sie erkennt Kanten durch Abweichungen dieser Eigenschaften zuverlässiger als eine Kontrastauswertung des Signals allein. Ermittelte Abweichungen führen zu einer Kontrasterhöhung, welche die Konturevolution stoppt.

Zusätzlich wurden Glättungsverfahren eingesetzt und getestet, welche entlang der Konturnormalen kantenerhaltend arbeiten und orthogonal hierzu das Sampling der Intensitätswerte auf der Konturnormalen robuster machen. Lokale Glättungen und die kontrasterhöhende Signaladaption können beliebig miteinander kombiniert werden.

Es hat sich gezeigt, dass die eingesetzte Glättung durch den auf der Konturnormalen gleitenden Median insbesondere bei dem höher aufgelösten Datensatz TN211 erwartungsgemäß zu besseren Segmentierungsergebnissen führt. Eine Überglättung hingegen bedingt Leakage. Wird die Glättung orthogonal zur Konturnormalen hinzugenommen, verkürzt sich der Radius, auf dem der Median für ein optimales Segmentierungsergebnis bestimmt wird.

Die mit den Glättungen kombinierte Signaladaption zeigt sehr gute Konvergenzeigenschaften und ein robustes Verhalten gegenüber der Wahl des regularisierenden Krümmungsgewichtes. Die Wahl der optimalen allgemeinen Level-Set-Parameter ist somit unkritischer.

Es konnte eine deutliche Korrelation zwischen dem Zielwert und der Kontrasterhöhung demonstriert werden, welche im gleichen Zuge die Qualität der Segmentierung steigert. Die zugehörigen Dice-Koeffizienten-Maxima werden breiter und zeigen dadurch gegenüber der Wahl eines optimalen Kontrasterhöhungsgewichtes auch hier einen größeren Spielraum.

6.1 Diskussion

2D versus 3D Durch den Übergang von dem Pseudo-3D-Ansatz nach Zhuang zu einem echt-dreidimensionalen Segmentierer gibt man die Möglichkeit einer automatischen Leakage-Detektion, basierend auf dem Vergleich unabhängiger 2D-Segmentierungsergebnisse, auf. Der Pseudo-3D-Ansatz begegnet detektiertem Leakage durch die sukzessive Reduzierung der Krümmung der entwickelten Kontur, wodurch das Segmentierungsergebnis tendenziell an Genauigkeit verliert, gleichzeitig aber robuster gegenüber lokalen Kontrastschwankungen wird.

Die Variation der Krümmung kann in aufeinanderfolgenden Schichten orthogonal zur Segmentierungsebene wiederum in starken Krümmungen der Oberfläche resultieren. Im Gegensatz dazu berücksichtigt der dreidimensionale Ansatz die morphologischen Strukturen in allen Raumrichtungen gleichförmig.

Ein zweidimensionaler Ansatz kann Krümmungen orthogonal zur Schnittebene nicht berücksichtigen. Daher bestand die These, dass ein dreidimensionaler Konturevolutionsansatz in einer Dimension stark gekrümmte Bereiche überglättet. Dies konnte jedoch nicht bestätigt werden, denn selbst der tubusförmige Kanal, welcher unterhalb des Dentate Gyrus in das Rattenhirn hinein verläuft (vgl. Abb. 5.7(a), S. 58), war in vielen Segmentierungsergebnissen gut sichtbar.

Bei einem Geschwindigkeitsvergleich zwischen dem Pseudo-3D-Ansatz von Zhuang im Vergleich zur Implementation des echt-dreidimensionalen Ansatzes dieser Arbeit gewinnt eindeutig der schichtweise zweidimensionale Ansatz. Allerdings sind die Anforderungen in diesem Bereich nicht zeitkritisch, so dass der Segmentierer mit den besseren Ergebnissen bevorzugt werden sollte.

Parameterraum Der hier eingeführte Segmentierungsalgorithmus besitzt sieben algorithmenspezifische Parameter und zusätzlich vier allgemeine Level-Set-Parameter. Einen solch hochdimensionalen Parameterraum mit einer ausreichend hohen Dichte abzutasten, um sicher ein globales Optimum zu identifizieren, ist kaum möglich.

Darüber hinaus verhalten sich die Parameter nicht orthogonal zueinander. Geeignete Beispiele hierfür sind die in Abschnitt 5.4.3 dargestellte Abhängigkeit zwischen der Kontrasterhöhung und dem optimalen Zielwert, oder auch das Verhalten des Segmentierers bei der Kombination von orthogonaler und paralleler Glättung (Abschnitt 5.4.2). Daher musste sich diese Arbeit auf eine darstellende Beschreibung des Parameterverhaltens beschränken.

Referenz Die Segmentierungsqualität wurde mit dem Dice-Koeffizienten bewertet um einen objektiven und einheitlichen Vergleichswert zu erhalten. Die Bewertung des Segmentierungsergebnisses gegenüber einer manuell segmentierten Referenz impliziert jedoch, dass diese auch hinreichend gut ist. Bei Überlappungsmaßen auf einem sehr hohen Niveau von teilweise über 95% müsste aber auch die Variabilität von mehrfach manuell durchgeführten Segmentierungen einer Person (intra-rater-variability) und die Segmentierungen mehrerer Personen (inter-rater-variability) in Betracht gezogen werden. Ein passender Ansatz wird beispielsweise von Warfield et al. in [War04] geliefert, der mehrere Referenzsegmentierungen berücksichtigt und diese Ergebnisse unter anderem nutzt, um die Leistungsfähigkeit eines Segmentierers zu bewerten.

Limitierungen Hell-dunkel-hell-Kontraste mit einer geringen örtlichen Ausdehnung führen bei dem hier vorgeschlagenen Segmentierer eher zu Leakage, als hell-dunkel-Kontraste. Für einen Ansatz, der lediglich das lokale Intensitätsminimum und -maximum auswertet, sollte es jedoch keine Rolle spielen, wie groß der Bereich ist, auf dem die Extrema erreicht werden. Es bleibt abzuklären ob beispielsweise die lineare Interpolation der auf der Konturnormalen gesampelten Intensitätswerte diesen Effekt begünstigt.

Daher ist insbesondere bei den verwendeten Datensätzen mit einer niedrigeren örtlichen Auflösung eine Glättung in Konturnormalenrichtung möglichst klein zu wählen. Ein hell-dunkel-hell-Kontrast auf einer kurzen Distanz wird von einem Medianfilter als Ausreißer gewertet und folglich eliminiert. Solche Kontraste sind in oberen Bereichen des cerebralen Cortex zu erwarten, in denen Gehirngewebe an morphologische Strukturen mit einem ebenfalls hellen Bildsignal grenzt.

Stark gekrümmte Bereiche, welche in einer kleinen, örtliche Ausdehnung mehr als einen Halbkreis bzw. eine Halbkugel beschreiben, können dazu führen, dass der Suchbereich für das Minimum und das Maximum auf der Konturnormalen in beiden Richtungen außerhalb des Objektes liegt. Hierdurch kann die Evolution der Kontur in solche Bereiche hinein vorzeitig stoppen. In Rattenhirnen finden

sich Strukturen dieser Form im Bereich des an das Cerebellum angrenzende Paraflocculus.

Der kontrastbasierte Ansatz führt beim Skull-Stripping von Rattenhirn-MRTs zu guten Ergebnissen. Dies begründet sich auch in dem wenig kontrastreichen Gehirngewebe, so ist die Rattenhirnkontur deutlich durch einen lokal hohen Kontrast zu identifizieren. Der Ansatz wird jedoch in dem Augenblick fehlschlagen, wenn ein Objekt mit einem kontrastreichen Innenbereich segmentiert werden soll.

Stärken Die Versuche zur Abhängigkeit des Segmentierungsergebnisses von der Startkontur haben gezeigt, dass diese nicht komplett im Innern des zu segmentierenden Objektes liegen muss. In einem begrenzten Bereich besitzt der Segmentierer die Fähigkeit, die Kontur auch von außen an die Objektgrenze anzunähern.

Die lokal adaptive Bestimmung der Geschwindigkeitsfunktion liefert einen hohen Grad an Flexibilität im Vergleich zur globalen Auswertung der Geschwindigkeitsfunktion vor der Konturevolution (siehe beispielsweise [Cas97]). Auf diese Weise können lokale Signaladaptionen gezielt das Verhalten der Kontur steuern.

6.2 Ausblick

Diese Arbeit lässt noch Raum für weitere Untersuchungen. Darin könnte beispielsweise evaluiert werden, inwiefern Datensätze in Problemklassen zusammengefasst werden können. Scanner und Aufnahmeparameter könnten in die Klassifizierung einfließen. Gelingt dies, würde sich der Parameterraum auf die Wahl der korrekten Problemklasse reduzieren und das Segmentierungsverfahren wäre leichter zu handhaben.

Die hier vorgestellte Signaladaption hat das Segmentierungsergebnis verbessert, indem das in der Geschwindigkeitsfunktion ausgewertete Signal, durch lokale Bildeigenschaften gesteuert, verändert wurde. Es wäre auch denkbar, nicht das Signal, sondern die Parameter der Geschwindigkeitsfunktion in Abhängigkeit dieser Eigenschaften zu variieren. Alternativ kann auch die Glättung nur in Bereichen eingesetzt werden, welche ein schlechtes Signal-zu-Rauschverhältnis aufweisen.

Darüber hinaus ist die Einbindung eines Formmodells wie beispielsweise von Leventon [Lev00] in die Konturpropagation denkbar. Hierzu sollte die Anzahl der manuell segmentierten Datensätze für eine hinreichend hohe statistische Aussagekraft jedoch erhöht werden. Auftretendes Leakage könnte so verhindert werden. Dieses Ziel wäre erstrebenswert, da auf eine automatische Leakagedetektion, wie sie von Zhuang et al. [Zhu06] vorgeschlagen worden ist, verzichtet werden musste. Um dennoch Leakage schon während des Segmentierungsprozesses detektieren zu können, kann die Beobachtung des RMS-Wertes aufschlussreich sein. Bei Datensätzen ein und derselben Problemklasse sollte das Konvergenzverhalten in einem vorgegebenen Toleranzbereich ähnlich aussehen.

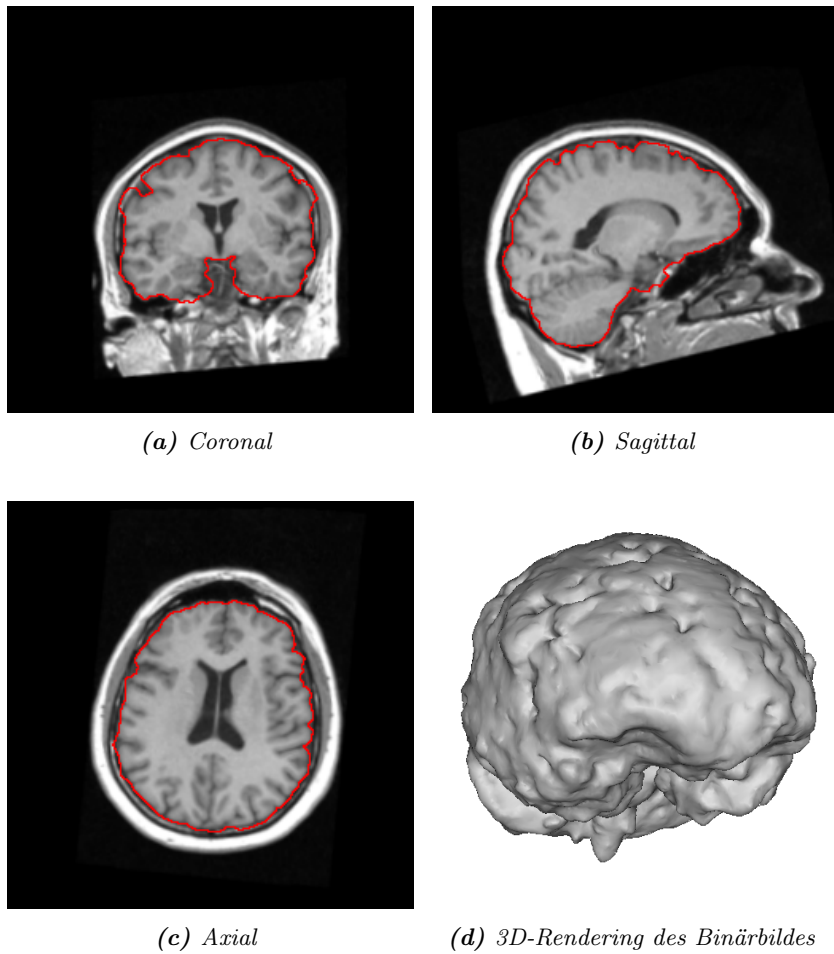


Abbildung 6.1: Segmentierungsergebnis eines Humanhirns (vgl. Abb. 4.1, S. 34). Segmentierung mittels eines echt-dreidimensionalen Ansatzes mit Signaladaption ($d_m = 8$, $d_M = 16$, $T = 0,5$, $r_M = 0,5$, $w = 6$ und $d_{\text{Var}} = 6$).

Der kontrastbasierte Segmentierungsansatz hat seine Ursprünge im Problemfeld des Skull-Stripping von Menschenhirn-MRTs. Die in dieser Arbeit vorgenommenen Modifikationen sollten somit auch wieder zurück in dieses Feld übertragen werden können, so dass mit geeigneter Parameteradaption auch die Segmentierung von Menschenhirn-MRTs möglich ist. Erste Tests haben diese Möglichkeit bereits bestätigt (Abbildung 6.1).

Anhang A

Herleitungen

A.1 Krümmung 3D

Eine ausführliche Herleitung der Berechnung der Krümmung wird in der gängigen Literatur zu Level-Sets meist weggelassen. Daher soll es an dieser Stelle einmal exemplarisch für den dreidimensionalen Fall durchgeführt werden.

Die Krümmung κ ist definiert als die Divergenz der Einheitsnormalen \vec{N} der Level-Set-Funktion ϕ , welche sich mit Hilfe des Skalarproduktes mit dem Nabla-Operator berechnen lässt.

$$\begin{aligned}\kappa &= \operatorname{div}(\vec{N}) \\ &= \nabla \cdot \vec{N} \\ &= \nabla \cdot \frac{\nabla \phi}{|\nabla \phi|}\end{aligned}\tag{A.1}$$

Dabei ist im dreidimensionalen Fall die 2-Norm der Level-Set-Funktion gegeben durch

$$|\phi| = \sqrt{\phi_x^2 + \phi_y^2 + \phi_z^2}\tag{A.2}$$

Zu beachten ist, dass sich der Bruch in der letzten Zeile von (A.1) als Multiplikation eines skalaren Feldes $\frac{1}{|\nabla \phi|}$ mit einem Vektorfeld $\nabla \phi$ betrachten lässt. In diesem Fall greift die Produktregel:

$$\begin{aligned}\kappa &= \nabla \cdot \left(\frac{1}{|\nabla \phi|} \nabla \phi \right) \\ &= \frac{1}{|\nabla \phi|} \nabla \cdot \nabla \phi + \nabla \phi \cdot \nabla \left(\frac{1}{|\nabla \phi|} \right) \\ &= \frac{\Delta \phi}{|\nabla \phi|} + (\phi_x, \phi_y, \phi_z) \cdot \begin{pmatrix} \frac{\partial}{\partial x} \frac{1}{|\nabla \phi|} \\ \frac{\partial}{\partial y} \frac{1}{|\nabla \phi|} \\ \frac{\partial}{\partial z} \frac{1}{|\nabla \phi|} \end{pmatrix}\end{aligned}\tag{A.3}$$

Dabei ist Δ der Laplace-Operator mit $\Delta \phi = \phi_{xx} + \phi_{yy} + \phi_{zz}$.

Die partiellen Ableitungen des rechten Spaltenvektors berechnen sich wie folgt:

$$\begin{aligned}\frac{\partial}{\partial x} \frac{1}{|\nabla\phi|} &= \frac{\partial}{\partial x} (\phi_x^2 + \phi_y^2 + \phi_z^2)^{-\frac{1}{2}} \\ &= -\frac{1}{2} (\phi_x^2 + \phi_y^2 + \phi_z^2)^{-\frac{3}{2}} (2\phi_x\phi_{xx} + 2\phi_y\phi_{xy} + 2\phi_z\phi_{xz})\end{aligned}\quad (\text{A.4})$$

Die entsprechenden partiellen Ableitungen in y - und z -Richtung ergeben sich analog. Setzt man nun das Ergebnis aus (A.4) in (A.3) ein, so folgt:

$$\begin{aligned}\kappa &= \frac{\Delta\phi}{|\nabla\phi|} + (\phi_x, \phi_y, \phi_z) \cdot \begin{pmatrix} -\frac{1}{|\nabla\phi|^3} \cdot (\phi_x\phi_{xx} + \phi_y\phi_{xy} + \phi_z\phi_{xz}) \\ -\frac{1}{|\nabla\phi|^3} \cdot (\phi_x\phi_{xy} + \phi_y\phi_{yy} + \phi_z\phi_{yz}) \\ -\frac{1}{|\nabla\phi|^3} \cdot (\phi_x\phi_{xz} + \phi_y\phi_{yz} + \phi_z\phi_{zz}) \end{pmatrix} \\ &= |\nabla\phi|^{-3} (\Delta\phi|\nabla\phi|^2 - \phi_x(\phi_x\phi_{xx} + \phi_y\phi_{xy} + \phi_z\phi_{xz}) \\ &\quad - \phi_y(\phi_x\phi_{xy} + \phi_y\phi_{yy} + \phi_z\phi_{yz}) \\ &\quad - \phi_z(\phi_x\phi_{xz} + \phi_y\phi_{yz} + \phi_z\phi_{zz}))\end{aligned}\quad (\text{A.5})$$

Durch Ausmultiplizieren gelangt man zur bekannten Berechnungsvorschrift für die Krümmung:

$$\kappa = \frac{\phi_x^2\phi_{yy} + \phi_x^2\phi_{zz} + \phi_y^2\phi_{xx} + \phi_y^2\phi_{zz} + \phi_z^2\phi_{xx} + \phi_z^2\phi_{yy} - 2\phi_x\phi_y\phi_{xy} - 2\phi_x\phi_z\phi_{xz} - 2\phi_y\phi_z\phi_{yz}}{|\nabla\phi|^3}\quad (\text{A.6})$$

Die Krümmung für den zweidimensionalen Fall ergibt sich direkt, wenn man die Level-Set-Funktion in der z -Richtung auf Null setzt, wodurch auch die partiellen Ableitungen in z -Richtung zu Null werden. Die entsprechenden Terme fallen damit weg und es bleibt:

$$\kappa = \frac{\phi_x^2\phi_{yy} + \phi_y^2\phi_{xx} - 2\phi_x\phi_y\phi_{xy}}{|\nabla\phi|^3}\quad (\text{A.7})$$

Anhang B

Zusatzmaterial

B.1 Ergänzungen zum Upwind-Schema

Dieser Abschnitt verdeutlicht den Stabilitätsgewinn durch die Verwendung des Upwind-Schemas bei der Lösung einer partiellen Differentialgleichung.

Dazu wird der eindimensionale Fall einer Level-Set-Funktion $\phi(x)$ und deren diskrete Variante $\Phi(k_x)$ mit einer Abtastrasterweite $h_x = 1$ betrachtet. Diese soll allein durch Propagation gemäß (3.21) variiert werden. Die Differentialgleichung lautet in umgestellter Form:

$$\frac{\partial \phi}{\partial t} = -\beta S(\vec{x}) |\nabla \phi| \quad (\text{B.1})$$

Mögliche Diskretisierungen ergeben sich durch die Verwendung von Vorwärts- oder Rückwärtsdifferenzen zur Berechnung des Gradienten der Funktion und einer einfachen finiten Differenz in zeitlicher Richtung:

$$\Phi_{D^{+x}}(k_x, t + \Delta t) := \Phi(k_x, t) - \beta S(k_x) |D^{+x} \Phi(k_x)| \Delta t \quad (\text{B.2})$$

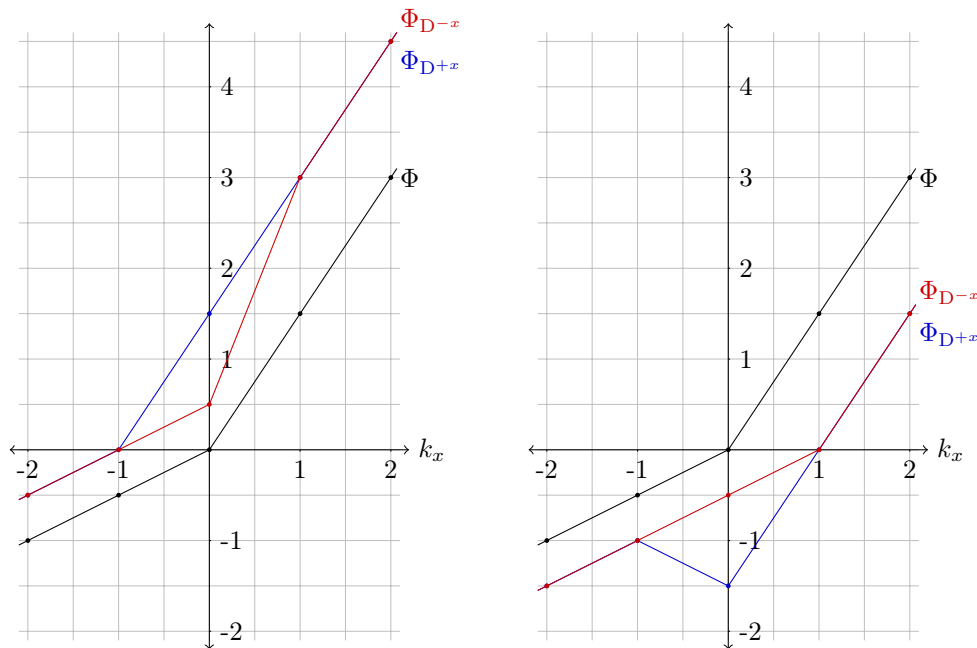
$$\Phi_{D^{-x}}(k_x, t + \Delta t) := \Phi(k_x, t) - \beta S(k_x) |D^{-x} \Phi(k_x)| \Delta t \quad (\text{B.3})$$

Der Einfachheit halber wird zunächst eine konstante Propagation mit

$$-\beta S(k_x) = 1 \quad (\text{B.4})$$

angenommen. Daraus folgt, dass die Zeitschrittgröße gemäß (3.28) den Wert $\Delta t = 1$ annehmen darf.

Abbildung B.1(a) zeigt exemplarisch eine Level-Set-Funktion und deren zeitliche Entwicklung, welche zu Visualisierungszwecken eben keine vorzeichenbehaftete Distanzfunktion ist. Die Level-Set-Funktion wird gemäß (B.4) in ihrer Wertigkeit nach oben gedrückt. Eine Erhöhung der Wertigkeit der Level-Set-Funktion entspricht, wie in Abschnitt 3.2.2 näher beschrieben, einer Verschiebung nach innen – im hier dargestellten Fall also nach links. Die lokale Veränderung der Funktion ist nur noch von dem approximierten Gradienten abhängig, welcher im linken Teil der Funktion $\frac{1}{2}$ und im rechten $\frac{3}{2}$ beträgt (Gleichungen (B.2) und (B.3)). Bei dieser Bewegungsrichtung liefert nur die Vorwärtsdifferenz das gewünschte Ergebnis.



(a) Bewegungsrichtung von rechts nach links, bedingt durch $-\beta S(k_x) = 1$. Die Vorwärtsdifferenz liefert das gewünschte Ergebnis, die Rückwärtsdifferenz produziert Instabilitäten.

(b) Bewegungsrichtung von links nach rechts, bedingt durch $-\beta S(k_x) = -1$. Die Rückwärtsdifferenz liefert das gewünschte Ergebnis, die Vorwärtsdifferenz produziert Instabilitäten.

Abbildung B.1: Visualisierung des Upwind-Schemas. Abhängig von der Entwicklungsrichtung der Level-Set-Funktion Φ (schwarze Kurve) muss die entsprechende Differenz (Vorwärtsdifferenz Φ_{D+x} blaue Kurve, Rückwärtsdifferenz Φ_{D-x} rote Kurve) zur Berechnung des Gradienten herangezogen werden.

Die Level-Set-Funktion bewegt sich in die entgegengesetzte Richtung, wenn die Propagationsgeschwindigkeit (B.4) negiert wird.

$$\beta S(k_x) = 1 \quad (\text{B.5})$$

Diesen Fall zeigt Abbildung B.1(b).

Es bleibt also festzuhalten, dass für das Upwind-Schema immer die Differenz genutzt wird, welche in die Richtung blickt, aus der die Funktion kommt.

Anhang C

Symbole

Symbol	Bedeutung
$I(\vec{x})$	Bildsignal
$\mathcal{C}(s)$	Über s parametrisierte Kurve
E	Energierterm bei der Entwicklung einer Snake
\vec{N}	Nach innen gerichtete Einheitsnormale einer Kurve \mathcal{C}
$g(\nabla I)$	Funktion zur Kantendetektion im Bild I
δ	Dirac-Funktional
δ_ϵ	Weichgezeichnetes Dirac-Funktional
H	Heavyside-Funktion
H_ϵ	Weichgezeichnete Heavyside-Funktion
ϕ	Kontinuierliche Level-Set-Funktion
Φ	Diskrete Level-Set-Funktion
κ	Krümmung
$\partial\Omega$	Kontur durch Level-Set-Funktion
Ω_-	Innenbereich
Ω_+	Außenbereich
\vec{N}	Nach außen gerichtete Normale der Level-Set-Funktion
$D(\vec{x})$	Distanzfunktion zu einer Kontur $\partial\Omega$
$\vec{V}(\vec{x})$	Advektion bestimmendes Vektorfeld
$S(\vec{x})$	Propagation bestimmende Geschwindigkeitsfunktion
α	Advektionsgewicht
β	Propagationsgewicht
γ	Krümmungsgewicht
h_x	Abtastrasterweite in x -Richtung
Δt	Zeitschrittgröße
L_0	Aktive Menge der Sparse-Field-Level-Sets
L_i	An L_0 angrenzende Mengen
RMS_{\max}	Konvergenzkriterium (root-mean-square)
n_{\max}	Abbruchkriterium durch maximale Iterationszahl
n_I	Iteration
$I_{\min}(\vec{x}, \vec{N})$	Lokales Intensitätsminimum auf \vec{N}
$I_{\max}(\vec{x}, \vec{N})$	Lokales Intensitätsmaximum auf \vec{N}
$I_{r,\min}(\vec{x}, \vec{N})$	Robustes Intensitätsminimum auf \vec{N}

$I_{r,\max}(\vec{x}, \vec{N})$	Robustes Intensitätsmaximum auf \vec{N}
$I_{l,\min}(\vec{x}, \vec{N})$	Rein lokales Intensitätsminimum auf \vec{N}
$I_{l,\max}(\vec{x}, \vec{N})$	Rein lokales Intensitätsmaximum auf \vec{N}
d_m	Distanz zur Ermittlung von I_{\min} , $I_{r,\min}$ bzw. $I_{l,\min}$
d_M	Distanz zur Ermittlung von I_{\max} , $I_{r,\max}$ bzw. $I_{l,\max}$
h_c	Kumuliertes Histogramm der relativen Häufigkeiten
$h_{c,0}$	Kumuliertes Histogramm innerhalb des Zero-Level-Set
$h_{2\%}$	2%-Schwellen in h_c
h_{med}	Median von h_c
T	Kontrastbestimmter Zielwert T
Var_a	Nach außen ermittelte, gleitende Varianz
Var_i	Nach innen ermittelte, gleitende Varianz
$\text{Median}(I_{\vec{x},\vec{N}}, r_M)$	Auf der Länge r_M bestimmter Median
r_M	Radius zur lokalen Medianbestimmung
d_{Var}	Distanz zur Varianzbestimmung
w	Varianzgewicht
$I'_{\vec{x},\vec{N}}$	Lokales Signal nach Signaladaption
$I^o_{\vec{x},\vec{N}}$	Orthogonal zu \vec{N} geglättetes Signal
<hr/>	
M_G	Konturpropagation mit globaler Medianauswertung
M_L	Konturpropagation mit rein lokaler Medianauswertung
G_{\parallel}	M_L mit paralleler Signalglättung
G_{\perp}	M_L mit orthogonaler Signalglättung
$G_{\parallel,\perp}$	M_L mit paralleler und orthogonaler Signalglättung
$G_{\parallel,\perp,w}$	$G_{\parallel,\perp}$ mit Signalaufsteilung
<hr/>	
∇	Nabla-Operator $\nabla = \left(\frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n} \right)$
Δ	Laplace-Operator $\Delta = \left(\frac{\partial^2}{\partial x_1^2}, \dots, \frac{\partial^2}{\partial x_n^2} \right)$
div	Divergenzoperator $\text{div}(u) = \sum_{i=1}^n \frac{\partial}{\partial x_i} u_i$
$\text{dim}(I)$	Dimension des Bildes I
D^{+x}	Vorwärtsdifferenz
D^{-x}	Rückwärtsdifferenz
D^{0x}	Zentrale Differenz

Literaturverzeichnis

- [Ada95] D. Adalsteinsson and J. A. Sethian. A fast level set method for propagating interfaces. *Journal of Computational Physics*, 118:269–277, 1995.
- [Atk98] M. S. Atkins and B. T. Mackiewicz. Fully automatic segmentation of the brain in MRI. *IEEE Transactions on Medical Imaging*, 17:98–107, 1998.
- [Bai01] C. Baillard, P. Helliera, and C. Barillot. Segmentation of brain 3D MR images using level sets and dense registration. *Medical Image Analysis*, 5:185–194, 2001.
- [Bro04] T. Brox and J. Weickert. *Pattern Recognition, Level Set Based Image Segmentation with Multiple Regions*, 415–423. Lecture Notes in Computer Science. Springer Berlin/Heidelberg, 2004.
- [Cas93] V. Caselles, F. Catt’e, T. Coll, and F. Dibos. A geometric model for active contours in image processing. *Numerische Mathematik*, 66:1–31, 1993.
- [Cas97] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22:61–79, 1997.
- [Cha01] T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001.
- [Che05] L. Cheng, J. Yang, X. Fan, and Y. Zhu. *Information Processing in Medical Imaging, A Generalized Level Set Formulation of the Mumford-Shah Functional for Brain MR Image Segmentation*, 418–430. Lecture Notes in Computer Science. Springer Berlin/Heidelberg, 2005.
- [Cio04] C. Ciofalo. *Medical Image Computing and Computer-Assisted Intervention, Atlas-Based Segmentation Using Level Sets and Fuzzy Labels*, 310–317. Lecture Notes in Computer Science. Springer Berlin/Heidelberg, 2004.
- [Cio05] C. Ciofalo and C. Barillot. *Information Processing in Medical Imaging, Brain Segmentation with Competitive Level Sets and Fuzzy Control*, 333–344. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2005.
- [Dal99] A. M. Dale, B. Fischl, and M. I. Sereno. Cortical surface-based analysis: I. segmentation and surface reconstruction. *NeuroImage*, 9:179–194, 1999.

- [Dua05] V. Duay, N. Houhou, and J.-P. Thiran. Atlas-based segmentation of medical images locally constrained by level sets. In *IEEE International Conference on Image Processing*, II-1286–9, 2005.
- [Hah00] H. K. Hahn and H.-O. Peitgen. *Medical Image Computing and Computer-Assisted Intervention*, The Skull Stripping Problem in MRI Solved by a Single 3D Watershed Transform, 134–143. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2000.
- [Iba03] L. Ibanez, W. Schroeder, L. Ng, and J. Cates. *The ITK Software Guide*. Kitware Inc., 2003.
- [Jä02] B. Jähne. *Digitale Bildverarbeitung*. Springer Berlin/Heidelberg, 2002.
- [Kas88] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1:321–331, 1988.
- [Lap02] R. J. Lapeer, A. C. Tan, and R. Aldridge. A combined approach to 3D medical image segmentation using marker-based watersheds and active contours: the active watershed method. In *Medical Image Understanding and Analysis*, 2002.
- [Lev00] M. E. Leventon, E. Grimson, and O. Faugeras. Statistical shape influence in geodesic active contours. *Computer Vision and Pattern Recognition*, 01:1316, 2000.
- [Mal93] R. Malladi, J. A. Sethian, and B. C. Vemuri. A topology independent shape modeling scheme. *Proceedings of the SPIE*, .2031:246–258, 1993.
- [Mal96] R. Malladi, J. A. Sethian, and B. C. Vemuri. A fast level set based algorithm for topology-independent shape modeling. *Journal of Mathematical Imaging and Vision*, 6:269–289, 1996.
- [Osh88] S. Osher and J. A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations. *Journal of Computational Physics*, 79:12–49, 1988.
- [Osh02] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*. Springer, Berlin / Heidelberg, 2002.
- [Pal08] C. Palm and U. Pietrzyk. Time-dependent joint probability speed function for level-set segmentation of rat brain slices. In *Medical Imaging 2008: Image Processing. Proceedings of the SPIE*, 69143U–69143U–8, 2008.
- [Pax98] G. Paxinos and C. Watson. *The Rat Brain in Stereotaxic Coordinates*. Academic Press, 1998.
- [Pen04] G. P. Penney, J. A. Schnabel, D. Rueckert, M. A. Viergever, and W. J. Niessen. Registration-based interpolation. *IEEE Transactions on Medical Imaging*, 23:922–926, 2004.

- [Per90] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:629–639, 1990.
- [Seg04] F. Segonne, A. M. Dale, E. Busa, M. Glessner, D. Salat, H. K. Hahn, and B. Fisch. A hybrid approach to the skull stripping problem in MRI. *NeuroImage*, 22:1060 – 1075, 2004.
- [Ser06] J. Serra. A lattice approach to image segmentation. *Journal of Mathematical Imaging and Vision*, 24:83–130, 2006.
- [Set99] J. A. Sethian. *Level Set Methods and Fast Marching Methods*. Cambridge University Press, 2nd edition, 1999.
- [Smi02] S. M. Smith. BET: Brain extraction tool. Technical report, Oxford Centre for Functional MRI of the Brain (FMRIB), 2002.
- [Sze93] R. Szeliski, D. Tonnesen, and D. Terzopoulos. Modeling surfaces of arbitrary topology with dynamic particles. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 82–87, 1993.
- [Ter86] D. Terzopoulos. Regularization of inverse visual problems involving discontinuities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8:413–242, 1986.
- [War04] S. K. Warfield, K. H. Zou, and W. M. Wells. Simultaneous truth and performance level estimation (STAPLE): An algorithm for the validation of image segmentation. *IEEE Transactions on Medical Imaging*, 23:903–921, 2004.
- [Whi98] R. T. Whitaker. A level-set approach to 3D reconstruction from range data. *International Journal of Computer Vision*, 29:203–231, 1998.
- [Xu97] C. Xu and J. L. Prince. Gradient vector flow: A new external force for snakes. *Computer Vision and Pattern Recognition*, 00:66, 1997.
- [Xu98] C. Xu and J. L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Transactions on Image Processing*, 7:359–369, 1998.
- [Zha96] H.-K. Zhao, T. Chan, B. Merriman, and S. Osher. A variational level set approach to multiphase motion. *Journal of Computational Physics*, 127:179–195, 1996.
- [Zhu03] F. Zhu and J. Tian. Modified fast marching and level set method for medical image segmentation. *Journal of X-Ray Science and Technology*, 11:193–204, 2003.
- [Zhu06] A. H. Zhuang, D. J. Valentino, and A. W. Toga. Skull-stripping magnetic resonance brain images using a model-based level set. *NeuroImage*, 32:79–92, 2006.

Erklärungen

Eidesstattliche Erklärung

Ich versichere hiermit, die vorgelegte Arbeit in dem gemeldeten Zeitraum ohne fremde Hilfe verfasst und mich keiner anderen als der angegebenen Hilfsmittel und Quellen bedient zu haben.

Jülich, 1. Dezember 2008

Weitergabeerklärung

Ich erkläre hiermit mein Einverständnis, dass das vorliegende Exemplar meiner Diplomarbeit oder eine Kopie hiervon für wissenschaftliche Zwecke verwendet werden darf.

Jülich, 1. Dezember 2008

Danksagung

Zu guter Letzt möchte ich allen ganz herzlich danken, die diese Masterarbeit ermöglicht und zu ihrem Gelingen beigetragen haben ...

- ... Prof. Dr. Dietmar Kunz für die Einführung in faszinierende Gebiete der Bildverarbeitung, anregende Problemlösungsvorschläge und die Unterstützung bei Fragestellungen fachlicher und formaler Art,
- ... Dr. Christoph Palm für die hervorragende und kompetente Betreuung im Forschungszentrum und die motivierende, bereitwillige Hilfe bei allen aufgetretenen Problemen,
- ... Prof. Dr. Uwe Pietrzyk für die Unterstützung dieser Arbeit und interessante Fragestellungen zu ImageJ,
- ... der *gesamten* Arbeitsgruppe „Multimodale Bildverarbeitung“ für die sehr angenehme Zeit im Forschungszentrum und interessante Diskussionen,
- ... meinen Kommilitonen Michaela, Dirk, Borys, Matthias, Holger und Andy für alle Erfahrungen im Masterstudiengang und ihre stete Hilfsbereitschaft,
- ... meiner Familie für die enorme Geduld und Unterstützung in jeder Phase der Arbeit,
- ... Meike für ihr Vertrauen in meine Fähigkeiten.

Danke!