# Grenzkonturen aus Klassenbildern

G. Wagenknecht, D. Schmitz, U. Büll

Klinik für Nuklearmedizin
RWTH Aachen, Pauwelsstr. 30, 52057 Aachen
g.wagenknecht@nuk-gate.nukmed.rwth-aachen.de

*Zur quantitativen Analyse funktioneller Bilddaten des Gehirns (PET, SPECT) soll die korrelierte morphologische Modalität (MRT) der genauen anatomischen Zuordnung funktioneller Parameter dienen. Die kernspintomographischen Schnittbilder werden hierzu durch einen neuronalen Klassifikator in fünf Klassen segmentiert. Zur regionenorientierten Auswertung funktioneller Parameter mit der ECAT-Software[1] werden die Regionengrenzen als geschlossene Polygonzüge aus den Klassenbildern extrahiert. Jede Klasse wird separat bearbeitet. Die Analyse möglicher Regionentypen und deren Verbundenheit führt zur Berücksichtigung mehrfach zu durchlaufener Konturpunkte in einer Wertigkeitsmatrix. Die darauf basierende Konturverfolgung unterscheidet zwischen Außen- und Innenkonturen einer Region. Die Optimierung der Polygonstruktur dient der schnellen graphischen Überlagerung komplexer Regionengrenzen auf funktionelle Bilddaten.*

**Keywords:** Konturverfolgung, Klassifizierung, Quantifizierung, Morphologie, Funktion

## 1 Motivation

Im Rahmen des IZKF: ZNS wurde ein Projekt zur multimodalen Bildanalyse auf der Basis segmentierter MRT-Regionen definiert [2]. Ziel ist die quantitative Analyse funktioneller Parameter (PET, SPECT) des Gehirns innerhalb der auf Basis der korrelierten kernspintomographischen Bilddaten des Patienten segmentierten anatomischen Regionen (MRT). Der neuronale Klassifikator segmentiert hierzu die MRT-Schnittbilder in fünf Klassen (Hintergrund (1); Fettgewebe / Knochen / Hirnhäute (2); Liquorraum / Ventrikel (3); Graue Substanz: Kortex / Nucleus Caudatus / Putamen / Thalamus (4); Weiße Substanz (5)) (Abb.1).

Die statistische Auswertung funktioneller Parameter innerhalb sogenannter „Regions of Interest“ (ROI’s) mit der ECAT-Software basiert auf der Definition von ROI’s als geschlossene Polygonzüge der Regionaußenkonturen in einer speziellen Filestruktur. Die Extraktion dieser Grenzkonturen in Form von Polygonzügen aus den Klassenbildern ist somit erforderlich.

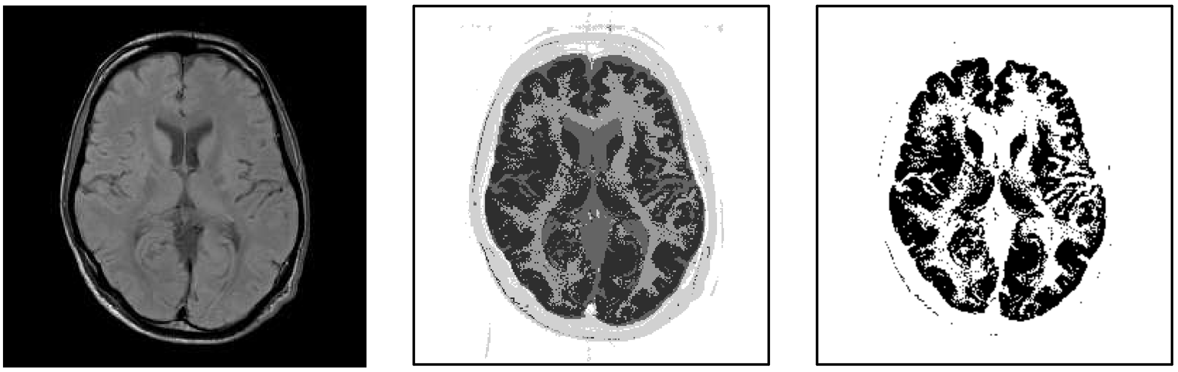

Abb.1: MRT-Schnittbild des Gehirns (a); Klassenbild (b); Klasse: Graue Substanz (c).

# 2 Problemstellung

## 2.1 Definitionen

**Definition 1:** Zwei Pixel werden direkte Nachbarn (*d-Nachbarn*) genannt, wenn sich beide Pixel eine gemeinsame Kante teilen, und indirekte Nachbarn (*i-Nachbarn*), wenn sie nur über eine Ecke miteinander verbunden sind. Die Bezeichnung *Nachbar* meint beide Arten von Nachbarschaft. Der Begriff *N-Nachbar* mit $0 \leq N \leq 7$ wird für die Bestimmung der Position N in Abb.2a verwendet [3].

Zu beachten ist, daß d-Nachbarn N-Nachbarn mit geradem N sind und bei i-Nachbarn N ungerade ist.

**Definition 2:** Ein *i-Pfad* ist eine Abfolge von Pixeln $A_1$, $A_2$, ..., $A_n$, so daß für alle k>1 $A_{k-1}$ i-Nachbar von $A_k$ ist und für alle k<n $A_{k+1}$ i-Nachbar von $A_k$ ist. Der Ausdruck *d-Pfad* bezeichnet Pfade, deren Pixel d-Nachbarn sein müssen. Ein *Pfad* kann sowohl i-Pfade als auch d-Pfade enthalten [3].

**Definition 3:** Eine gegebene Anzahl von Pixeln S ist *i-verbunden*, wenn für jedes Paar von Pixeln A und B aus S ein i-Pfad existiert, wobei A das erste und B das letzte Element ist und alle weiteren Pixel des Pfades in S liegen. Die Ausdrücke *d-verbunden* und *verbunden* sind entsprechend definiert [3] (Abb2.b-d).

**Definition 4:** Eine *i-Kontur* einer verbundenen Pixelmenge R ist definiert als Menge aller Pixel in R, die mindestens einen d-Nachbarn besitzen, der nicht in R liegt. I-Konturen können aus i- und d-Pfaden bestehen. Die d-Kontur von R ist die Menge aller Pixel in R, die mindestens einen Nachbarn besitzen, der nicht in R liegt. D-Konturen dürfen nur d-Pfade enthalten [3].

| | | |
|---|---|---|
| 3 | 2 | 1 |
| 4 | P | 0 |
| 5 | 6 | 7 |

Abb.2a: Position von Nachbarpixeln bezüglich P

Abb.2b: i-verbundene Pixel

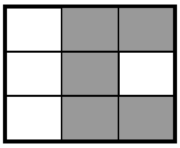

Abb.2c: d-verbundene Pixel

Abb.2d: verbundene Pixel

## 2.2 Regionentypen

Ein segmentiertes Klassenbild kann verschiedene Regionentypen (Abb.3) enthalten, deren Analyse zu einer Unterscheidung von Außen- und Innenkonturen von Regionen führt und zudem Grundlage für die Entscheidung ist, ob für die Konturverfolgung die i- oder d-Verbundenheit von Regionen berücksichtigt werden soll.

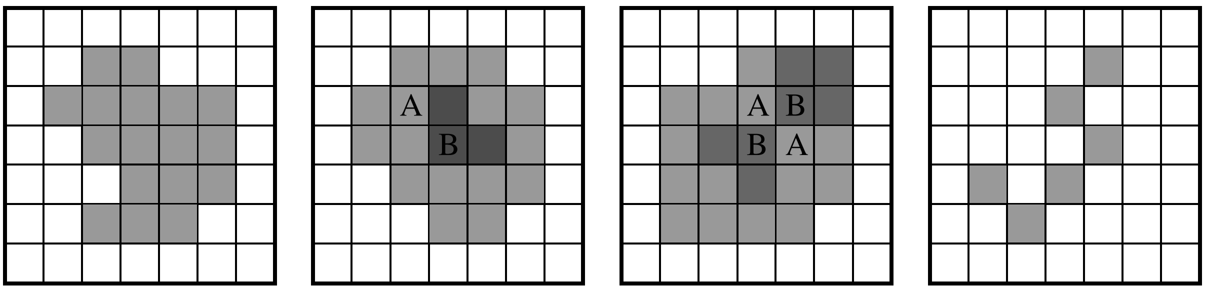


Abb.3 a-d: Beispiele für Regionentypen

Abb.3a zeigt ein Beispiel des einfachsten Regionentyps. Die Regionenkontur läßt sich als i- oder d-Kontur definieren.

Abb.3b zeigt eine Region, die durch zwei Konturen begrenzt ist. Dies ist immer dann der Fall, wenn eine Region der Klasse A eine Region der Klasse B vollständig umgibt. Man unterscheidet daher Außen- und Innenkonturen von Regionen. Die ECAT-Software kann nur Außenkonturen von Regionen verarbeiten, so daß bei der statistischen Analyse von Region A das durch die Außenkontur von Region B definierte Gebiet von dem durch die Außenkontur von Region A definierte Gebiet subtrahiert werden muß. In die ROI-Filestruktur der ECAT-Software dürfen somit nur Außenkonturen von Regionen aufgenommen werden.

Abb.3c zeigt ein Problem, das bei Berücksichtigung der i-Verbundenheit von Regionen auftreten kann. Region B liegt gleichzeitig innerhalb und außerhalb von Region A [4]. Für die quantitative Analyse auf Basis der ECAT-Software kann man dann Region B nicht einfach von Region A subtrahieren. Dieses Problem umgeht man, indem man nur d-verbundene Regionen betrachtet.

Abb.3d zeigt einen Nachteil, den die Betrachtung der d-Verbundenheit von Regionen mit sich bringt. Eine ausschließlich i-verbundene Pixelmenge wird nicht als zusammenhängende Region betrachtet. Jedes Pixel stellt eine eigene Region dar.

# 3 Algorithmus

Die Abwägung der durch die beschriebenen Regionentypen induzierten Problematiken führt zu der Entscheidung, nur die d-Verbundenheit von Regionen zu berücksichtigen und Konturen nur in Form von d-Konturen zu extrahieren. Bei der Konturverfolgung sollen Außen- und Innenkonturen unterschieden werden. Jede Klasse wird separat bearbeitet. Der Algorithmus gliedert sich in drei Teile. Im ersten Teil wird eine sogenannte Wertigkeitsmatrix zur Berücksichtigung mehrfach zu durchlaufener Konturpunkte erstellt, im zweiten Teil wird die eigentliche Konturverfolgung vollzogen und im dritten Teil werden die gewonnenen Polygone optimiert.

## 3.1 Wertigkeitsmatrix

Die durch eine Klasse definierte Klassenmatrix wird zunächst durch Betrachtung der acht nächsten Nachbarn jedes Pixels in eine Wertigkeitsmatrix umgewandelt. Die Wertigkeit gibt an, ob ein Pixel zur Kontur einer Region der betrachteten Klasse gehört und wie oft es von einem Polygonzug erfaßt werden soll.

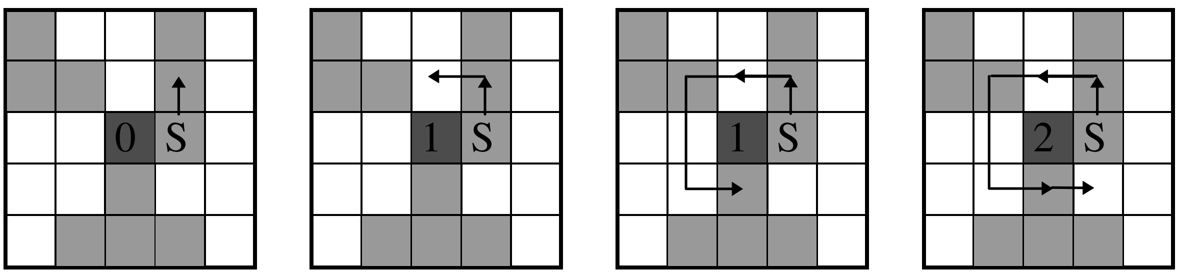


Abb.4: Bestimmung der Wertigkeit eines Pixels P: Startpunkt S ist d-Nachbar, Wertigkeit: 0 (a); Klasse wird verlassen, Wertigkeit: 1 (b); 6-Nachbar ist d-Nachbar: Wertigkeit: 1 (c); Klasse wird verlassen, Wertigkeit: 2 (d).

Die Bestimmung der Wertigkeit eines Pixels P erfolgt beginnend beim 0-Nachbarn als Startpunkt (Abb.4a) entgegen dem Uhrzeigersinn in der N-Nachbarschaft des Pixels. Wird ein zur Klasse gehörender d-Nachbar gefunden, im Beispiel der Abb.4 der 0-Nachbar und der 6-Nachbar (Abb.4a,c), so werden die folgenden N-Nachbarn auf Verbundenheit untersucht. Wird die betrachtete Klasse verlassen, so wird die Wertigkeit des Pixels P um eins erhöht. (Abb.4b,d).

Ein Wert größer null in der Wertigkeitsmatrix bedeutet, daß der Eintrag in der Matrix zu einer Regionengrenze gehört. Einen Sonderfall bilden 1-Pixel große Regionen, die zu der betrachteten Klasse gehören. Sie erhalten ebenfalls die Wertigkeit eins (Abb.5a).

Ein Wert von zwei bedeutet, daß diese Koordinate zweimal erfaßt werden muß. Werte größer eins entstehen z.B. durch Schleifen, oder durch Überlagerung von Außen- und Innenkontur einer Region (Abb.5b). Werte größer zwei entstehen z.B. durch Verzweigungen (Abb.5c).

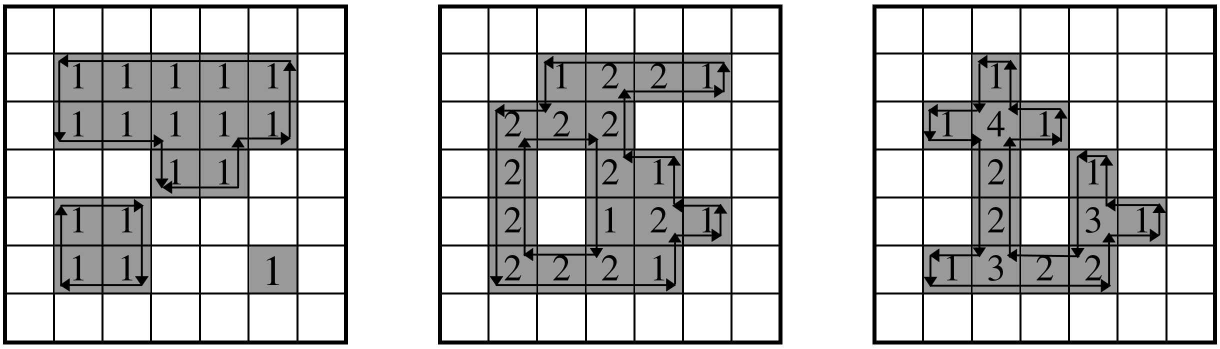

Abb.5: Wertigkeitsmatrix: Einträge 0-1 (a); Einträge 0-2 (b); Einträge 0-4 (c).

## 3.2 Konturverfolgung

Um die Startposition eines Polygonzuges zu finden, wird zunächst die Wertigkeitsmatrix zeilenweise von links nach rechts nach einem Eintrag größer null durchsucht. Der erste Eintrag größer null definiert die Startkoordinaten (x,y) des Polygonzuges. Die Wertigkeitsmatrix wird an dieser Position um eins erniedrigt.

Bei der Konturverfolgung soll zwischen Außen- und Innenkonturen einer Region unterschieden werden. Während Außenkonturen gegen den Uhrzeigersinn abgearbeitet werden, verfolgt man Innenkonturen im Uhrzeigersinn. Daher wird zunächst untersucht, zu welcher Art das gefundene Startpixel gehört. Man betrachtet dazu den Eintrag im Klassenbild an der Position (x-1,y). Ist der Eintrag an dieser Position ungleich der betrachteten Klasse, so gehört die Position (x,y) zu einer Außenkontur. Ist der Eintrag an der Position (x-1,y) im Klassenbild gleich der betrachteten Klasse, so befindet man sich innerhalb einer Region und die Position (x,y) gehört zu einer Innenkontur.

Wie oben erläutert, soll durch die Konturverfolgung ein d-Pfad erzeugt werden. Bei der Verfolgung einer Außenkontur werden zunächst in der Wertigkeitsmatrix die d-Nachbarn des Startpunktes P in der Reihenfolge 2, 4, 6, 0 auf einen Eintrag größer null untersucht. Wird ein solcher Eintrag gefunden, so wird der Eintrag in der Wertigkeitsmatrix an der gefundenen Position um eins erniedrigt und die Koordinaten der neuen Position in die Polygonliste aufgenommen. Nun werden in der Wertigkeitsmatrix die d-Nachbarn der neuen Position untersucht. In die Betrachtung der Umgebung fließt dabei zusätzlich ein, aus welcher Richtung die aktuelle Position erreicht wurde. Erreicht man das aktuelle Pixel z.B. über den 2-Nachbarn, so müssen die d-Nachbarn des untersuchten Pixels in der Reihenfolge 4, 6, 0, 2 nach Einträgen größer null untersucht werden (Abb.6a). Kommt man z.B. vom 6-Nachbarn, so werden die d-

Nachbarn in der Reihenfolge 0, 2, 4, 6 untersucht (Abb.6b). Wird in der d-Nachbarschaft ein Eintrag größer null gefunden, so wird dieser in der Wertigkeitsmatrix um eins erniedrigt und das neue Pixel in die Polygonliste aufgenommen u.s.w.

Bei der Innenkonturverfolgung werden die d-Nachbarn des untersuchten Pixels P in umgekehrter Reihenfolge betrachtet. Die d-Nachbarn des Startpunktes werden in der Reihenfolge 6, 4, 2, 0 untersucht. Die Reihenfolge der Untersuchung der d-Nachbarn jedes anderen Konturpunktes hängt wiederum vom Vorgänger ab. Kommt man z.B. vom 2-Nachbarn, so ist die Reihenfolge 0, 6, 4, 2 (Abb.6c); kommt man vom 6-Nachbarn, ergibt sich die Reihenfolge 4, 2, 0, 6 (Abb.6d).

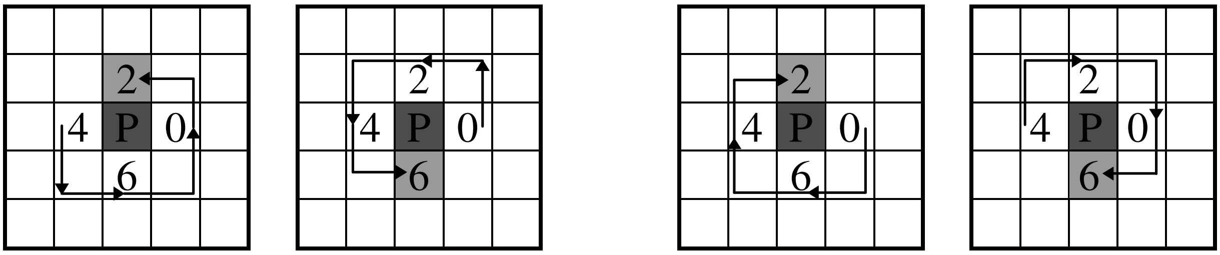


Abb.6: Beispiele für die Reihenfolge der Untersuchung der d-Nachbarn eines Konturpixels P. Außenkontur (a-b); Innenkontur (c-d).

Gelangt man durch die Konturverfolgung wieder zur Startposition oder werden an einer Position keine weiteren d-Nachbarn gefunden, so hat man einen geschlossenen Polygonzug extrahiert. Der geschlossene Polygonzug einer Außenkontur entspricht in der ROI-Filestruktur der ECAT-Software einer ROI.

Ausgehend von der Startposition der zuletzt extrahierten Kontur wird die Wertigkeitsmatrix nach einem neuen Startpunkt für einen neuen Polygonzug durchsucht.

Problematisch stellt sich die Detektion einer Innenkontur nach der erläuterten Methode dar, wenn sich Außen- und Innenkontur einer Region an der Startposition der Innenkonturverfolgung überlagern. Die Innenkontur wird dann als Außenkontur erkannt, weil der Eintrag an Position (x-1,y) ungleich der betrachteten Klasse ist. Dieses Problem wird gelöst, indem man bei der Außenkonturverfolgung für jeden Polygonpunkt (x,y) den Wert in der Klassenmatrix an der Position (x-1,y) auf den Wert der betrachteten Klasse setzt.

## 3.3 Optimierung

Die Optimierung der erhaltenen Polygonzüge basiert auf der Umwandlung aller d-Konturen (Abb.7a) in i-Konturen (Abb.7b). In einem zweiten Schritt werden dann nur solche Konturpunkte in die Polygonliste aufgenommen, die einer Richtungsänderung der i-Kontur entsprechen (Abb.7c). Die hierdurch erzielte Einsparung an Polygonpunkten entspricht für die in Abb.1c dargestellte Klasse der Grauen Substanz ungefähr 60%.

Durch die automatische Segmentierung kann es zu Fehlklassifikationen kommen, die sich z.B. als wenige Pixel große Regionen einer Klasse A im Gebiet einer sonst homogenen Region B auswirken. Um den Einfluß solcher Fehlklassifikationen zu vermindern, werden wenige Pixel große Regionen durch Vorgabe einer Konturmindestlänge von der Konturextraktion ausgeschlossen.

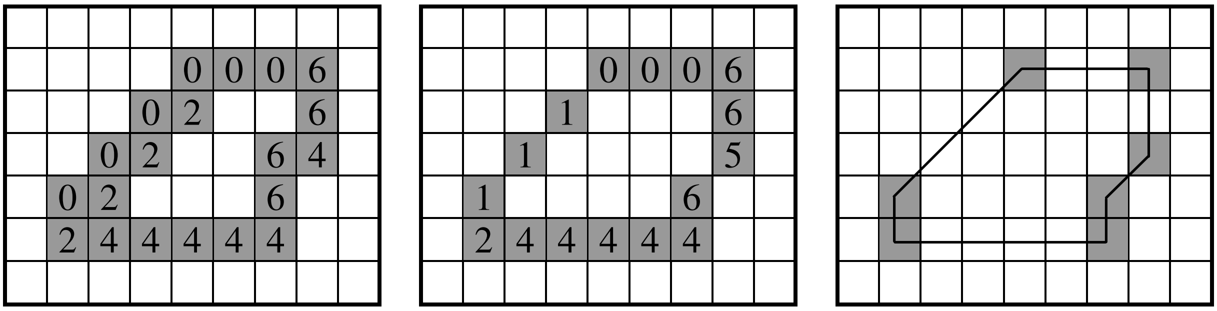

Abb.7: Regionengrenze als d-Kontur (a); Regionengrenze als i-Kontur (b); Polygonpunkte entsprechend der Chain-Code-Änderung der i-Kontur (c).

# 4 Ergebnis

Als Ergebnis der Konturverfolgung erhält man ein ROI-File, das die Außenkonturen aller Regionen aller segmentierten Klassen des bearbeiteten morphologischen Schnittbildes als geschlossene Polygonzüge beinhaltet. Die Klassenzugehörigkeit der Polygonzüge ist dabei in der ROI-Filestruktur als Header-Information zu jeder ROI gespeichert. In der ECAT-Software lassen sich nun die ROI's dem korrelierten funktionellen Schnittbild graphisch überlagern. Jede ROI läßt sich separat ansprechen und quantitativ auswerten. Das Beispiel einer solchen Überlagerung ist in Abb.8 dargestellt.

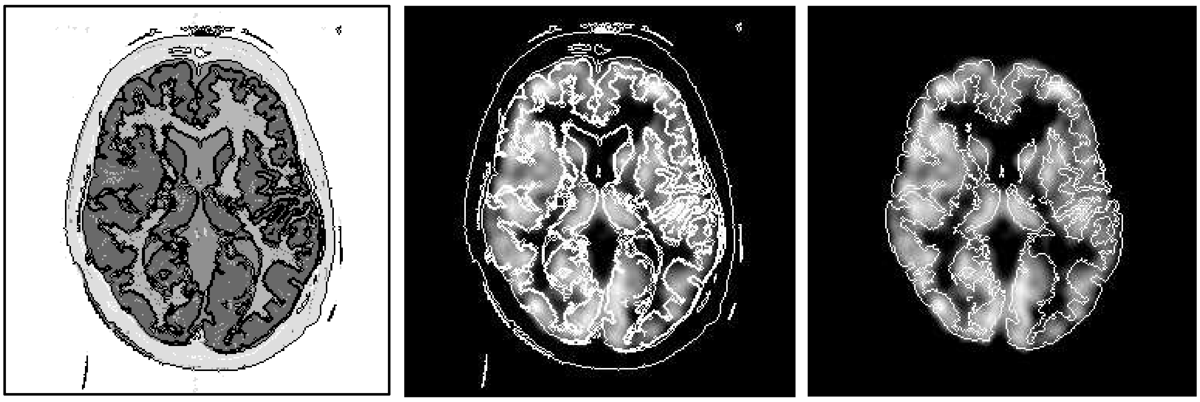
Abb.8: Klassenbild mit überlagerten Grenzkonturen (a); Korreliertes PET-Bild mit überlagerten Grenzkonturen (b); Korreliertes PET-Bild mit überlagerten Grenzkonturen der Klasse der Grauen Substanz (c).

# Literatur

[1] ECAT-Software, Release 6.4c, CTI Knoxville USA, 1994.

[2] Wagenknecht G, Büll U: Segmentierung und Korrelation multimodaler tomographischer Bilddaten des menschlichen Gehirns. Aachener Workshop: Bildverarbeitung für die Medizin, 1996.

[3] Pavlidis T: Algorithms for Graphics and Image Processing. Computer Science Press. Rockville, MD, 1982.

[4] Turner MJ, Wiseman NE: Efficent Lossless Image Contour Coding. Computer Graphics Forum, 15(2): 107-117, 1996.