

Konzept zur interaktiven Auswertung multidimensionaler biologischer Bilddaten

Zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
der Fakultät für Maschinenbau
Karlsruher Institut für Technologie (KIT)

genehmigte

Dissertation

von

Dipl.-Inform. Sebastian Pfeiffer

Tag der mündlichen Prüfung:	03. Juli 2013
Hauptreferent:	Prof. Dr.-Ing. habil. G. Bretthauer
Korreferent 1:	Prof. Dr.-Ing. habil. J. Wernstedt
Korreferent 2:	Prof. Dr.-Ing. habil. R. Mikut

Selbständigkeitserklärung

Hiermit versichere ich, Sebastian Pfeiffer, die vorliegende Arbeit selbständig und nur unter Verwendung der angegebenen Quellen angefertigt zu haben.

Karlsruhe, den 9. Juli 2013

Sebastian Pfeiffer

Danksagung

Für die Möglichkeit an einem sehr interessanten und umfangreichen Gebiet forschen zu können, möchte ich Herrn Prof. Dr.-Ing. habil. Georg Bretthauer herzlich danken. Stets war ich mir seiner Unterstützung sicher und fand an seinem Institut jederzeit ein offenes Ohr, um Probleme und angestrebte Lösungen konstruktiv zu diskutieren.

Weiterhin danke ich Herrn Prof. Dr.-Ing. habil. Jürgen Wernstedt für die Übernahme des Korreferats. Mit Interesse hat er den Fortschritt der Arbeit verfolgt und so zu deren Gelingen beigetragen.

Besonderer Dank gilt Herrn apl. Prof. Dr.-Ing. habil. Ralf Mikut, der mir seit meinen ersten Tagen am Institut durch angeregte Diskussionen eine hervorragende fachliche Betreuung zukommen ließ.

Für die Mithilfe bei der Akquisition von biologischen Daten möchte ich mich bei den Mitarbeitern des Instituts für Toxikologie und Genetik (ITG) bedanken. Besonders die enge Zusammenarbeit mit Herrn Dipl.-Biol. M.Sc. Jens Otte brachte mir einen tiefen Einblick in den biologischen Sektor sowie einen wichtigen Erfahrungsaustausch und eine interessante Zusammenarbeit. Gespräche nahmen freundschaftliche Züge an und weckten kindliche Neugier an den weiten Problemen aus dem Sektor der Bioinformatik bzw. der Analyse biologischer Datensätze.

Für wertvolle und interessante Diskussionen gilt mein Dank meinen Gruppenkollegen Christian Bauer, Oliver Schill, Markus Grube, Rüdiger Alshut und Markus Reischl sowie allen anderen Mitarbeitern des Instituts für Angewandte Informatik (IAI). Viele innovative Gedankenflüsse habe ich unseren Gesprächen zu verdanken.

Weiterhin danke ich allen Studenten, die im Rahmen von hilfswissenschaftlicher Arbeit, Studien- oder Diplomarbeiten einen Beitrag zu der vorliegenden Dissertationsschrift geleistet haben. Ich danke insbesondere Philip Beermann, Sarah Flade, Eric Heitz, Denise Kemler, Thomas Schenker, Christian Schmidt, Inga Shamkhalova, Martin Thoma und Emile de Weerd.

Im Allgemeinen möchte ich den Studenten und Doktoranden des KIT danken. Manche Gespräche brachten mich auf neue Gedanken und Ideen, was zu gemeinsamen Veröffentlichungen führte. Ich danke hiermit speziell Jan Calliess, Wolfgang Globke, Michael Mai aber auch Jessica Legradi, Lars Hegenbart, Per Niklas Hedde, Rebecca Klages und der Truppe der Doktoranden-Initiative.

Mein Gastaufenthalt in China im Jahr 2010 hat mir zusätzliche Denkanstöße und Ideen für die Promotion gegeben. Das KHYS (Karlsruher House of Young Scientists) hat mir die Reise nach China ermöglicht. Ich danke hiermit speziell Prof. Xiaojun Wu, Prof. Guocheng Du, Ass.-Prof. Pan Lingyun, Zhenhua Feng und vielen weiteren gastfreundlichen Menschen an der Jiangnan University (Wuxi, China).

Darüber hinaus möchte ich Prof. Dr. Ernst H.K. Stelzer am *Buchmann Institute for Molecular Life Sciences*, an der Goethe Universität in Frankfurt am Main, sehr danken. Der Austausch mit ihm hat mich in vielerlei Hinsicht gefördert. Ich habe mich dort sehr wohl gefühlt. Ich danke an dieser Stelle auch Alexander Atzberger, Alexander Schmitz, Berit Langer, Bo-Jui Chang, Christian Mattheyer, Daniel von Wangenheim, Francesco Pampaloni, Frederic Strobl, Heinz Strewe, Kai Schmale, Michaela Koller, Marko Reinhardt, Nariman Ansari, Roli Richa, Sabine Fischer, Sigrun Becker und Sonja Stamness.

Mein herzlicher Dank gilt schließlich meinen Eltern, meinen Großeltern sowie meinen Freundinnen und Freunden für Rückhalt und Verständnis. Besonders meiner Mutter und meiner Freundin möchte ich zu guter Letzt für ihre Unterstützung doppelt und dreifach danken. Sie waren mir stets eine Quelle der Inspiration und Anregung.

Karlsruhe, 9. Juli 2013

Sebastian Pfeiffer

Inhaltsverzeichnis

Danksagung	iii
1 Einleitung	1
1.1 Motivation	1
1.2 Darstellung des Entwicklungsstandes	2
1.2.1 Akquisition	2
1.2.2 Datenmanagement	5
1.2.3 Multidimensionale Bildanalyse	7
1.2.4 Offene Probleme	12
1.3 Ziele und Aufgaben	17
2 Neues effizientes multidimensionales Bearbeitungsverfahren	21
2.1 Problemdarstellung	21
2.2 Filterentwurf	22
2.2.1 Separierbare zweidimensionale Filter	24
2.2.2 Separierbare multidimensionale Filter	26
2.3 Simulationsergebnisse	30
2.3.1 Separiert-basierte Methode vs. Naive Methode	30
2.3.2 Experimentelles Vorgehen	30
2.3.3 Datensätze und Aufnahmen	31
2.3.4 Ergebnis	32
2.3.5 Vergleich mit FFT-basierten Konvolutionen	38
2.3.6 Separierung und Nicht-Separierung des Medianfilters	42
2.4 Auswertung	45
2.5 Anwendung multidimensionaler Filter	47
3 Neue Verknüpfung multidimensionaler Segmentierungsverfahren	49
3.1 Problemdarstellung	49
3.2 Lösungsverfahren	51
3.2.1 Evaluierung verschiedener Segmentierungen für Zellkerne	51
3.2.2 Verknüpfung mehrerer Segmentierungsansätze	56
3.3 Auswertung	60

4	Neuartige Merkmalsgewinnung multidimensionaler Segmente	75
4.1	Graphenbasierte morphologische Merkmalsextraktion	77
4.1.1	Berechnung eines Skeletts aus einem Segment	79
4.1.2	Berechnung eines Graphen aus einem Skelett	79
4.1.3	Inhomogenität im Verhältnis zur Anzahl von Zyklen	80
4.1.4	Gegensatz von euklidischen Distanzen zu geodätischen Wegen	83
4.2	Isoperimetrisch morphologische Merkmalsextraktion	87
4.3	Auswertung	93
5	Implementierung	95
5.1	Übersicht	95
5.2	Datenbank	98
5.3	ImagingProjectGui	99
5.4	KnowledgeBaseGui	100
5.5	Verknüpfung zu Gait-CAD	104
6	Zusammenfassung	107
7	Literaturverzeichnis	111
A	Gegenüberstellung separierter und nicht-separierter Filterung	I
A.1	Beispiel einer nicht-separierten Filterung	I
A.2	Beispiel einer separierten Filterung	II
A.3	Vergleich von Filterungsbeispielen	III
B	Verschiedene Softwarepakete zur Bildanalyse	V
C	Wichtige Symbole und Bezeichnungen	VII
D	Abbildungsverzeichnis	IX
E	Tabellenverzeichnis	XI

1. Einleitung

1.1. Motivation

Das Thema der Arbeit beinhaltet die Erstellung eines Konzeptes zur Analyse großer biologischer Bilddatenmengen. Das Verständnis von Abläufen im Bereich biologischer Grenzflächen konnte in den letzten Jahren durch zahlreiche biomedizinische Studien erweitert werden. Der Fortschritt im biologischen Sektor ist u. a. der Weiterentwicklung im Bereich der bildgebenden Verfahren und der Biotechnologien zu verdanken. Hierfür stehen heutzutage eine Reihe von Akquisitionsverfahren bereit, die dreidimensionale, vierdimensionale, sogar schon fünfdimensionale digitalisierte biologische Bilddaten liefern.

Die offene Fragestellung der Arbeit lautet: Wie können biologische, stark fehlerhafte, große multidimensionale Datensätze effektiv und effizient analysiert werden? Das gilt, wenn, innerhalb von begrenzter Zeit, viele erarbeitete Daten qualitativ akzeptabel für Expertenentscheidungen genutzt werden könnten.

Lösungsverfahren sollen ermittelt und getestet werden, die es ermöglichen, den langwierigen Verlauf der Bildbearbeitung bzw. des Filterns zu beschleunigen und zu verbessern. Mit speziellen multidimensionalen Segmentierungsverfahren sollen die Probleme der gegebenen multidimensionalen Akquisitionen erkannt und korrigiert werden.

Momentane Probleme bei der Analyse von Bilddaten aus der Mikroskopie werden betrachtet, eruiert und modifiziert. Mit speziellen, erarbeiteten Be- und Verarbeitungsverfahren werden neue Möglichkeiten bereitgestellt, um die Schritte der Merkmalsextraktion und Klassifikation zu erweitern. Auf neuartige Weise werden morphologisch geometrische Eigenschaften verfolgt – wie z. B. Homogenität oder Heterogenität, dreidimensional- und vierdimensional-bedingt durch spezielle, gesteuerte toxikologische Einwirkungen. Für direkt vorliegende biologische Forschungsansätze im Bereich molekularer und zellulärer Wechselwirkungen funktioneller Grenzflächen werden entsprechende Verfahren der (semi-)automatischen Bildanalyse praxisorientiert angewandt und in bereits vorliegende Algorithmen integriert.

1.2. Darstellung des Entwicklungsstandes

Im Folgenden wird der momentane Stand der Entwicklung interaktiver multidimensionaler Bildanalysen von biomedizinischen Daten beschrieben. Nachdem die Bilddaten akquiriert wurden, folgt das Datenmanagement. Daraufhin folgen die thematisch wichtigen Schritte der Bildanalysen.

1.2.1. Akquisition

Bei all den heute üblichen mikroskopischen Verfahren muss nicht nur die Auflösungsstärke, sondern auch das Volumen, die Geschwindigkeit der Aufnahme sowie deren Kosten bedacht werden. In der Arbeit und den anlehenden biologischen Experimenten werden Auflösungen von zum Teil unter 100 Nanometer benötigt.

Mikroskopie

Mikroskopie ist eine wissenschaftliche Disziplin, die Gegenstände vergrößert darstellt, um sie betrachten zu können. Die Absicht des Zweiges der Wissenschaft ist es, die Gegenstände, die mit dem bloßen Auge nicht erkennbar sind, für Studien und Untersuchungen sichtbar zu machen. Die Mikroskopie erlaubt Forschern – z. B. aus der Physik oder aus der Biologie – mehr Informationen über ihren wissenschaftlichen Sektor zu sammeln.

Im biologischen Forschungsbereich ist die Mikroskopie heutzutage tief verankert. Sie wird einerseits in biomedizinisch wissenschaftlichen, andererseits in klinischen Zentren und Laboratorien angewandt [108].

In einer Vielzahl von biologischen Systemen können transgenetisch Lebewesen so beeinflusst werden, dass fluoreszierende Proteine spezielle Zellen, besondere Gewebe oder ganze Embryos etikettieren. Werden daraufhin die fluoreszenten Proteine von Licht einer bestimmten Wellenlänge getroffen, emittieren sie Licht einer anderen Wellenlänge. Die ungewollten Lichtfrequenzen werden daraufhin herausgefiltert, so dass nur noch das von der Markierung ausgestrahlte Licht verbleibt.

Solche transgenetischen fluoreszierenden Organismen bieten die Gelegenheit, spezielle Zellen zu visualisieren und das Gewebeverhalten während der Entwicklungsprozesse zu vergegenwärtigen.

Es gibt mehrere Möglichkeiten eine Zelle zu markieren und zu visualisieren [70]:

- Zellen können so verändert werden, dass sie sich selbst markieren. Hierbei handelt es sich um eine Transfektion. Dadurch wird die Zelle transgen¹.
- Der Zelle kann etwas beigefügt werden², was ihr die Möglichkeit gibt, sich selbst zu markieren. Die Zelle erhält somit ihre natürliche Eigenschaft. Sie ist jedoch nicht transgen. Die Gegebenheit ist aber zeitlich begrenzt, weil das Zugegebene abgebaut wird.

Eine Zelle kann in beiden Verfahren auch mit mehreren Markern versehen werden. Es ist allerdings begrenzt möglich, unterschiedlich farbige Marker mittels Mikroskop zu unterscheiden.

Die Absicht vieler biologischer Forschungsprojekte besteht darin, Entwicklungen oder Bewegungen einzelner Zellen und Gewebestrukturen *in vivo* nachzuvollziehen. Das war bis vor kurzer Zeit mit zur Verfügung stehenden Akquisitionsverfahren nicht wirklich möglich. Lediglich bei dünnen und durchsichtigen Proben war das durchführbar.

Selective Plane Illumination Microscopy

Die Größe und Undurchsichtigkeit der Fischembryos³ machen es in der Zellbiologie⁴ besonders schwierig, einzelne Zellen⁵ und deren Bewegungen bzw. Entwicklungen zu verfolgen [48].

Auf Grund:

- der Größe der Probe,
- der Streuungen des intakten und undurchsichtigen Gewebes,
- der Pigmente von sich entwickelnden Tieren,
- der Bewegung von lebenden Organen (Muskulatur, Herz, Blut, Augen, usw.) und
- des Bedürfnisses, die Probe unter physiologisch akzeptablen Bedingungen zu behalten,

wird eine hohe Auflösung in lebenden Strukturen benötigt.

Durch die Einschränkung ist es oft nötig, Proben zu fixieren, auch wenn viele der hochdynamischen Prozesse, die während Entwicklungen und Bewegungen vorkommen, (nur im vollen Detail) im intakten, lebenden Embryo studiert werden können [61].

¹transgen – genetisch verändert

²ähnlich dem Immunsystem

³häufig mehrere Millimeter dick

⁴Cytologie

⁵ca. 10 μ m und oft mehrere 100 μ m tief im Inneren eines Embryos

”Selective Plane Illumination Microscopy” (SPIM) ist ein Verfahren der Fluoreszenzmikroskopie, bei dem ein biologisches Objekt systematisch durch einen hauchdünnen Lichtvorhang bewegt wird. Das biologische Objekt wird Schicht für Schicht durchleuchtet, wobei jedes Mal eine Aufnahme entsteht [62]. Das SPIM bietet somit die Möglichkeit, Objekte räumlich detailgetreu (3D), in relativ geringen Zeitabständen aufzunehmen. Die Auflösung ist abhängig von den integrierten Kameras. Bei den neuesten integrierten Modellen handelt es sich z. B. um *Andor NEO* [106]. Dabei handelt es sich um eine 14-bit CCD-Kamera, die mit Qualität 14-bit einen Rahmen von 500 nm aufnehmen und in 2048×2048 Pixel mit entsprechender räumlicher Tiefe abbilden kann. Pro Sekunde ergeben sich somit 160 MB an gewonnenen Bilddaten [68].

Das *SPIM* basiert u. a. auf Ideen der konfokalen Fluoreszenzmikroskopie. Es arbeitet deutlich effizienter, akquiriert mehr und viel schneller Bilddatensätze und ist relativ kostengünstig.

Bei dem Verfahren wird nur beleuchtet, was tatsächlich beobachtet werden soll. Deshalb müssen zu untersuchende Proben nicht – wie oft in der traditionellen Mikroskopie nötig – zerschnitten und zerstört werden. Das Verfahren bietet relativ scharfe und multidimensionale Aufnahmen [153, 68, 2, 87, 106].

Mikro- / Nanotomographie

Die Computertomographie erstellt – nach dem Prinzip der Radontransformation – aus unterschiedlichen Richtungen eine große Anzahl von Röntgenbildern eines Objektes und rekonstruiert aus den Abbildungen nachträglich die nicht erfasste Volumenstruktur. Es handelt sich hierbei um eine sogenannte Rekonstruktion oder Rückprojektion [109]. Somit kann für jedes Volumenelement des Objektes der Absorptionsgrad ermittelt und daraufhin zum Beispiel die Dichte einer Probe bestimmt werden. Die wissenschaftlichen Veröffentlichungen und die Entwicklung von 3D-Röntgen-Mikroskopen und Mikro-Computertomographen begannen in den 1980er Jahren des 20. Jahrhunderts [129].

Mikro-CT-Scanner⁶ haben, wie der Name schon sagt, Ortsauflösungen im Mikrometerbereich und ermöglichen mithilfe der Tomographie zerstörungsfreie Untersuchungen von biologischen Strukturen [75]. So können aus einer Menge von 2D-Daten, mittels Schattenbildern aus vielen verschiedenen Blickwinkeln und der im Anschluss durchgeführten Radontransformation, 3D-Daten bzw. Volumendatensätze bestimmt werden [109, 134].

⁶kurz auch μ CT geschrieben

Seit Anfang 2000 können Volumendaten von kleinen biologischen Objekten in vivo aufgenommen werden [75]. Dadurch sind neuartige Studien an lebenden Tieren, biomedizinischen Proben, Nahrungsmitteln, Mikrofossilien und andere Studien möglich.

Nano-CTs können seit etwa 2005 eingesetzt werden. Wie aus dem Namen schon ersichtlich, ermöglicht die zerstörungsfreie 3D-Mikroskopie Untersuchungen in der Größe von wenigen Nanometern [134, 53]. Experimentell kommt das Verfahren in der vorliegenden Dissertationsschrift nicht zur Anwendung. Da die Tomographie jedoch eine stabile und robuste Alternative zu zuvor beschriebenen Verfahren darstellt, und hier eher eine Menge Schattenabbildungen, und nicht eine Menge von Reflexionen, vorliegt, wird der Tomographie ein Abschnitt in der Arbeit gewidmet.

Jedoch auf Grund der Tatsache, dass die Tomographie eine stabile und robuste Alternative zu zuvor beschriebenen Verfahren ist, und es sich hierbei eher um eine Menge von Schattenabbildungen und nicht um eine Menge von Reflexionen handelt, wird der Tomographie ein Abschnitt in der Arbeit gewidmet.

1.2.2. Datenmanagement

Das Datenmanagement umfasst die Entwicklung und Durchführung von Architekturen, Absicherungen, Praktiken und Prozeduren, die den ganzen Datenlebenszyklus eines Vorhabens genau festlegen. In der Dissertation wird der Bereich mit erläutert, weil auf dem Sektor der Lebenswissenschaften in Zukunft außerordentlich große Datenmengen eruiert werden müssen⁷ [118, 104].

Das Ausmaß der Datenmengen und der dadurch beanspruchte Aufwand wächst bei steigender Dimension exponentiell [64, 21]. Inzwischen haben sich Ansätze ergeben, derartige Datenmengen z. B. per Grid-Computing parallel zu bearbeiten [46, 124, 123, 147].

An der ETH in Zürich wird zur Bewahrung großer Bilddatensätze der Stadt Zürich die ZuBuD⁸ genutzt. Das Ziel ZuBuD-Bilddatenbank besteht darin, Bilddatensätze von Forschungsgruppen aus aller Welt zu speichern und zur Verfügung zu stellen [131].

Zur Bewahrung großer Datensätze wird am EMBL⁹ z. B. BLAST¹⁰ genutzt. BLAST dient dazu, experimentell ermittelte DNA- oder Protein-Sequenzen mit bereits in einer Datenbank vorhandenen Sequenzen zu vergleichen, um Ähnlichkeiten festzustellen. Der Aufwand liegt momentan im Bereich von einigen Petabytes [9, 76, 30].

⁷Große Datenmengen umfassen im ersten Jahrzehnt des 21. Jahrhunderts die Größenordnung von Tera-Bytes.

⁸ZuBuD – Zurich Building Image Database

⁹European Molecular Biology Laboratory

¹⁰Basic Local Alignment Search

Am KIT¹¹ wird mittlerweile ein neu entwickeltes System namens LSDF¹² eingesetzt. Platten- und Bandspeichervolumen wurden auf Petabytes erweitert und stehen für die biologische Forschung zur Verfügung. Die Large Scale Data Facility legt Daten in einer abgestuften Speicherstruktur nach einem Schichtenmodell ab. Genutzt werden dazu Hochgeschwindigkeitsspeicher am Erzeugungsort, Plattenspeicher in der zentralen Schicht sowie Bandspeicher in der Archivierungsschicht [122]. Sie ermöglichen eine schnelle Datenannahme, eine hochleistungsfähige Bilddatenanalyse und die Langzeitarchivierung der anfallenden Datenmengen [60].

In der Arbeit wird die LSDF genutzt und mit dem Prinzip des Grid-Computings für große Berechnungen angewandt.

Mithilfe von Grid-Computing können Berechnungsschritte der Bildanalyse auf mehreren, verschiedenen, geographisch voneinander unabhängigen Computern berechnet werden. Für das Grid-Computing kann z. B. *HDFS*¹³ genutzt werden. *Hadoop* ist ein Framework, das ermöglicht, Berechnungen von Daten mit Tausenden von Knoten und Petabytes zu berechnen [147].

Des Weiteren müssen die verschiedenen Datentypen bedacht werden. Transaktionen und die Nutzung verschiedener Datentypen können zu einem Informationsverlust und Fehlern führen. Eine Übersicht zu den verschiedenen Datentypen befindet sich im Anhang A.

Für ein akzeptables Management bestehen die Anforderungen an Algorithmen aus:

- der Nachvollziehbarkeit,
- dem geringen Rechenaufwand,
- der hohen Qualität,
- der guten Auswertung und
- der Parallelisierbarkeit.

¹¹Karlsruhe Institute of Technology

¹²Large Scale Data Facility

¹³The *Hadoop* Distributed File System

1.2.3. Multidimensionale Bildanalyse

Für das biologische Verständnis werden aus vorliegenden Daten dazugehörige Modelle erstellt. Die Erfahrungen wurden vor Jahren durch 2D-Daten gewonnen. In der Akquisition (Abs. 1.2.1) bietet der aktuelle Stand der Technik die Möglichkeit, 3D-Aufnahmen zu gewinnen – also Dinge räumlich zu betrachten – somit verschiedene Aufnahmen des gleichen Objektes miteinander zu fusionieren. Sogar Folgen von 3D-Aufnahmen über einen fest definierten Zeitraum, also 4D-Aufnahmen, sind heutzutage realisierbar und können für Bewegungen und Entwicklungen von biologischen Strukturen angewandt werden. Das bedeutet für die Biologie ein großer Fortschritt [7, 151, 108].

Die Aufgabe von Informatikern, Mathematikern und Ingenieuren besteht schließlich darin, die Möglichkeiten der Auswertung multidimensionaler Datensätze zu unterstützen und in ihrer Effizienz weiterzuentwickeln. In den letzten Jahrzehnten entstanden daraufhin eine Reihe von Unternehmen und Softwareprojekten, die sich auf die Bildanalyse spezialisierten. Das Ziel der sich daraus ergebenden Systeme und Konzepte liegt in der Unterstützung – mittels qualitativen Automatisierungen und effizienten Quantisierungen – bei der Entdeckung neuer, noch unbekannter biologischer Zusammenhänge.

Bildbearbeitung

Die Bildbearbeitung befasst sich mit Modifikationen des Bildes, die, ohne Wissen über Art und Inhalt, durchgeführt werden können. Ihre hauptsächliche Aufgabe liegt in der Aufbereitung der Aufnahme. Prinzipiell kann jedes Bild bearbeitet werden [84].

Spezielle Aufgaben der Bildbearbeitung sind z. B. die Reduzierung des Rauscheinflusses, die Eliminierung von Artefakten und Störeinflüssen, die Verbesserung des Kontrastes oder die Normierung der Bildgröße. Hierfür stehen spezielle Filter und Algorithmen zur Verfügung [84, 64].

Ein digitales Abbild der Dimension zwei kann als diskrete Funktion $D: \mathbb{Z}^2 \rightarrow \mathbb{R}$ innerhalb eines begrenzten zweidimensionalen Raumes interpretiert werden. $D(X)$ repräsentiert den Grauwert des Abbildes D im Bildpunkt X .

Ein digitales Bild kann unter Zuhilfenahme verschiedener Operatoren bearbeitet werden. Operatoren sind mathematische Anweisungen zur Bearbeitung von Funktionen und werden in drei verschiedene Gruppen aufgliedert:

Punktoperator Punktoperatoren modifizieren einen Bildpunkt X unabhängig von seinen Nachbarn, egal ob er zu einem bestimmten Segment gehört oder nicht. Sie werden zur globalen Änderung von Parametern, wie Helligkeit oder Kontrast, genutzt. Handelt es sich z. B. bei dem Funktionswert eines Bildpunktes um einen Grauwert $D(X)$, so wird der als Grauwerttransformation bezeichnet. Unterschieden werden hierbei lineare und nichtlineare Operatoren, die sich mit einer entsprechenden Funktion F wie folgt als Punktoperatoren umschreiben lassen [64]: $F : D(X) \rightarrow F(D(X))$

Bekannte Punktoperatoren sind *Schwellwert-Filter*, die *Negativtransformation* und die *Histogrammspreizung* [64].

Lokaler Operator Bei einem lokalen Operator handelt es sich um eine Funktion, die, in Wechselwirkung mit den benachbarten Bildpunkten, auf einen Bildpunkt wirkt¹⁴. Der neu berechnete Grauwert hängt also von der Umgebung des Bildpunktes ab, sowie von den Gewichtungsfaktoren, die von entsprechenden Koeffizienten festgelegt werden. Mit zunehmender Operatorgröße nimmt die Rechenzeit überproportional zu. Für eine zweidimensionale Bildfunktion $D = [D_{i_1, i_2}]$ hat die Faltung mit einem Maskenoperator $m = [m_{x_1, x_2}]$, der nur für den Bereich $x_1 \in [-r_1, \dots, r_1] \wedge x_2 \in [-r_2, \dots, r_2]$ des Nullpunktes von 0 verschiedene Werte besitzt, die Form [64]:

$$D_{i_1, i_2} \rightarrow [D \otimes m]_{i_1, i_2} = \sum_{x_1=-r_1}^{r_1} \sum_{x_2=-r_2}^{r_2} D_{i_1-x_1, i_2-x_2} \cdot m_{x_1, x_2} \quad (1.1)$$

Beliebiger Operator Der beliebige Operator kann als Extremfall des lokalen Operators interpretiert werden, wobei potentiell jeder Wert der Bildfunktion von allen Grauwerten des Bildes abhängig ist [109, 64].

Bildverarbeitung

In der Arbeit wird Bildverarbeitung¹⁵ als die inhaltliche Zusammenfassung benachbarter Bildpunkte, die einem bestimmten Homogenitätskriterium genügen, verstanden. Bei der Bildverarbeitung biologischer Daten handelt es sich somit um eine Vermessung und Interpretation von Segmenten biologischen Hintergrunds [84].

¹⁴z. B. Glättungsfilter - (Gauß-Filter, Medianfilter etc.) oder Kantenfilter - (Gradientenfilter, Laplace-Filter etc.)

¹⁵oft auch Segmentierung genannt

Die Interpretation ist selbstverständlich nicht mit den Beurteilungen und Erfahrungen eines Biologen gleichzusetzen. Die maßgeblichen Entscheidungen, sowie die Verantwortung für seine Handlungen im biologischen Sektor liegen beim Biologen selbst. Unerlässlich ist daher die enge Zusammenarbeit von Biologen und Informatikern, um den Erfolg so einer (semi-)automatisierten Analyse biologischer Bilddaten zu ermöglichen.

Einen Überblick über verschiedene Verfahren zur Segmentierung geben folgende Quellen: [84, 7, 51, 63, 64, 108]. Die Segmentierung bildet den ersten Schritt der Bildinterpretation, da sie eine Verbindung zwischen unstrukturierten Bildelementen und semantisch zusammengefassten Bereichen herstellt. Das *Regionenwachstum*¹⁶ ist eine Gruppe von Verfahren, mit denen relativ schnell Segmentierungen vollzogen werden können. Die Verfahren können auf beliebig dimensionale Bilddaten¹⁷ angewendet werden.

Bei komplexeren Bilddaten sind jedoch Speziallösungen notwendig, um eine zufriedenstellende Segmentierung zu erhalten. Die Segmentierungsansätze können grob in drei Gruppen unterteilt werden:

Interaktive Segmentierung: Interaktive¹⁸ Ansätze bieten dem Nutzer die Möglichkeit, im Wechselspiel mit der Software, spezielle Strukturen in Bilddaten manuell auszuwählen. Folglich muss viel Arbeit in die Entwicklung geeigneter grafischer Benutzeroberflächen¹⁹ investiert werden [16, 12, 135].

Modellbasierte Segmentierung: Modellbasierte Ansätze nutzen das Wissen über Form- und Oberflächenbeschaffenheit der zu segmentierenden Objekte. Das Wissen ist meist in mathematischer Form repräsentiert [29, 66, 55, 3].

Wissensbasierte Segmentierung: Wissensbasierte Ansätze nutzen Vorwissen über die zu erwartenden biologischen Daten. Mögliche Quellen des Wissens sind biologische Atlanten oder Trainingsdaten, die mithilfe entsprechender Interpreten zur spezifischen Segmentierung genutzt werden können [20, 148, 10].

An die Präzision der Bildauswertung sind höchste Anforderungen zu stellen, da sich ein Fehler in einem der Teilschritte auf alle folgenden Schritte und somit auch auf das Endergebnis auswirkt. Ein Fehler kann zudem später kaum automatisch korrigiert werden. Es ist nur die manuelle Korrektur möglich, was wieder zusätzlichen Nutzeroaufwand – in Form von Konzentration, Wissen und Zeit – bedeutet [130].

¹⁶<engl> *Region Growing*

¹⁷z. B.: bei 1D - Frequenzen; bei 2D/3D - absorptionsbasierte 2D-/3D-Aufnahmen; bei 4D - zeitliche Folge von 3D-Aufnahmen; bei 5D - verschiedene 4D-Fluoreszenzaufnahmen

¹⁸<lat> *inter agere* – <dt> wechselseitig

¹⁹<engl> *Graphical User Interface* – Abk.: GUI

Merkmalsextraktion

Merkmale²⁰ sind beobachtbare bzw. physikalisch messbare Eigenschaften von Objekten, die im Kontext der Mustererkennung geeignet sein sollen, eine möglichst fehlerfreie Klassifikation zu ermöglichen. Zur Schilderung von Merkmalen dienen Charakteristiken, die als Grundlage für die Zuordnung von Eigenschaften gegebener bildlicher Segmente dienen sollen. Die Merkmalsextraktion entspricht der Umwandlung von Eingangsdaten in eine Liste signifikanter Merkmale und dazugehöriger Gewichtungsfaktoren [93]. Die menschliche und maschinelle Bestimmung haben hierbei ihre jeweiligen Vorteile (Tab. 1.1). Auf Grund der Umwandlung in entsprechende Listen wird die Merkmalsextraktion auch oft als Dimensionsminimierung betrachtet.

	assoziativ	kombinatorisch
Maschine	0	++
Mensch	++	-

Tab. 1.1.: Mustererkennung bei Mensch und Maschine
[130]

Gewünschte Anforderungen sind [100, 74]:

- beobachtbar bzw. physikalisch messbar,
- mit geringem Aufwand zu gewinnen,
- interpretierbar und anschaulich,
- invariant bezüglich irrelevanter Variationen,
- hoch relevant bezüglich der Aufgabenstellung, die für die spätere Klassifikation geeignet ist.
- robust (unempfindlich gegenüber kleinen Störungen),
- in möglichst geringem Maße abhängig von unterschiedlichen Merkmalen und
- möglichst gering in der Anzahl von Merkmalen (Merkmalsvektor niedriger Dimension).

Sie empfehlen sich für eine anschließende (semi-)automatische Klassifikation.

Derartige Merkmale von Segmenten, angewandt auf entsprechende Bilddatensätze, sind z. B. Anzahl, Farben, Verhältnisse, Bewegungen sowie geometrische Eigenschaften (z. B. Volumen, Umfang, Länge, Tiefe, Breite, Krümmungen).

²⁰Merkmal – z. B. Deskriptoren, Kenngrößen, Eigenschaften

Weitere Merkmale sind z. B. Grauwert (Min, Max, Median, Durchschnitt), Rundheit, integrierte Dichte, Schwellenlimit, Scale-Invariant Feature Transform (SIFT) [88], multiScale Oriented Patches (MOPS) [19] und Speeded Up Robust Features (SURF) [8], Schwerpunkt, Flächeninhalt, kleinster umhüllender Kreis, größter innenliegender Kreis sowie Rundheit usw.

Generell lassen sich zwei verschiedene Arten von Merkmalen differenzieren. Einerseits können Merkmale durch reelle oder komplexe Zahlen beschrieben werden. Andererseits gibt es kategorische Merkmale, die durch Symbole gekennzeichnet werden. In beiden Fällen werden die Merkmale extrahiert, indem Transformationen auf den Mustern durchgeführt werden [84].

Merkmale unterstützen die Klassifikation von Objekten, Zuständen, Situationen und Zusammenhängen. Spezielle problembezogene Betrachtungen der Merkmale sollen der engen Verbindung der Mustererkennung mit biologischen Merkmalen dienen.

Klassifikation

Die durch Merkmale gewonnene Zuordnung von Segmenten in bestimmte Klassen wird Klassifikation genannt. Bei derartigen Zuordnungen handelt es sich um eine Einteilung eines bestimmten Merkmalsraumes in disjunkte Klassen [27].

Eine Klassifikation entspricht zum Beispiel der Zuordnung in verschiedene Farbklassen (z. B. rot, grün, blau). Verschiedene vorliegende farbliche Objekte können daraufhin direkt und eindeutig klassifiziert werden. Selbstverständlich können auch Klassifikationen manchmal nicht eindeutig sein (z. B. braun). Hierbei handelt es sich dann um unscharfe Mengen, die zu Fuzzy-Klassifikationen führen, die in der vorliegenden Arbeit allerdings nicht weiter betrachtet werden.

Eine Bildklassifikation von Segmenten kann anhand

- der Grauwertverteilung,
- der Position,
- der morphologischen Eigenschaften oder
- anhand von Basiswissen

erfolgen.

Abhängig von der Kommunikation zwischen Analytiker und Computer gibt es darüberhinaus die Unterteilung zwischen vollautomatischer, semiautomatischer und manueller Klassifikation [93, 64].

1.2.4. Offene Probleme

Der Forschungsbereich für Biologische Grenzflächen (z. B. Toxikologie, Genetik) ist ein umfassender naturwissenschaftlicher Sektor, in dem eine große Anzahl an Fragen noch unbeantwortet ist. Auf Grund des Fortschrittes in der Biologie (z. B. Fluoreszenz [133]) und der weitreichenden Möglichkeiten in der Akquisition (z. B. Mikroskopie [114], Spim) ist der Umgang mit den anfallenden immensen Datenmengen und deren tiefgehende Analyse noch nicht ausreichend hinterfragt. Die Auseinandersetzung mit dem Thema ist unabdingbar, um zu ermöglichen, in vielen Bereichen der Biologie, der Biochemie und der Biomedizin (z. B. toxikologische Einwirkungen) tiefergehende Fragestellungen effizient zu ergründen.

Außerdem ist ein weiteres Ziel der vorliegenden Arbeit, experimentell darzulegen, welche technischen und mathematischen Verfahren es ermöglichen, derartige Mengen akquirierter Daten zu Informationen zu verarbeiten und aus deren Analysen biologische Schlüsse zu ziehen [77]. Multidimensional können somit u. a. Entwicklungen von ganzen Zellstrukturen nachverfolgt und gedeutet werden. Die sich heutzutage ergebenden immensen Datenmengen erfordern allerdings einen unangemessenen Aufwand bei der (semi-)automatischen Analyse.

Desweiteren existiert noch nicht die Gruppe von Algorithmen bzw. Operatoren, die zeiteffizient genutzt und entsprechend qualitativ abgeändert werden kann. Bei vielen Abläufen fehlt die Balance zwischen Zeit, Qualität und Quantität, um einen möglichst hohen Informationsgehalt gewinnen zu können (Abb. 1.1).

Zur praktischen Umsetzung der multidimensionalen Bildbearbeitung fehlen bislang Konzepte zur theoretischen Analyse und zur praktischen Realisierung. Das im zweidimensionalen Sektor schon bekannte Modul der Separierung wurde noch nicht auf multidimensionale Datensätze erweitert. Deshalb steigt der Rechenaufwand bei multidimensionalen Problematiken und Datensätzen exponentiell.

Darüber hinaus werden Anwendungen und Auswertungen morphologischer Eigenschaften multidimensionaler biologischer Grenzflächen zur Gewinnung nützlicher Merkmale noch nicht beherrscht. Auch fehlen bislang Algorithmen zur Analyse der Inhomogenität von Netzstrukturen und Segmenten.

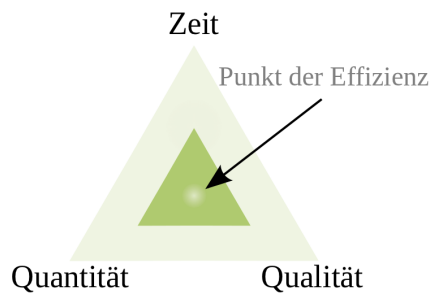


Abb. 1.1.: Balance zwischen Qualität, Quantität und Zeitaufwand

Problem toxikologischer Einwirkungen

In vielen biologischen Projekten wird hinterfragt, wie und womit *Noxen*²¹ – also toxisch wirkende Substanzen – bei relativ frühen Prozessen und Stadien, während der Entwicklung von Wirbeltieren [138], interagieren. Um in frühen Stadien unterschiedliche toxikologische Einwirkungen in Zebrafischarbmlingen²² studieren zu können, wurden verschiedene Substanzen gewählt.

Prozesse der frühen Entwicklung von Wirbeltieren werden hoch synchronisiert und hängen von komplizierten Signalkaskaden ab. Sie kontrollieren die Unterscheidungen und räumlichen Verläufe von Zellen und Zellgruppen.

Ziel der Biologie ist es u. a., *in vivo* die Wechselwirkung zwischen den *toxikologischen Wirkstoffen* und Prozessen früher Wirbeltierentwicklungen nachzuverfolgen.

Um derartige Projekte umzusetzen und wissenschaftlich zu unterlegen, wird eine große Datenmenge von sich entwickelnden Zellkulturen bzw. Zebrafischarbmlingen benötigt. Derartige Daten werden mithilfe von zeitlichen Folgen fluoreszenzmikroskopischer Aufnahmen gewonnen [104].

Ein Ausschnitt der sich entwickelnden Zebrafischarbmlinge muss zunächst experimentell bestimmt und kalibriert werden. Die Ursache hierfür liegt in der zeitlichen und qualitativen Begrenzung. Eine Zellteilung dauert in den gegebenen Experimenten ungefähr zwei Minuten [47]. Um die Verfolgung von Zellen zu gewährleisten, wurde die Aufnahme auf mindestens drei Zustände begrenzt. Eine Aufnahme darf somit jeweils maximal 40 Sekunden dauern.

²¹<engl> Noxa bzw. <dt> Noxen – Stoffe, die negativ auf einen Organismus einwirken

²²Zebrafischarbmling – *Danio rerio*

Die Fluoreszenzmikroskopie wird eingesetzt, um die Möglichkeit zu bieten, verschiedene fluoreszierende Stoffe mit verschiedenen Lasersequenzen anzuregen. Hier ist zu bedenken, dass die Vielzahl an möglichen Sequenzen seriell abläuft und zusammen maximal 40 Sekunden benötigen darf.

Deswegen wurden lediglich zwei Sequenzen gewählt. Eine Sequenz dient zum Anregen spezieller Zellkerne und eine weitere Sequenz wird für spezifische biologische Fragestellungen (z. B. zum Aufleuchten bestimmter Zellmembranen) genutzt. Jede einzelne Aufnahme darf deshalb je Sequenz maximal 20 Sekunden dauern.

Abwägungen müssen somit in Qualität und Quantität der Bilddaten getroffen werden. Die heutigen technischen Möglichkeiten sind hierzu noch begrenzt. Die Größe eines Voxels beträgt, laut des vorgegebenen Mikroskops (SP5), in dem Experiment $892 \times 892 \times 881 \text{ nm}^3$. Weiterhin liegt, um qualitativ akzeptable Beleuchtungen zu gewährleisten, die Grauwerttiefe bei 16-bit.

All die Ansprüche führten, für den erwähnten Ausschnitt eines Zebrafischblinds, zu der Eingrenzung von ca. $512 \times 512 \times 256$ Voxel.

Auf die sich daraufhin ergebenden Probleme des Datenmanagements immenser Datenmengen²³ und des großen Aufwands der Datenanalyse wurde keine Rücksicht genommen, sondern als Herausforderung und Möglichkeit betrachtet, es zum Teil in der Dissertationsschrift zu bewältigen [111].

Bei der multidimensionalen Bildanalyse wird durch die vielen nötigen Einschränkungen und durch die vielen möglichen Algorithmen eine Balance gesucht, die zu akzeptablen (semi-)automatischen Auswertungen der akquirierten Bilddatensätze führt. Eine derartige Auswertung soll den Biologen dabei unterstützen, die Wirbeltierentwicklungen und Zellbewegungen unter toxikologischen Einwirkungen multidimensional nachzuverfolgen und wissenschaftlich tragfähige Schlussfolgerungen zu ziehen.

²³ $512 \cdot 512 \cdot 256 \cdot 16\text{bit} = 128 \text{ MB}$ pro Zeitpunkt

Probleme des Griscelli-Syndroms

In dem Projekt *"Physiology and Pathology of Skeletal Muscle In Vivo"* werden Entwicklungen des Griscelli-Syndroms verfolgt [116]. Hierbei werden gesunde und kranke Nervenstrukturen in vivo verglichen. An Ratten, die vom Griscelli-Syndrom befallen sind, kann eine unterentwickelte Muskelbeschaffenheit nachgewiesen werden.

Die Synapsen gesunder Ratten weisen eine inhomogene Struktur auf. Die inhomogene Struktur entspricht einer "bretzeligen" Form, einer Form mit Ungleichmäßigkeit und vielen Löchern²⁴.

Synapsen von Ratten, die mit dem Griscelli-Syndrom befallen sind, weisen hingegen eher eine homogene Struktur auf. Das ist auf die Tatsache zurückzuführen, dass eine Vielzahl zu untersuchenden funktionierenden Rezeptoren nicht wieder zur entsprechenden Synapse zurücktransportiert werden kann. Folglich bleibt das Wachstum der Synapse zurück. Weitere Details sind unter [116] erläutert.

Von gesunden und kranken (mit dem Griscelli-Syndrom belasteten) Ratten werden mittels konfokaler Fluoreszenzmikroskopie Bilder von deren Diaphragmen²⁵ akquiriert. Hinterfragt wurde:

- die Gesamtzahl der Synapsen pro Muskel / Diaphragma,
- das Volumen pro Synapse,
- die Größe und die Morphologie der Synapse,
- die statistische Varianz innerhalb der Muskeln,
- der Unterschied zwischen gesund und krank.

Zur Aufnahme dienten Mikroskope des Typs SP2 bzw. SP5 [116]. Genutzt wurden blaue und rote Sequenzen. *Blau* diente für eine gute Qualität, während *Rot* für die direkte Visualisierung genutzt wurde.

Auf Grund des für die Mikroskopie relativ großen Diaphragmas wurde die Grid-Mikroskopie genutzt. Die Grid-Mikroskopie entsprach einer Folge, einem Netz, von 20x20 verschiedenen positionierten konfokalen Aufnahmen, welche den relativ großen Raum abdeckte.

²⁴Von Biologen wurde das mit *löchrigem Schweizer Käse* assoziiert

²⁵<lat> Diaphragma – <dt> Zwerchfell

Um den Aufwand zu senken, Zeit zu gewinnen und Wissen über die Form eines Diaphragmas²⁶ zu nutzen, wurden je Grid vom biologischen Experten drei Aufnahmen angefertigt (oberste, mittlere und unterste räumliche Ebene). Nun wurde die Entscheidung getroffen, ob an der Stelle später detailliertere Aufnahmen durchgeführt werden sollen.

Eine einzelne Grid-Mikroskopie enthielt insgesamt $2048 \times 2048 \times 60$ Voxel. Dem genutzten Mikroskop zufolge beträgt die Größe eines Voxels in dem Experiment $775 \times 775 \times 1000 \text{ nm}^3$. Als Grauwerttiefe wird bei dem Experiment vom Biologen 8-bit gewählt. Auf noch höher auflösende Bilder mit der Tiefe von 12- bzw. 16-bit wird hier nicht weiter eingegangen.

Ein typisches Problem bei Mikroskopaufnahmen ist die Segmentierung der Synapsen bzw. Zellen. Hierbei ist nicht nur das Auffinden eines solchen Segmentes relevant, sondern auch das Bewahren und das anschließende automatische Auswerten der Morphologie.

Der Zeitaufwand bei der Aufnahme eines ganzen Diaphragmas beläuft sich auf ungefähr drei Stunden. Trotz der relativ festen Struktur der Synapsen können Unregelmäßigkeiten auftreten.

Ein weiteres Problem liegt in der Tatsache, dass eine Vielzahl von Synapsen ziemlich nahe beieinander liegen, und es relativ schwer wird, die Synapsen voneinander klar getrennt zu extrahieren.

Weitere Ziele der Arbeit bestehen darin, Verfahren zu entwickeln, die:

- die Positionen von Synapsen finden,
- das Rauschen und die Artefakte behandeln,
- eine Mindestanzahl von Synapsen in jeglichen Umkreisen ermitteln,
- kranke von gesunden Zellen anhand morphologischer Eigenschaften unterscheiden,
- (semi-)automatische Segmentierungen vornehmen und dem Biologen manuelle Segmentierungen abnehmen.

²⁶Zwerchfelle bei Ratten sind U-förmig

1.3. Ziele und Aufgaben

Heutzutage ist es nicht mehr ausreichend lediglich 2D-Datensätze bzw. Bilder von biologischen Objekten aufzunehmen und zu analysieren [1]. Der momentane Stand der Technik ermöglicht es, ganze 3D-/4D- sogar 5D-/6D-Datensätze zu akquirieren. Darum ist es möglich, Daten in großem biologischen Rahmen aufzunehmen und sie zu speichern. Das Problem liegt aber darin, die Daten in einem akzeptablen Zeitrahmen und in akzeptabler Qualität (semi-)automatisch zu analysieren.

Die Promotionsarbeit hat das Ziel, neue Teilkonzepte und Algorithmen zur Gewinnung von Informationen aus zur Verfügung stehenden multidimensionalen Datensätzen zu erarbeiten und zu evaluieren. Auf Grund des extrem wachsenden Aufwands der Datenhaltung bis zum Endpunkt der Analyse in multidimensionalen Datensätzen ist eine genaue Betrachtung des gesamten Ablaufes und der Balance zwischen den verschiedenen Notwendigkeiten für eine nutzbare Datenanalyse nötig. Die multidimensionalen Datensätze werden u. a. mit neuartigen Akquisitionsverfahren²⁷ gewonnen.

Angewandt werden die Verfahren z. B. bei der Erforschung von neurologischen Krankheiten (wie dem Griscelli-Syndrom) und der dazugehörigen Auswirkungen (Neuromuscular-Junction) [120, 36, 149] oder von toxikologischen Einwirkungen verschiedener Substanzen auf Zellen [68, 50]. Optische Limits und Signalstörungen bei biologischen Grenzflächen²⁸ in vitro oder sogar in vivo sind Gründe für die Komplexität, genaue, qualitativ akzeptable Daten zu gewinnen und zu analysieren.

Die Entwicklung von Konzepten und Algorithmen zur multidimensionalen Bildanalyse soll die interaktive Nutzung im Forschungsbereich der biologischen Grenzflächen erleichtern. Die interaktive Nutzung soll die Effizienz beim Erlangen von Expertenwissen und Schlussfolgerungen steigern, um somit Fortschritte im Forschungsbereich biologischer Grenzflächen zu unterstützen.

In der Dissertationsschrift orientiert sich die Entwicklung der Teilkonzepte an folgendem Datenverarbeitungsablauf:

Datenakquisition Neuartige Akquisitionsverfahren bieten die Möglichkeit, qualitativ, quantitativ und zeitlich viel mehr Informationen und Daten bestimmter biologischer Abläufe zu gewinnen. Das führt allerdings auch zu neuen Problemen der folgenden Datenanalyse.

²⁷Fluoreszenzmikroskopie, SPIM, etc.

²⁸BioInterface

Datenmanagement Auf Grund der sich ergebenden riesigen Bilddatenmengen wird in der Arbeit das Bilddatenmanagement neu behandelt (s. Kap. 5).

Datenbearbeitung Ein neues multidimensionales Konzept bringt den Vorteil, eine Gruppe von Bildbearbeitungs- und Filterverfahren von exponentiellem Rechenaufwand auf linearen Rechenaufwand zu minimieren und dadurch zeitlich stark zu beschleunigen. Qualität und Quantität bleiben gleichermaßen bestehen (s. Kap. 2).

Datenverarbeitung Neuartige Verarbeitungs- und Segmentierungsabläufe helfen dabei, multidimensionale Daten zu segmentieren. Es wurde angestrebt, nicht nur 2D-Daten zu verarbeiten, sondern Volumen oder sogar Bewegungen in optimaler Weise zu analysieren (s. Kap. 3).

Merkmalsextraktion

Graphentheoretische mathematische Ansätze werden neuartig angewendet, um multidimensionale morphologische Eigenschaften zu extrahieren. Die isoperimetrische Ungleichung wird zur Bestimmung von multidimensionalen Eigenschaften in neuartiger Form angewendet (s. Kap. 4).

Cluster-Computing und Klassifikation Cluster-Computing und Klassifikation werden mittels der für Data-Mining-Aufgaben entwickelten MATLAB-Toolbox Gait-CAD bestimmt. Das Neue sind nicht die Verfahren, sondern die Anwendungen der gegebenen und per Gait-CAD genutzten Datenbanken (s. Kap. 5).

Interaktive (semi-)automatische Korrekturen Mit neu erstellten Programmen erhalten Experten die Möglichkeit, entsprechende Wissensdatenbanken multidimensional zu erweitern. Die Interaktivität erleichtert es, den multidimensionalen Ablauf von der Bildbearbeitung bis hin zur Klassifikation zu verbessern und zu beschleunigen (s. Kap. 5).

Das Ziel der vorliegenden Dissertationsschrift besteht darin, ein neues Konzept für die interaktive Auswertung multidimensionaler biologischer Bilddaten zu entwickeln. Einzelne Bildanalyseschritte sollen als neuwertige Module erarbeitet und in entsprechend neuwertige Konzepte integriert werden (Abb. 1.2). Hier werden praktische Probleme in der Analyse biologischer multidimensionaler Datensätze behandelt.

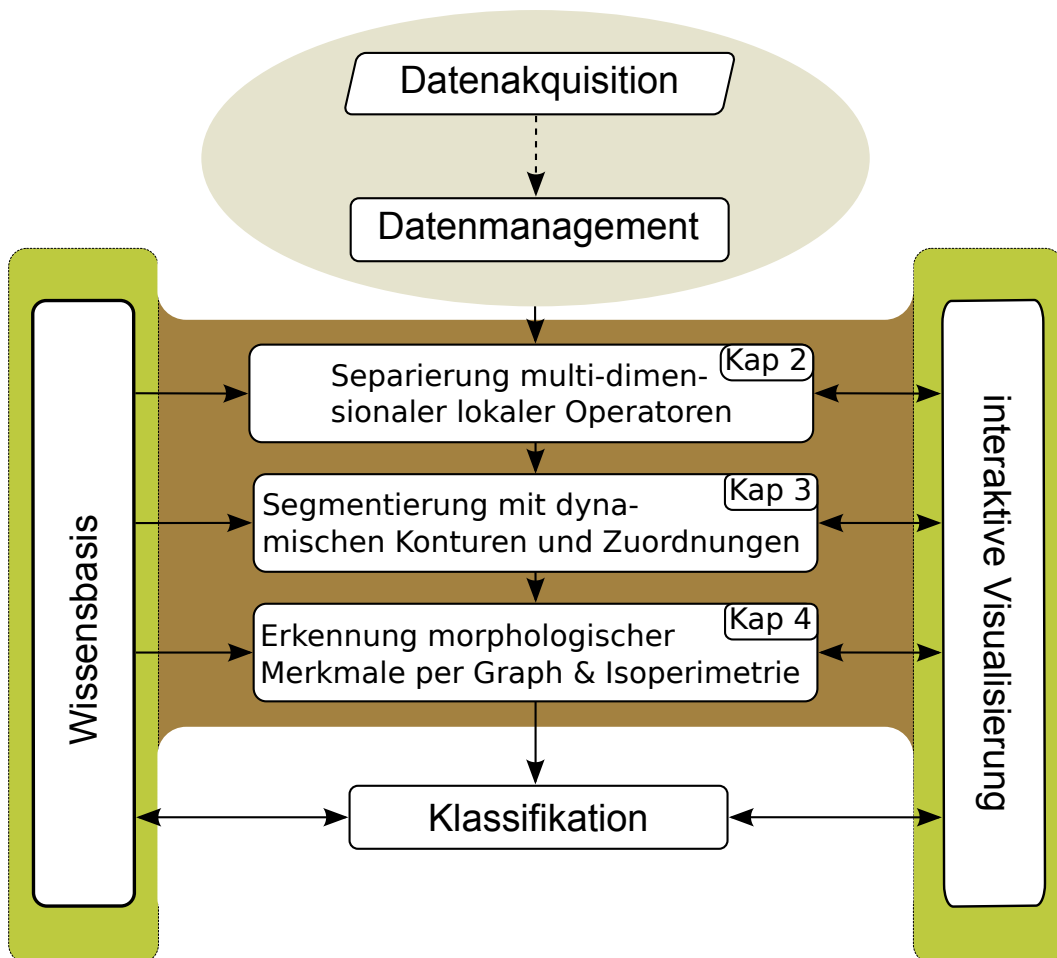


Abb. 1.2.: Konzept strukturierter multidimensionaler Bildanalyse großer biologischer Datensätze

Wie in der Einleitung schon beschrieben und in der Abb. 1.2 dargestellt, bestehen große Probleme in der Datenakquisition und im Datenmanagement. Hier sei auf die Zusammenarbeit mit anderen Instituten verwiesen. Bei der Zusammenarbeit wurde das Datenmanagement innerhalb des Projektes *LSDF* (Large Scale Data Facility) [138] und das Cluster-Computing mittels des *Hadoop*-Clusters [46] in Angriff genommen [104].

Um akzeptable Informationen aus den akquirierten Daten gewinnen zu können, müssen entsprechende Bilddaten zunächst bearbeitet werden (Kap. 2). Mit Filtern werden Fehler, Artefakte und Rauschen beseitigt – und das nicht nur zweidimensional, sondern multidimensional. Eine Vielzahl linearer lokaler Operatoren kann, auf Grund ihrer Faltungen, separiert angewandt werden [21]. Die Dissertation befasst sich des Weiteren mit multidimensionalen Anwendungen (Kap. 2). Derartige Separierbarkeit von linearen lokalen Operatoren führt, wie in der Dissertation beschrieben, zur fehlerfreien und deutlichen Effizienzsteigerung [110, 21] der multidimensionalen Bildbearbeitung.

In einem nächsten Schritt können Segmentierungen (Kap. 3) durchgeführt werden (Abb. 1.2). Zunächst werden in der vorliegenden Arbeit verschiedene Segmentierungsverfahren miteinander verglichen. Die Ergebnisse aus den Segmentierungen werden im Folgenden verwendet, um entsprechende Vorteile zu verknüpfen und die Nachteile möglichst zu vermeiden. Das entspricht einem neuartigen Verfahren zur multidimensionalen Verarbeitung.

Aus den segmentierten Bereichen können Muster und Merkmale gewonnen werden. Mit neuartigen Verfahren sollen daraufhin multidimensionale morphologische Eigenschaften bestimmt werden. Derartige Mustererkennungen ermöglichen die Erstellung von Wissensbasen und werden ebenso in der Arbeit beschrieben (Kap. 4).

Auf Verfahren der Klassifikationen, wie in Abb. 1.2 dargestellt, wird nicht weiter eingegangen. Es wird jedoch auf weitere Projekte verwiesen.

2. Neues effizientes multidimensionales Bearbeitungsverfahren

2.1. Problemdarstellung

Um Nutzen aus den enormen Datenmengen in der Biologie und steigenden Dimensionen ziehen zu können, werden realisierbare Filtertechniken benötigt, die mit den Entwicklungen von Akquisitionsverfahren und Biotechnologien Schritt halten. Die Berechnungszeit vorliegender Daten steht im Verhältnis zur Dimension der aufzunehmenden Objekte. Eine Aufnahme von Volumendaten über einen Zeitrahmen entspricht also einem vierdimensionalen Datensatz.

Es bietet sich an, die Berechnungszeit zu minimieren und den Rechenaufwand der Algorithmen für multidimensionale Eingangsdaten zu senken. Ein Fortschritt liegt demzufolge darin, keine Verluste in Qualität und Quantität hinzunehmen.

Um gegenwärtige Probleme und Beschränkungen vorliegender Algorithmen und deren Umsetzung zu überwinden, bieten sich Lösungsansätze aus dem Bereich der Signalverarbeitung sowie der Visualisierungstechnik an [71, 31, 35]. In der Dissertationsschrift wird die Aufmerksamkeit zunächst auf diskrete, lineare lokale Operatoren bzw. Filter gerichtet. Eine Filterung kann durch eine Faltung von Eingangsdaten (multidimensionales, diskretes Signal) mit einer Filtermaske¹ erreicht werden. Wenn die Tensor-Repräsentanz einer Filtermaske separierbar ist, wird, wie in der Dissertationsschrift theoretisch und praktisch dargelegt, der Rechenaufwand, entsprechend der linearen Filterung, wirksam minimiert. Das gilt für die Signaldimension sowie für die Kantenlänge der Filtermasken.

Auf Separierung basierende Methoden sind in zweidimensionalen Gebieten, wie der zweidimensionalen Bildverarbeitung, bekannt [38, 78, 141, 144, 126]. Die generelle Betrachtung von Tensoren d ter-Ordnung in der Dissertation ist durch das rapide Wachstum der Nachfragen nach schnellen Berechnungen multidimensionaler Datensätze – besonders bedingt durch Entwicklungen in biologischen und medizinischen Bildanalyse-sektoren – begründet.

¹Beides wird weitergehend als Tensor betrachtet.

Die Annahme, dass ein Effizienzgewinn in der Separierbarkeit multidimensionaler Filtermasken liegt, ist nicht nur auf die Tatsache beschränkt, dass viele prominente Filter (z.B. der Mittelwertfilter oder der Gauß-Filter) tatsächlich separierbar sind. Es kann außerdem in vielen Fällen ein nicht-separierbarer Tensor durch eine identische begrenzte Reihe von separierbaren Tensoren beschrieben werden (z. B. CANDECOMP² [25] und PARAFAC³ [56, 71, 57, 39]).

Zusammengefasst liegt der Hauptzweck des Vorgehens darin, zu verdeutlichen, wie die Separierbarkeit von Tensoren genutzt werden kann, um die Berechnungszeit für multidimensionale lineare Filter mit bestimmten Größenordnungen zu verringern. Zur Stärkung der theoretischen Beschreibungen werden experimentelle Validierungen mit künstlichen und biologischen Daten demonstriert [21, 110].

Im nun kommenden Abschnitt Abs. 2.2 werden zunächst Filterentwürfe beschrieben und definiert. Berechnungen und Beweise folgen in den Abschnitten danach.

2.2. Filterentwurf

In dem Abschnitt Abs. 2.2 werden diskrete Daten und lokale Operatoren behandelt. Beide werden als multidimensionale Tensoren betrachtet. Einen Überblick über Grundwissen der tensorischen Arithmetik, Separierbarkeit (Def. 5) und rechnerbasierten Fragestellungen kann in folgender Literatur gefunden werden: [91, 105, 31]. Bewältigt wurde das Problem der Separierung multidimensionaler linearer lokaler Operatoren in [110, 21, 69].

Die multidimensionalen Daten und entsprechende Filter sind d -dimensional, wobei d eine willkürliche aber feststehende natürliche Zahl ist⁴. Ein Element eines Eingangssignalraumes (Daten) wird weitergehend als Abbildung $\mathfrak{D} : \mathbb{Z}^d \rightarrow \mathbb{R}$ betrachtet. Um die weiteren theoretischen Vorgehen und Beweise möglichst abstrakt zu betrachten, entspricht die Mächtigkeit⁵ des Abbildes der Mächtigkeit von reellen Zahlen \mathbb{R} .

In der d -dimensionalen Datenanalyse bezeichnet das Wort *Separierbarkeit* die Eigenschaft, dass sich die Impulsantwort eines d -dimensionalen Filters durch d miteinander verknüpften 1-dimensionalen mathematischen Operationen darstellen lässt (Def. 5).

²CANonical DECOMPosition

³PARAllel FACTors Decomposition

⁴ $d \in \mathbb{N}$

⁵Anzahl der Elemente

Somit kann eine d -dimensionale Faltung auf d 1-dimensionale Faltungen reduziert werden. Oder, anstatt zwei d -dimensionale Filter hintereinander anzuwenden, kann zuerst ein Filter auf den anderen angewandt und dann mit dem Resultat das eigentliche Signal gefaltet werden. Das kann die Berechnungszeit immens senken, da der Rechenaufwand nicht mehr dem exponentiellen Umfang, sondern dem linearen Umfang entspricht [21, 110].

Ein 640×480 grau-skaliertes Bild wird daraufhin als $\mathfrak{D} : \mathbb{Z}^2 \rightarrow \mathbb{R}$ betrachtet, mit $S = \{(j_1, j_2) \in \mathbb{N}^2 \mid 1 \leq j_1 \leq p_1; 1 \leq j_2 \leq p_2\}$. Dessen Kantenlängen betragen deshalb $p_1 = 640$ und $p_2 = 480$.

Definition 1 (Muxel) *Anstelle der Begriffe Pixel⁶ oder Voxel⁷ sollen fortwährend multidimensionale Elemente eines Bildes / Datensatzes als **Muxel**⁸ bezeichnet werden. Ein Muxel eines Eingangssignals $\mathfrak{f} : \mathbb{Z}^d \rightarrow \mathbb{R}$ ist ein Paar $(i, f(i))$, wobei i ein Index in \mathbb{Z}^d ist [21, 110].*

Definition 2 (Filter) *Ein **Filter** \mathfrak{F} ist eine Abbildung zwischen zwei Signorräumen [21, 110, 64].*

Definition 3 (Filtermaske) *Eine **Filtermaske** eines Filters \mathfrak{F} (Def. 2) ist eine Funktion $g : \mathbb{Z} \rightarrow \mathbb{R}$, so dass $g(X) = F \star X, \forall X \in \mathbb{Z} : \mathbb{Z} \rightarrow \mathbb{R}$ [21, 110, 64].*

Zur Vereinfachung wird vorausgesetzt, dass jede Kantenlänge eines jeden Filtertensors ungerade ist und mit m_α bezeichnet wird. Der Radius ist somit wie folgt definiert: $r_\alpha := \frac{m_\alpha - 1}{2}$ ($\alpha = \{1, \dots, d\}$), wobei d eben der Filterdimension entspricht.

Definition 4 (Lokaler Operator) ***Lokale Operatoren** sind Filter (Def. 2) mit dazugehörigen Trägern⁹ [21, 110, 64].*

Theoretisch sind die Daten auf einem unbeschränkten Definitionsbereich definiert. In der Praxis jedoch sind gegebene (Bild)Daten oft diskret und stehen in Verbindung zu einem kompakten Träger. Deshalb werden weiterhin die Daten als Tensor dargestellt.

Im weiteren Vorgehen beschränkt sich die Aufmerksamkeit auf *lineare* lokale Operatoren (Def. 4), die mit einer Filtermaske (Def. 3) abgebildet werden können. Im zweidimensionalen Beispiel ist solch eine Filtermaske eine Matrix, während im endlich-dimensionalen Fall sie einem allgemeinen Tensor entspricht.

⁶PIcture (2D) ELement

⁷VOlume (3D) ELement

⁸MULTidimensionale (dD) ELement

⁹Ein "Träger" ist eine Nichtnullstellenmenge einer Funktion

Definition 5 (Separierbarkeit) *Ein Tensor \mathfrak{F} mit d -ter-Ordnung – der unter anderem als multidimensionaler lokaler Operator genutzt werden kann (Def. 4) – ist **separierbar**, wenn er sich mit gegebenen Vektoren v_1, \dots, v_d wie folgt darstellen lässt [21, 110, 64].:*

$$\mathfrak{F} = v_1 \otimes \dots \otimes v_d \quad (2.1)$$

Rang_d ist ein Begriff aus der linearen Algebra, der einer linearen Abbildung zugeordnet wird. Er entspricht hiermit der Dimension der linearen Hülle¹⁰ des entsprechenden Filters. Für separierbare Filter (Def. 5) gilt folglich: $\text{Rang}_d = 1$.

Bemerkung 1 *Separierbare Filter haben eine kanonische Dekomposition [25], die nur einen Term umfasst. Anderweitig, gemäß Kruskal [79], kann der schon enthaltene Rang des Tensors nicht größer als 1 sein.*

Die nun vorliegenden Definitionen und Bemerkungen werden benötigt, um weitere Details in den kommenden Abschnitten festzulegen und für die Beweise zu nutzen.

Im nun folgenden Abschnitt Abs. 2.2.1 wird für zweidimensionale und im Abschnitt Abs. 2.2.2 für multidimensionale lokale Filter der gegebene Rechenaufwand von separierten und nicht-separierten Filtern berechnet und gegenübergestellt [21, 110].

2.2.1. Separierbare zweidimensionale Filter

Hier wird untersucht, wie die Separierung von zunächst zweidimensionalen Filtern Berechnungszeit ersparen kann. Für das Verständnis wird zuerst der am meisten angewandte zweidimensionale Fall betrachtet, bevor anschließend die volle Allgemeinheit multidimensional und abstrakt in Betracht gezogen wird.

Ein zweidimensionaler Filter F ist repräsentiert durch eine $m_1 \times m_2$ -Matrix $F = (f_{i_1, i_2})$ ¹¹ und agiert auf zweidimensionalen Datensätzen, dargestellt durch ein $p_1 \times p_2$ -Array $X = (x_{j_1, j_2})$ ($1 \leq j_1 \leq p_1$; $1 \leq j_2 \leq p_2$).

Hier sei darauf verwiesen, dass $r_\alpha := \frac{m_\alpha - 1}{2}$ ¹². Verständnishafter wird weitergehend auf die Radien und nicht auf die Kantenlängen eingegangen.

¹⁰Die lineare Hülle entspricht der Menge aller Linearkombinationen der gegebenen Basisvektoren.

¹¹ $(-r_1 \leq i_1 \leq r_1; -r_2 \leq i_2 \leq r_2)$

¹² $\alpha \in \{1, 2\}$, ist ein Index für entsprechende Dimension.

Berechnungen von F auf X sind durch eine diskrete Konvolution / Faltung¹³ gegeben

$$y_{k_1, k_2} = F \star_{(k_1, k_2)} X = \sum_{i_1=-r_1}^{r_1} \sum_{i_2=-r_2}^{r_2} f_{i_1, i_2} \cdot x_{k_1-i_1, k_2-i_2}. \quad (2.2)$$

Der Wert für x_{j_1, j_2} wird in der Regel auf 0 gesetzt, wenn deren Indizes sich außerhalb der Grenzen befinden. Die Muxel y_{kl} entsprechen den Werten eines dyadischen Produkts [18] ($1 \leq k_1 \leq p_1$; $1 \leq k_2 \leq p_2$)

$$Y = F \star X = (y_{k_1, k_2}) \in \mathbb{R}^{p_1 \times p_2}. \quad (2.3)$$

Im zweidimensionalen Fall kann jeder lokale Operator (Def. 4) als Matrix beschrieben werden. Ist die Matrix des lokalen Operators bzw. der Filtermatrix F separierbar (Def. 5) integriert, so sind dessen Rang 1 (Bem. 1) und dessen Reihen und Spalten linear abhängig. Infolgedessen kann $F = ((f_{i_1, i_2}))$ als Produkt zweier Vektoren $v_1 = (v_{1, i_1})$ und $v_2 = (v_{2, i_2})$ geschrieben werden:

$$F = v_1 \cdot v_2^\top,$$

so dass

$$f_{i_1, i_2} = v_{1, i_1} \cdot v_{2, i_2}. \quad (2.4)$$

Definition 6 (Tensorprodukt) Die Faktorisierung in Konstruktion (Gl. 2.4) wird als zweidimensionales **Tensorprodukt** $F = v_1 \otimes v_2$ bezeichnet.

Die Separierbarkeit eines Filters kann mit Hilfe des Gauß-Algorithmus aus der Algebra getestet werden. Außerdem kann, wenn F nicht-separierbar ist, der Gauß-Algorithmus dazu genutzt werden, eine Dekomposition in eine Reihe von separierbaren Filtern umzuwandeln.

Wenn der Filter \mathfrak{F} separierbar integriert (Gl. 2.4, Gl. 2.3) und entsprechend dem zweiten Term von Gl. 2.5 angeordnet ist, wird eine Reduzierung der Anzahl von Multiplikationen und Additionen ersichtlich:

$$\begin{aligned} y_{k_1, k_2} &= \sum_{i_1=-r_1}^{r_1} \sum_{i_2=-r_2}^{r_2} v_{1, i_1} \cdot v_{2, i_2} \cdot x_{k_1-i_1, k_2-i_2} \\ &= \sum_{i_1=-r_1}^{r_1} v_{1, i_1} \sum_{i_2=-r_2}^{r_2} v_{2, i_2} \cdot x_{k_1-i_1, k_2-i_2} \end{aligned} \quad (2.5)$$

¹³Eine Faltung ist linear und kommutativ [21, 110, 64].

Hier sei angemerkt, dass der Term $\sum_{i_2=-r_2}^{r_2} v_{2,i_2} \cdot x_{k_1-i_1, k_2-i_2}$ nur einmal berechnet werden muss (benötigt $m_2 = 2 \cdot r_2 + 1$ Multiplikationen) und anschließend mit jedem anderen Muxel v_{1,i_1} multipliziert wird (weitere $m_1 = 2 \cdot r_1 + 1$ Multiplikationen).

Es werden lediglich $m_1 + m_2$ Multiplikationen für jeden der y_{k_1, k_2} Muxel bzw. Pixel benötigt. Daher werden zur Generierung der gesamten zweidimensionalen Ausgangsdaten, also der Ergebnismatrix Y in Gl. 2.3 $(m_1 + m_2) \cdot p_1 p_2$ Multiplikationen erforderlich.

Das ist bei Kantenlängen größer als Zwei eine signifikante Verbesserung gegenüber den nicht-separierten Methoden – nicht-separierten Methoden, die $m_1 \cdot m_2$ Multiplikationen pro Pixel bzw. $(m_1 \cdot m_2) \cdot p_1 p_2$ Multiplikationen für die gesamten zweidimensionalen Ausgangsdaten Y benötigen. Die Änderungen bei der Anzahl von Additionen verhalten sich ebenso.

Während nun in dem Abschnitt Abs. 2.2.1 die Separierbarkeit zweidimensionaler Filter betrachtet wurde, wird im folgenden Abschnitt Abs. 2.2.2 mit gleicher Herangehensweise abstrakter die Separierbarkeit multidimensionaler Filter beschrieben.

2.2.2. Separierbare multidimensionale Filter

Es wird vorausgesetzt, dass Eingangsdaten d -dimensionalen Arrays entsprechen (z. B. ein $\mathbb{R}^{p_1 \times \dots \times p_d}$ -Tensor $\mathfrak{X} = (x_{j_1, \dots, j_d})$). Filter sind folglich verkörpert durch einen $\mathbb{R}^{m_1 \times \dots \times m_d}$ -Tensor $\mathfrak{F} = (f_{i_1, \dots, i_d})$ mit Ordnung d , so dass gilt: $\forall \alpha \in \{1, \dots, d\} : -r_\alpha \leq i_\alpha \leq r_\alpha = \frac{m_\alpha - 1}{2}$ und $1 \leq j_\alpha \leq p_\alpha$.

Wie zuvor beschrieben, wird von Eingangsdaten x_{j_1, \dots, j_d} und Tensorfiltern f_{i_1, \dots, i_d} soweit festgelegt, dem Betrag 0 zu entsprechen, wo die Indizes außerhalb gegebener Grenzen sind. Die Indizes müssen infolgedessen nicht weiter berücksichtigt werden. Sinngemäß nach Gleichung Gl. 2.3 sind Muxel entsprechender Ausgangsdaten $\mathfrak{Y} \in \mathbb{R}^{p_1 \times \dots \times p_d}$

$$\mathfrak{Y} = \mathfrak{F} \star \mathfrak{X} = (y_{k_1, \dots, k_d}) \quad (2.6)$$

($1 \leq k_\alpha \leq p_\alpha$) gegeben durch eine d -dimensionale Faltung

$$y_{k_1, \dots, k_d} = \sum_{i_1=-r_1}^{r_1} \dots \sum_{i_d=-r_d}^{r_d} f_{i_1, \dots, i_d} \cdot x_{k_1-i_1, \dots, k_d-i_d}. \quad (2.7)$$

Das führt zu

$$\prod_{\alpha=1}^d m_{\alpha} \quad \text{Multiplikationen und} \quad (2.8)$$

$$\prod_{\alpha=1}^d m_{\alpha} - 1 \quad \text{Additionen} \quad (2.9)$$

für jeden Muxel y_{k_1, \dots, k_d} .

Bei gegebenen d -dimensionalen Ausgangsdaten, in dem Fall \mathfrak{Y} , die $\prod_{\alpha=1}^d p_{\alpha}$ Muxel umfassen, wird ein Berechnungsaufwand von:

$$\prod_{\alpha=1}^d p_{\alpha} \prod_{\alpha=1}^d m_{\alpha} \quad \text{Multiplikationen und} \quad (2.10)$$

$$\prod_{\alpha=1}^d p_{\alpha} \left(\prod_{\alpha=1}^d m_{\alpha} - 1 \right) \quad \text{Additionen} \quad (2.11)$$

benötigt, wenn die Berechnungen mit d -dimensionalen Tensorfiltern nicht separiert, sondern mit einer Tensorfaltung durchgeführt werden.

Aus einer einfachen Berechnung separierter multidimensionaler Operatoren resultiert, dass bei Separierungen und der darauf folgenden Berechnung, die Anzahl von Multiplikation und Addition je Muxel lediglich linear wächst – abhängig von der Größe des Filters¹⁴.

Genauer gesagt, wenn ein Tensor \mathfrak{F} separierbar ist, dann kann der Berechnungsaufwand für jeden einzelnen Ausgabemuxel von $\prod_{\alpha=1}^d m_{\alpha}$ zu $\sum_{\alpha=1}^d m_{\alpha}$ Multiplikationen und von $\prod_{\alpha=1}^d m_{\alpha}$ zu $\sum_{\alpha=1}^d (m_{\alpha} - 1)$ Additionen reduziert werden, wie in [21, 110] beschrieben.

Hier sei angemerkt: Wenn ein Tensor *separierbar* ist, kann er beschrieben werden als Tensorprodukt [21, 110]

$$\mathfrak{F} = v_1 \otimes \dots \otimes v_d. \quad (2.12)$$

¹⁴Tensorordnung und Kantenlängen

Mit Nutzung der Faktorisierung kann die Berechnung der Ausgangsdaten (Gl. 2.7) anhand von Auswertungen durchgeführt werden [110, 21] (s. Kap. A):

$$\begin{aligned}
 y_{k_1, \dots, k_d} &= \sum_{i_1=-r_1}^{r_1} \cdots \sum_{i_d=-r_d}^{r_d} v_{1,i_1} \cdots v_{d,i_d} \cdot x_{k_1-i_1, \dots, k_d-i_d} \\
 &= \sum_{i_1=-r_1}^{r_1} v_{1,i_1} \cdot \left(\cdots \sum_{i_{d-1}=-r_{d-1}}^{r_{d-1}} v_{d-1,i_{d-1}} \right. \\
 &\quad \left. \cdot \left(\sum_{i_d=-r_d}^{r_d} v_{d,i_d} \cdot x_{k_1-i_1, \dots, k_d-i_d} \right) \cdots \right).
 \end{aligned} \tag{2.13}$$

Der Berechnungsaufwand von y_{k_1, \dots, k_d} Muxel separierbarer Tensoren beträgt deshalb nur noch [110, 21] (s. Kap. A):

$$\sum_{\alpha=1}^d m_\alpha \quad \text{Multiplikationen und} \tag{2.14}$$

$$\sum_{\alpha=1}^d (m_\alpha - 1) \quad \text{Additionen.} \tag{2.15}$$

Wenn zur Bestimmung der Werte von $\prod_{\alpha=1}^d p_\alpha$ Muxel des gesamten Ausgangsdatensatzes separierbare Faltungen \mathfrak{M} (Gl. 2.6) angewandt werden, beträgt der Berechnungsaufwand [110, 21] (s. Kap. A):

$$\prod_{\alpha=1}^d p_\alpha \sum_{\alpha=1}^d m_\alpha \quad \text{Multiplikationen und} \tag{2.16}$$

$$\prod_{\alpha=1}^d p_\alpha \sum_{\alpha=1}^d (m_\alpha - 1) \quad \text{Additionen.} \tag{2.17}$$

Weitergehend bezeichnet T_{multi} die Berechnungszeit einer Multiplikation und T_{add} die einer Addition¹⁵. Die direkte Berechnung mit nicht-separierten Filtermasken $T_{\text{nicht-sep}}$ und die Berechnung mit der Nutzung separierter Filtermasken T_{sep} kann somit wie folgt beschrieben werden [110, 21]:

$$T_{\text{nicht-sep}} = \left(\prod_{\alpha=1}^d m_\alpha - 1 \right) \cdot T_{\text{add}} + \prod_{\alpha=1}^d m_\alpha \cdot T_{\text{multi}} \in \mathcal{O} \left(\prod_{\alpha=1}^d m_\alpha \right) \tag{2.18}$$

$$T_{\text{sep}} = \sum_{\alpha=1}^d (m_\alpha - 1) \cdot T_{\text{add}} + \sum_{\alpha=1}^d m_\alpha \cdot T_{\text{multi}} \in \mathcal{O} \left(\sum_{\alpha=1}^d m_\alpha \right) \tag{2.19}$$

¹⁵Zeiten von Zwischenspeicherungen o.ä. werden zunächst nicht beachtet.

Bei der Gegenüberstellung beider Gleichungen Gl. 2.19 und Gl. 2.18 können folgende Resümees gezogen werden: Ursprünglich lag ein polynomiales Wachstum bei den Kantenlängen und ein exponentielles Wachstum bzgl. der Tensorordnung d vor (Gl. 2.18). Nach der Separierung konnte gezeigt werden, wie der gesamte Aufwand – der für Multiplikationen und Additionen gebraucht wird – zu einem linearen Wachstum für Kantenlängen und Dimensionen (Gl. 2.19) reduziert werden kann.

Die Tatsache wird auch in Anbetracht einer deutlichen Beschleunigung und der Gegenüberstellung von $T_{\text{nicht-sep}}$ und T_{sep} verständlich [110, 21] (s. Kap. A).

$$\frac{T_{\text{nicht-sep}}}{T_{\text{sep}}} = \frac{(\prod_{\alpha=1}^d m_{\alpha} - 1) T_{\text{add}} + \prod_{\alpha=1}^d m_{\alpha} T_{\text{multi}}}{\sum_{\alpha=1}^d (m_{\alpha} - 1) T_{\text{add}} + \sum_{\alpha=1}^d m_{\alpha} T_{\text{multi}}} \quad (2.20)$$

Daher

$$\lim_{d \rightarrow \infty} \frac{T_{\text{nicht-sep}}}{T_{\text{sep}}} = \infty$$

und

$$\lim_{m_{\pi(1)}, \dots, m_{\pi(k)} \rightarrow \infty} \frac{T_{\text{nicht-sep}}}{T_{\text{sep}}} = \infty$$

Das gilt für alle Permutationen der Indizes $\pi \in \text{Perm}(\{1, \dots, d\})$ und $\forall k \in \{2, \dots, d\}$.

Zu bestimmen, ob ein Tensor – mit einer Ordnung größer zwei – separierbar ist, und wie er gegebenenfalls in separierbare Tensoren unterteilt werden kann (z. B. in ein Tensorprodukt), ist ein schwieriges noch zu lösendes Problem [39]. Es ist zu bedenken, dass nicht-separierbare Operatoren als Reihen separierbarer Operatoren approximiert werden können. Hierfür können bekannte und verschiedene Herangehensweisen aus der Literatur genutzt werden. Dazu gehören:

- die *higher order singular value decomposition* (HOSVD) [34],
- der *least squares approaches* [24, 154] oder
- die *Dekompositionen von Tensoren in einer minimalen Anzahl von Termen mit Rang 1*¹⁶ [32].

Nach der theoretischen Beschreibung multidimensionaler Filter und entsprechender Separierbarkeit Abs. 2.2, wird im folgenden Abschnitt der praktische bzw. experimentelle Bereich betrachtet (s. Abs. 2.3). In dem Bereich werden Experimente zur Separierung ausgewertet und deren Ergebnisse hinterfragt.

¹⁶CANDECOMP [25]: *canonical decomposition – kanonische Dekomposition* und PARAFAC [56]: *parallel factors decomposition*

2.3. Simulationsergebnisse

Um den Nutzen von separierten Tensor-Filtermasken, der in den theoretischen Auswertungen in Abs. 2.2 dargelegt wird, praktisch zu bestätigen, werden nun Experimente an (künstlichen) Eingabedaten durchgeführt. Die Daten werden per Mittelwert-Filter mit unterschiedlich großen Tensorordnungen und Kantenlängen (d. h. Radien) berechnet.

Um zu veranschaulichen, dass der zuvor beschriebene Berechnungsaufwand real ist, wurden die Berechnungszeiten

- des naiven bzw. nicht-separierten lokalen Filters,
- des separierten lokalen Filters sowie
- des Standard-FFT-basierten lokalen Filters¹⁷,

für die Ausführung der erforderlichen Konvolutionen, dokumentiert. Wie in Abb. 2.2, Abb. 2.3, Abb. 2.4, Tab. 2.1, Tab. 2.2, Abb. 2.5 und Abb. 2.6 ersichtlich, verdeutlichen die Ergebnisse die Vorteile der Separierung.

2.3.1. Separiert-basierte Methode vs. Naive Methode

In dem Abschnitt werden durchgeführte Experimente beschrieben, die separiert-basierte Methoden den naiven, nicht-separierten Methoden, zur Berechnung von Faltungen / Filterungen, gegenüberstellen.

Wie schon mit den theoretischen Analysen in vorherigen Abschnitten gezeigt, liegt bei separiert-basierten Methoden ein nur linearer Berechnungsaufwand vor. Die Methoden sind somit den naiven, nicht-separierten Methoden um Größenordnungen / Zehnerpotenzen überlegen.

2.3.2. Experimentelles Vorgehen

Die Berechnungen wurden ausgeführt an einem zwei-kernigen Intelprozessor[®] *Xeon* CPU, 3GHz mit 512KiB Cache, 2GiB RAM, 2GiB Swap, basierend auf *Linux 2.6.31 SMP*. Der Code wurde geschrieben in *C++*. Genutzt wurde die Bibliothek *Insight Toolkit (ITK)* [63], wobei die meisten vorliegenden Standardoperatoren nicht-separiert vorlagen Kap. B.

¹⁷Gemeint ist hierbei eine einfache Fast-Fourier-Transformation.

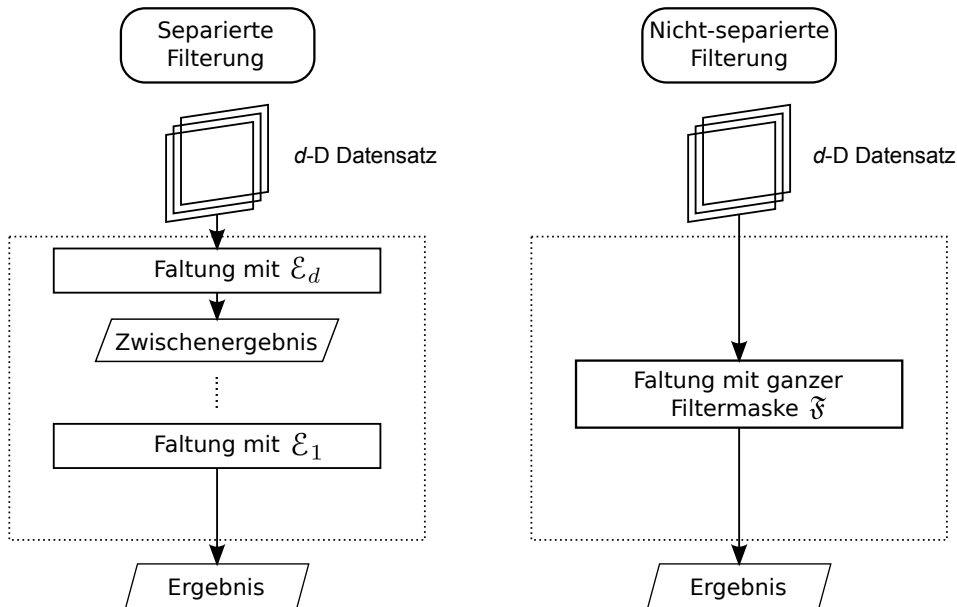


Abb. 2.1.: Schematische Darstellung von separiert-basierten Algorithmen (links) und nicht-separierten (naiven) Herangehensweisen (rechts) [21]

Die Implementierung wich aus praktischen Gründen ein wenig von der abstrakten Definition der Algorithmen ab. Vor allem war es nötig, die Ergebnisse eines jeden Konvolutionsschrittes zwischenspeichern. Auf Grund der Zwischenspeicherung ergab sich ein zusätzlicher Rechenaufwand bei separationsbasierten Algorithmen.

Der Aufwand für die Zwischenspeicherung ist jedoch bei Bilddaten, mit Dimensionen größer als zwei, im Verhältnis zu dem Rechenaufwand für die eigentliche Bildbearbeitung, zu vernachlässigen. Auch die Größe des Speicherplatzes ist vernachlässigbar, da die iterativen Zwischenspeicherungen nach den jeweiligen Berechnungen auch gleich wieder gelöscht werden können (Abb. 2.1).

2.3.3. Datensätze und Aufnahmen

Für jede Tensorordnung ($1 - 7$) wurden zufällige Tensoren generiert. Für jede feste Tensorordnung d und für jeden Radius $r \in 1, \dots, 5$ wurde ein Korpus von 30 Eingabedaten $\mathfrak{X}_1, \dots, \mathfrak{X}_{30}$ mit einheitlichen Radien (z. B. $R_{\mathfrak{X}_1}^1 = R_{\mathfrak{X}_1}^2 = \dots = R_{\mathfrak{X}_1}^d = \dots = R_{\mathfrak{X}_{30}}^1 = R_{\mathfrak{X}_{30}}^2 = \dots = R_{\mathfrak{X}_{30}}^d$) erstellt. Relativ lange Laufzeiten in der Bearbeitung, von theoretisch mindestens 1 Monat je Berechnung, haben dazu bewogen, auf die Aufnahme von Bearbeitungszeiten für Tensoren der Ordnung 7, mit Radien von 3 und mehr zu verzichten. Berechnungen für Radien 1 und 2 wurden aber durchgeführt. Ebenso wurde für Tensoren der Ordnung 6 und jeweils Kantenlängen 3 – 5 nur eine Messung aufgezeichnet.

2.3.4. Ergebnis

Um Vergleichsmessungen zu aggregieren, wurden Auswertungen¹⁸ in jedem multidimensionalen Testobjekt berechnet (s. Abs. 2.3.3). Bei dem Versuch wurden, an gegebenen multidimensionalen Datensätzen, multidimensionale lokale Operatoren angewandt.

Die Ergebnisse sind in Tab. 2.1 und Tab. 2.2 aufgelistet. Je Radius und dazugehöriger Tensorordnung wurden die entsprechenden Berechnungen 30 Mal durchgeführt. Ermittelt wurde daraufhin die durchschnittliche, maximale und minimale Berechnungsdauer. Die dazugehörige Standardabweichung wurde außerdem beigefügt.

Wie bereits in Abs. 2.3.3 beschrieben, konnte, bedingt durch den immensen Zeitaufwand, bei der Tensorordnung 7 nur eine einzelne Berechnung umgesetzt werden. Zum Verständnis des immensen Wachstums wurde das Ergebnis trotzdem in den Tabellen beigefügt. Werte für dessen Maximum, das Minimum und den Mittelwert sind folglich gleich und die Standardabweichung wird weggelassen.

Für die Tensorordnung von 1 bis 5 konnten allerdings mehrere Tests durchgeführt und entsprechende Standardabweichungen beigefügt werden. Die sich ergebenden Standardabweichungen sind bei niedrigen Radien und Tensorordnungen vernachlässigbar und können als repräsentative mittlere Bearbeitungszeiten für die Testobjekte aufgefasst werden. Das erklärt auch, warum entsprechende Standardabweichungen in den Tabellen Tab. 2.1 und Tab. 2.2 als 0-Wert abgerundet werden.

In den beiden Tabellen kommt zunächst zur Geltung, dass bei kleinen Tensorordnungen und kleinen Filterradien die Dauer der Berechnung des nicht-separierten lokalen Operators geringer ist, als die eines separierten lokalen Operators. Der Grund liegt darin, dass die Anzahl der Berechnungsschritte, egal ob separiert oder nicht-separiert, so gering ist, dass das Laden und Speichern der Zwischenergebnisse, was bei separierten Berechnungen öfter vorkommt, als bei nicht-separierten Berechnungen, länger dauert, als die jeweilige Berechnung. Weiterhin ist zu erkennen, dass die Berechnungen bei einigen nicht-separierten lokalen Operatoren (z. B. der Ordnung 1 mit den Filterradien von 1 bis 3 und der Ordnung 4 mit den Filterradien von 1 bis 2) vom Zeitaufwand her identisch scheinen. Der längere Zeitaufwand entsprechender separierter lokaler Operatoren liegt in dem benötigten, hardware-spezifischen Dateileseaufwand und Dateispeicheraufwand. Sollten die Tensorordnungen und die Filterradien des lokalen Operators so klein sein, dass deren eigentliche Berechnungsdauer geringer ist, empfiehlt sich die Separierung nicht. Bei höheren Tensorordnungen und Radien, deren Berechnungsdauer deutlich über dem Lese- und Speicheraufwand liegt, wird die separierte Berechnung lokaler Operatoren allerdings erheblich schneller (s. Abs. 2.3.2).

¹⁸z. B. Standardabweichung von Bearbeitungsmessungen zu einem festen Zustand

Radien	Tensor- ordnung	Nicht-Separiert (in Sek)			
		Mittelwert	Standard- abweichung	Max	Min
1	1	0.06	0.00	0.06	0.06
	2	0.06	0.00	0.06	0.06
	3	0.06	0.00	0.06	0.06
	4	0.09	0.00	0.09	0.09
	5	1.67	0.12	1.80	1.55
	6	65.66	0.37	66.03	65.29
	7	2625.91	3.26	2629.18	2622.65
2	1	0.06	0.00	0.06	0.06
	2	0.06	0.00	0.06	0.06
	3	0.06	0.00	0.06	0.06
	4	0.56	0.00	0.56	0.56
	5	40.09	0.30	40.39	39.79
	6	2825.48	10.54	2836.02	2814.93
	7	616736.17	-	616736.17	616736.17
3	1	0.06	0.00	0.06	0.06
	2	0.06	0.00	0.06	0.06
	3	0.08	0.01	0.09	0.08
	4	3.01	0.01	3.02	3.00
	5	150.87	7.57	158.44	143.34
	6	31985.61	-	31985.61	31985.61
4	1	0.06	0.00	0.06	0.06
	2	0.06	0.00	0.06	0.06
	3	0.13	0.01	0.15	0.12
	4	11.14	0.02	11.16	11.12
	5	729.84	27.04	756.88	702.79
	6	198624.06	-	198624.06	198624.06
5	1	0.06	0.00	0.06	0.06
	2	0.06	0.00	0.06	0.06
	3	0.22	0.00	0.23	0.22
	4	31.89	0.06	31.94	31.83
	5	2536.94	55.22	2592.16	2481.72
	6	420162.31	-	420162.31	420162.31

Tab. 2.1.: Statistiken von aufgezeichneten Berechnungszeiten $T_{nicht-sep}$ für unterschiedliche Filterradien und Tensorordnungen, bei je 30 Berechnungen eines Mittelwertfilters

Radien	Tensor- ordnung	Separiert (in Sek)			
		Mittelwert	Standard- abweichung	Max	Min
1	1	0.06	0.00	0.06	0.06
	2	0.11	0.00	0.12	0.11
	3	0.17	0.00	0.17	0.17
	4	0.23	0.00	0.24	0.23
	5	0.49	0.04	0.53	0.45
	6	2.45	0.14	2.59	2.31
	7	27.56	1.53	29.10	26.03
2	1	0.06	0.00	0.06	0.06
	2	0.11	0.00	0.12	0.11
	3	0.17	0.00	0.17	0.17
	4	0.24	0.00	0.24	0.23
	5	0.58	0.04	0.62	0.54
	6	4.26	0.16	4.38	4.06
	7	50.18	0.56	50.74	49.63
3	1	0.06	0.00	0.06	0.06
	2	0.11	0.00	0.12	0.11
	3	0.17	0.00	0.17	0.17
	4	0.24	0.00	0.24	0.24
	5	0.77	0.05	0.82	0.72
	6	6.73	0.11	6.84	6.62
4	1	0.06	0.00	0.06	0.06
	2	0.11	0.00	0.12	0.11
	3	0.17	0.00	0.18	0.17
	4	0.25	0.00	0.25	0.24
	5	1.03	0.07	1.10	0.96
	6	10.24	0.15	10.40	10.09
5	1	0.06	0.00	0.06	0.06
	2	0.11	0.00	0.12	0.11
	3	0.17	0.00	0.18	0.17
	4	0.25	0.00	0.25	0.25
	5	1.35	0.09	1.44	1.26
	6	14.54	0.20	14.74	14.34

Tab. 2.2.: Statistiken von aufgezeichneten Berechnungszeiten T_{sep} für unterschiedliche Filter-
radien und Tensorordnungen, bei je 30 Berechnungen eines Mittelwertfilters

Für eine graphische Beschreibung sei auf Abb. 2.2, Abb. 2.3 und Abb. 2.4 verwiesen. In Abb. 2.2 und Abb. 2.3 stehen die gelben Balken für die Ergebnisse von nicht-separierten und die roten gestreiften Balken für Ergebnisse von separierten Berechnungen. Abb. 2.2 verdeutlicht die Aufwandsentwicklung bei dem Radius 4, zu unterschiedlichen Tensorordnungen und Abb. 2.3 illustriert den Aufwand bei der Dimension 4, zu unterschiedlichen Filterradien. Die Bearbeitungszeiten wurden logarithmisch skaliert abgebildet, um die weit ausschwenkenden Verhältnisse noch graphisch zu verdeutlichen. In Abb. 2.4 werden die Berechnungszeiten, abhängig von Radius und Dimension, als dreidimensionales Diagramm dargestellt. Das rote bzw. von der Rückseite her grüne Netz stellt die Rechenaufwände nicht-separierter Berechnungen dar. Das blaue Netz steht hierbei für die Rechenaufwände von separierten Berechnungen. Entsprechende Fakten wurden aus Tab. 2.1 und Tab. 2.2 entnommen.

Da die graphische Darstellung der Messungen durch relativ große Zahlen bei höheren Dimensionen geprägt ist, wurde das Fortschreiten der Bearbeitungszeiten für eine feste Tensorordnung mit variablen Kantenlängen und umgekehrt dargestellt. Die Untersuchungen der Daten und Diagramme verdeutlichen, dass die experimentell gewonnenen Messungen im Einklang mit den theoretischen Vorhersagen stehen. Während die naiven, multidimensionalen Filtermethoden eine superlineare Progression der Bearbeitungszeit verursachen, reduzieren beschriebene Separierungen die Berechnungskomplexität auf ein lineares Wachstum, sowohl in Abhängigkeit der Radien, wie auch der Tensorordnung.

Ohne die beschriebenen Separierungen ist eine multidimensionale Filterung für große Dateneingabe-Tensoren und Filtermasken-Tensoren nicht angemessen durchführbar (Tab. 2.1).

Zur Filterung mit einer Filtermaske sechsten Grades und dem Radius vier in jeder Dimension benötigte eine naive, nicht-separierte Filterung mehr als zwei Tage. Im Gegensatz führte die Berechnung der gleichen Aufgabe mit einem separierbaren Filter in rund zehn Sekunden zum gleichen Ergebnis (s. Tab. 2.2).

Auf eine abstrakte Ebene gebracht führt das zu folgendem theoretischen Extremfall: Die Dauer der einzelnen Additionen wird für den weiteren Vergleich vernachlässigt. Allerdings, die Dauer einer einzelnen Multiplikation entspräche einer Sekunde. Die Berechnung eines 7-dimensionalen Datensatzes mit einem 7-dimensionalen nicht-separierten linearen lokalen Operator (mit gleichen Radien 8 bzw. Kantenlängen 17) entspräche somit 17^7 Sekunden bzw. ungefähr 13 Milliarden Jahre – der Dauer des vermuteten bisherigen Alters des Universums [72]. Wird hingegen die Berechnung mit separierten linearen lokalen Operatoren durchgeführt, beträgt der Zeitaufwand lediglich $17 \cdot 7$ Sekunden bzw. knapp zwei Minuten.

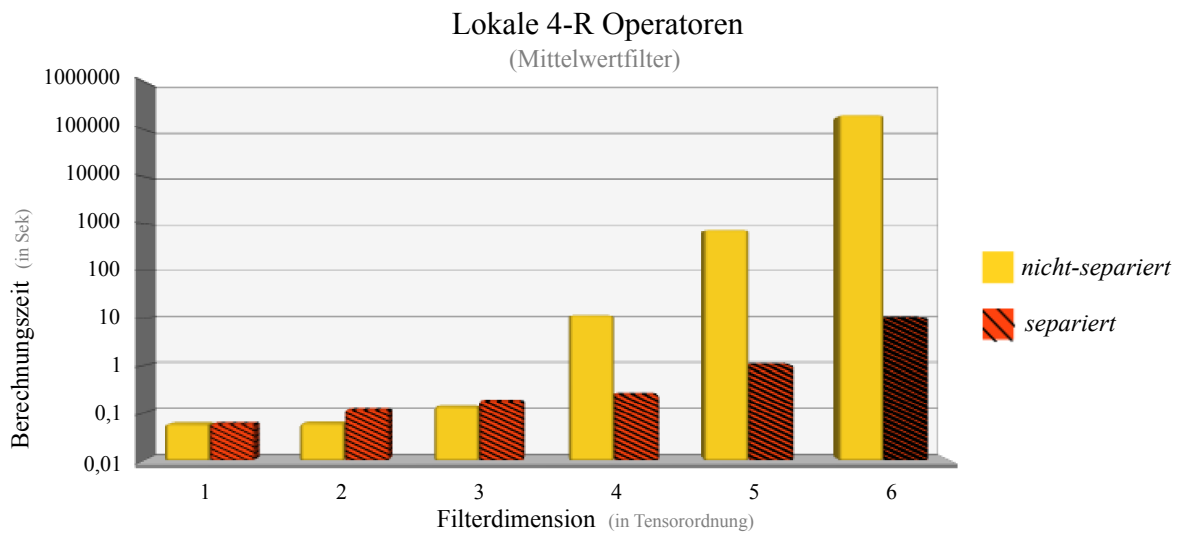


Abb. 2.2.: Diagramm für multidimensionale lokale Operatoren (Mittelwertfilter) mit Radius vier – separiert im Gegensatz zu nicht-separiert. Die Bearbeitungszeiten wurden logarithmisch skaliert abgebildet, um die weit ausschwenkenden Verhältnisse noch graphisch zu verdeutlichen.

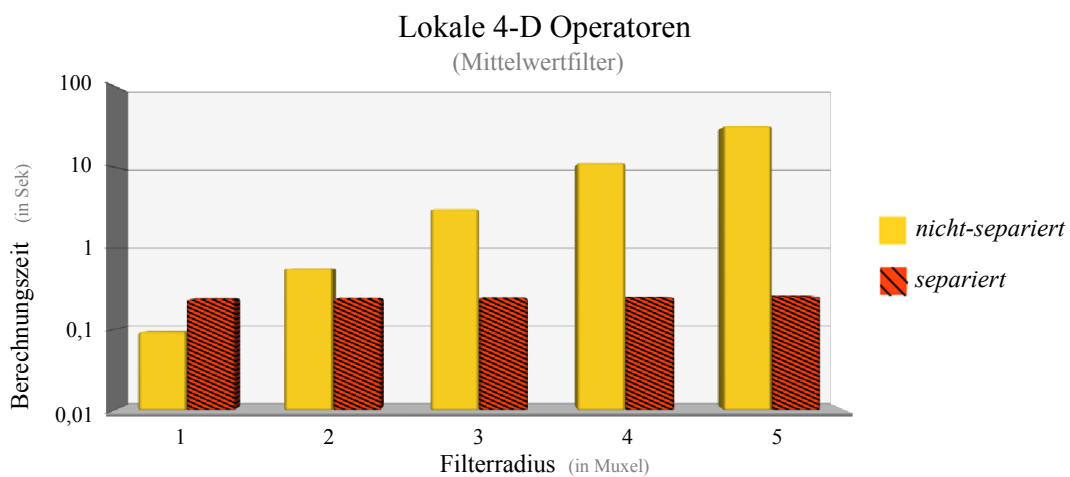


Abb. 2.3.: Diagramm für vierdimensionale lokale Operatoren (Mittelwertfilter) mit variierenden Radien – separiert im Gegensatz zu nicht-separiert. Die Bearbeitungszeiten wurden logarithmisch skaliert abgebildet, um die weit ausschwenkenden Verhältnisse noch graphisch zu verdeutlichen. Die Änderung der Berechnungszeit ist bei separierten Operatoren linear, hier nicht zu erkennen und im Verhältnis zu den nicht-separierten Operatoren vernachlässigbar.

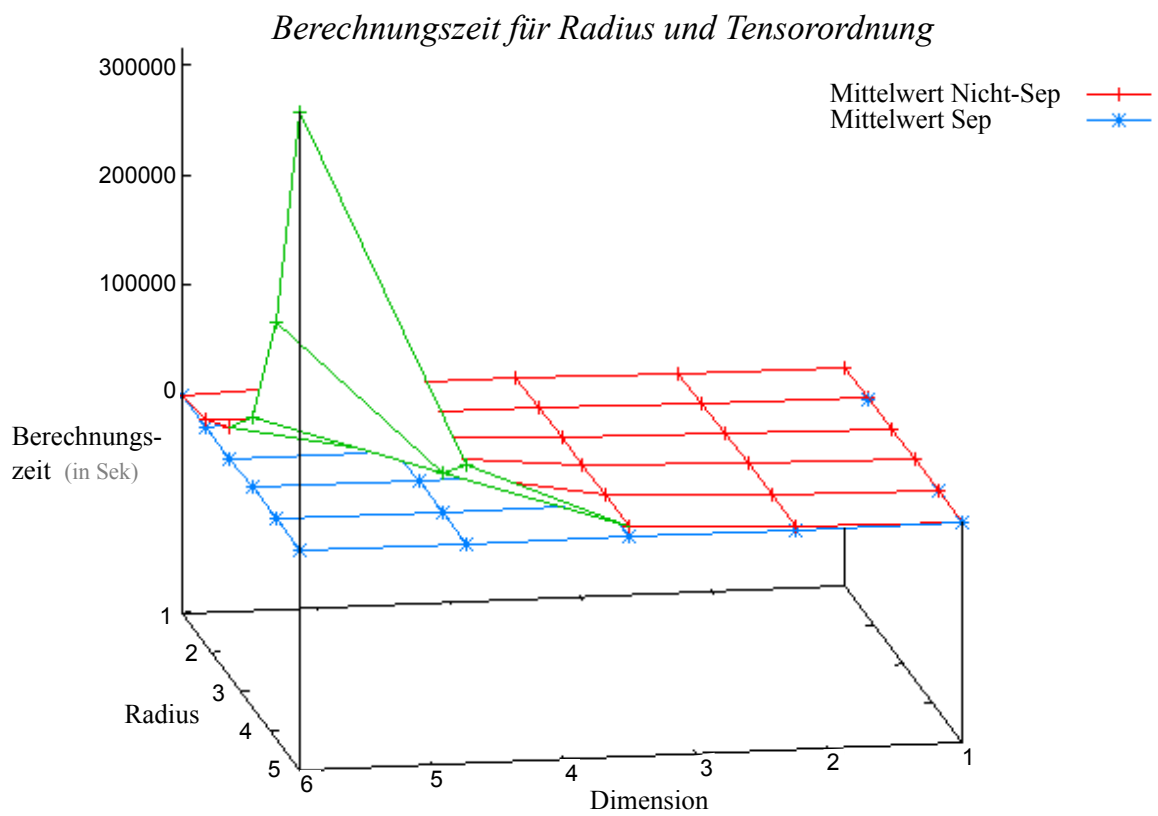


Abb. 2.4.: Wachstum der Verarbeitungszeit für die Faltung bzw. Filterung als eine Funktion von Tensorordnung und Filterradius

2.3.5. Vergleich mit FFT-basierten Konvolutionen

Zuvor wurden Experimente präsentiert, die die Komplexitätsgrenzen, die im Rahmen der theoretischen Überlegungen festgelegt wurden, illustrieren. Für die separierun-
sbasierte Methode waren die Ergebnisse im Einklang mit dem vorhergesagten linearen
Anstieg der Berechnungszeit – sowohl bezogen auf die Tensorordnung, als auch auf die
Kantenlängen der Filtermasken.

Während aus didaktischen Gründen sich ein Vergleich mit den naiven Verfahren emp-
fiehl, zählt in praktischer Hinsicht vielmehr ein Vergleich mit den aktuellen schnellen
Faltungsmethoden. Wie in [21] beschrieben, sind die Komplexitätsgrenzen der separier-
baren Methoden, im Vergleich zu FFT-basierten Verfahren, in Bezug auf die Tensor-
ordnung und die Kantenlängen deutlich effizienter.

FFT-basierte Verfahren liegen, trotz gegebener Separierbarkeit, in der superlinearen
Aufwandsklasse (s. Gl. 2.22) ^{19, 20}.

$$\sum_{i=1}^d \mathcal{O}(\log(m_i) \cdot m_i) \quad (2.21)$$

O. B. d. A. ²¹ $m = m_1 \leq \dots \leq m_d$. Somit gilt für den Rechenaufwand weitergehend:

$$\mathcal{O}(d \cdot \log(m) \cdot m) \quad (2.22)$$

Hingegen, separierte lokale Operatoren befinden sich in der linearen Aufwandsklasse
(s. Gl. 2.24) ²².

$$\sum_{i=1}^d \mathcal{O}(m_i) \quad (2.23)$$

O. B. d. A. $m = m_1 \leq \dots \leq m_d$. Somit gilt für den Rechenaufwand weitergehend:

$$\mathcal{O}(d \cdot m) \quad (2.24)$$

¹⁹ \mathcal{O} - Landau-Kalkül

²⁰super-lineare Aufwandsklasse: $\mathcal{O}(\log(x) \cdot x)$

²¹O. B. d. A. - ohne Beschränkung der Allgemeinheit

²²lineare Aufwandsklasse: $\mathcal{O}(x)$

Als praktisches Anliegen muss berücksichtigt werden, dass bei der multidimensionalen FFT erwartet werden kann, dass ein umfangreicherer Rechenaufwand impliziert wird [21]. Um die Argumente für die separierungs-basierte Methode zu bestätigen, wurden weitere Experimente durchgeführt, die in Abb. 2.5 und Abb. 2.6 dargestellt sind.

Bei dem Experiment wurden, wie zuvor, Mittelwert-Filter-Masken und die frei zur Verfügung stehende FFTW-Bibliothek [44, 45] genutzt. Sie wurden auf randomisierte Arrays, mit einheitlichen Kantenlängen von 20 Muxel in jeder Dimension, angewandt.

Die graphischen Darstellungen Abb. 2.5 und Abb. 2.6 sind im Einklang mit den theoretisch gewonnenen Erwartungen und zeigen, dass in der Praxis die separations-basierte Methode (grün) durchaus als Alternative zu den FFT-basierten Faltungsmethoden (rot) angewandt werden kann. Wie dargestellt, ist das nicht nur im Limes (wie von den Komplexitätsgrenzen vorhergesagt), sondern auch für Probleme einer geringen räumlichen Komplexität gegeben.

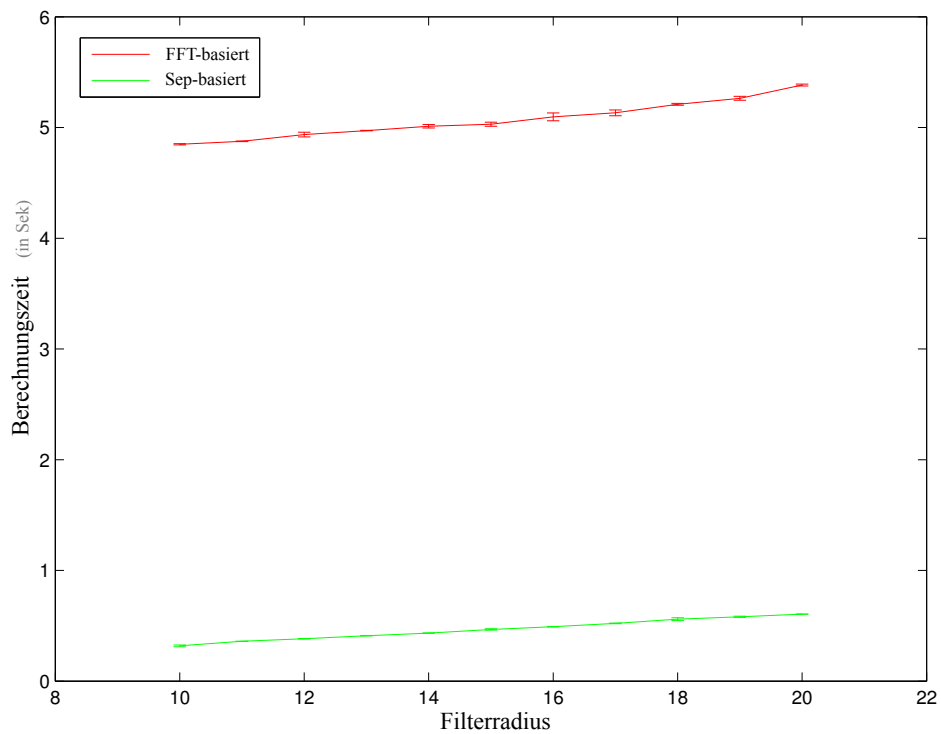


Abb. 2.5.: Berechnungszeit (Sek) für Mittelwertbestimmung per FFT-basierter Faltung (obere Werte, rot) und separierungs-basierter Methode (untere Werte, grün), als Funktion von variierenden Radien der Filter-Masken bei je 20 Versuchen. Das Eingangssignal war ein randomisierter Tensor 4ter-Ordnung (4D-Array) mit gleichen Kantenlängen und wurde gefaltet mit einem Mittelwert-Filter der Ordnung 4 und variierenden Radien (10 – 20).

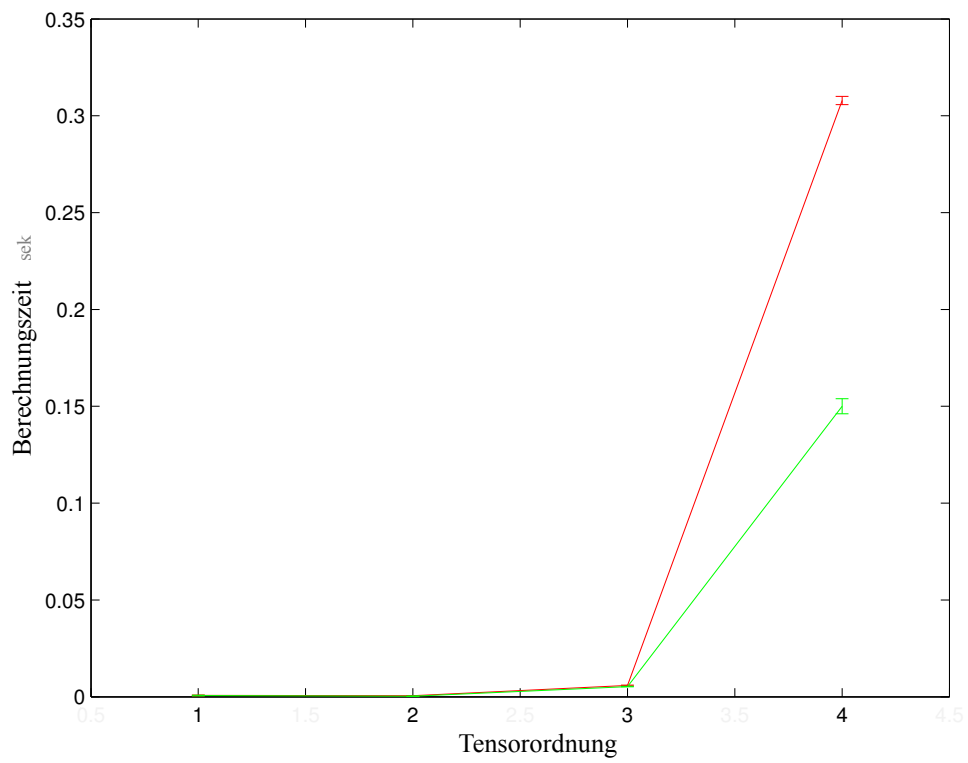


Abb. 2.6.: Berechnungszeit (Sek) für Mittelwertbestimmung per FFT-basierter Faltung (obere Werte, rot) und separierungs-basierter Methode (untere Werte, grün), als Funktion von variierenden Tensorordnungen (1 – 4) bei je 20 Versuchen.

2.3.6. Separierung und Nicht-Separierung des Medianfilters

Betrachtet und hinterfragt wurde die Separierung bisher bei linearen Filtern bzw. linearen lokalen Operatoren. Allerdings sind ebenso oft genutzte Operatoren - wie z. B. der Medianfilter [6] - nicht-linear.

Bei einem Medianfilter werden Sortierungen vorgenommen. Darum liegt hier eine superlineare Aufwandsklasse vor. Die Anzahl der Elemente eines Medianfilter-Tensors ist $\prod_{j=1}^d m_j$.

Eine Folge von d je 1-dimensionalen Medianfilter-Tensoren, mit Kantenlängen m_j , $1 \leq j \leq d$, entsprechen nicht unbedingt einer einmaligen d -dimensionalen Medianfilterung. Trotzdem ist eine Nutzbarkeit erkennbar.

Der Rechenaufwand, beschrieben mittels Landau-Kalkül, eines nicht-separierten Medianfilters, abhängig von Tensorordnung und Kantenlängen, beträgt daraufhin [140, 64, 84]:

$$\mathcal{O} \left(\log \left(\prod_{j=1}^d m_j \right) \cdot \prod_{i=1}^d m_i \right) \quad (2.25)$$

O. B. d. A. $m = m_1 \leq \dots \leq m_d$. Somit gilt für den Rechenaufwand weitergehend:

$$\mathcal{O} \left(\log \left(m^d \right) \cdot m^d \right) = \mathcal{O} \left(\log(m) \cdot d \cdot m^d \right) \quad (2.26)$$

Hingegen beträgt der Rechenaufwand eines separierten Medianfilters [107]:

$$\mathcal{O}(\log(m_1) m_1) + \dots + \mathcal{O}(\log(m_d) m_d) = \sum_{k=1}^d (\mathcal{O}(\log(m_k) m_k)) \quad (2.27)$$

O. B. d. A. $m = m_1 \leq \dots \leq m_d$. Somit gilt für den Rechenaufwand weitergehend:

$$\mathcal{O}(\log(m) \cdot d \cdot m) \quad (2.28)$$

Die Separierung nicht-linearer Filter, wie des Medianfilters (Gl. 2.26), kann durchaus schnellere Berechnungen gewährleisten (Gl. 2.28). Eine lineare Rechenaufwandsklasse, wie bei den linearen lokalen Operatoren (Gl. 2.19), (Gl. 2.18), ist hierbei allerdings nicht erreichbar.

Gerade in höheren Dimensionen ist bei solchen nicht-linearen Operatoren zwischen der Rechenzeit und der Qualität abzuwägen, wobei je nach Umstand die Qualität entsprechender separierter Operatoren sogar besser sein kann [101].

Deutlich wird es an einem einfachen Beispiel, der Filterung eines Segmentes in der Form eines Quaders und der daraufhin folgenden Visualisierung. Im zweidimensionalen wurde es schon an Beispielen von Rechtecken gezeigt [102, 107]. Wie in Abb. 2.7 und Abb. 2.8 ersichtlich, entsprechen die mittels nicht-separiertem Medianfilter bearbeiteten Ecken²³ nicht denen des Originals. Mit einem separierten Medianfilter (Abb. 2.9) bearbeitete Ecken entsprechen hingegen denen des Originals. Detaillierte Beschreibungen der Unterschiede und deren Ursachen sind in folgenden Schriften zu finden: [101, 102, 107].

Zur Bewahrung von Kanten und der Bearbeitung in möglichst kurzer Zeit empfehlen sich separierte Medianfilter. Eine stärkere Glättung kann wiederum mit nicht-separierten Medianfiltern erlangt werden.

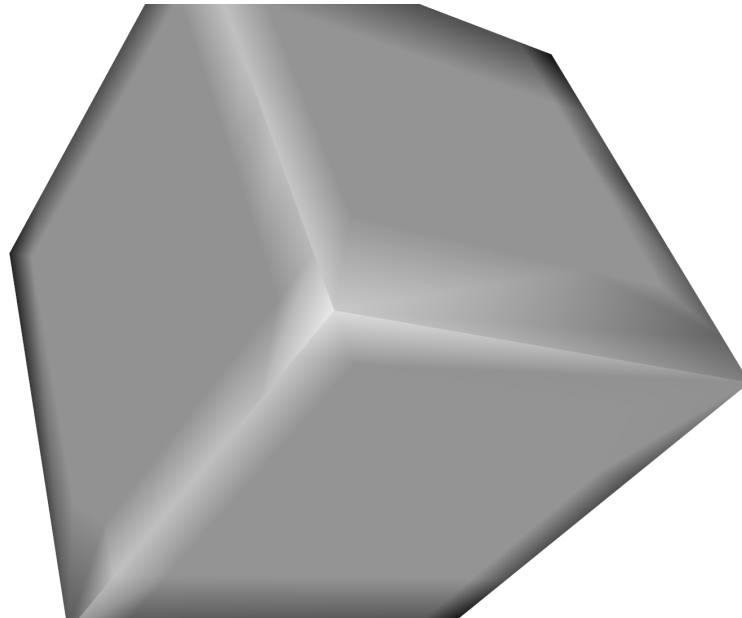


Abb. 2.7.: Original Volumendatensatz in Form eines Quaders

²³Ecken sind Schnittpunkte verschiedener, nicht parallel im Raum liegender Kanten

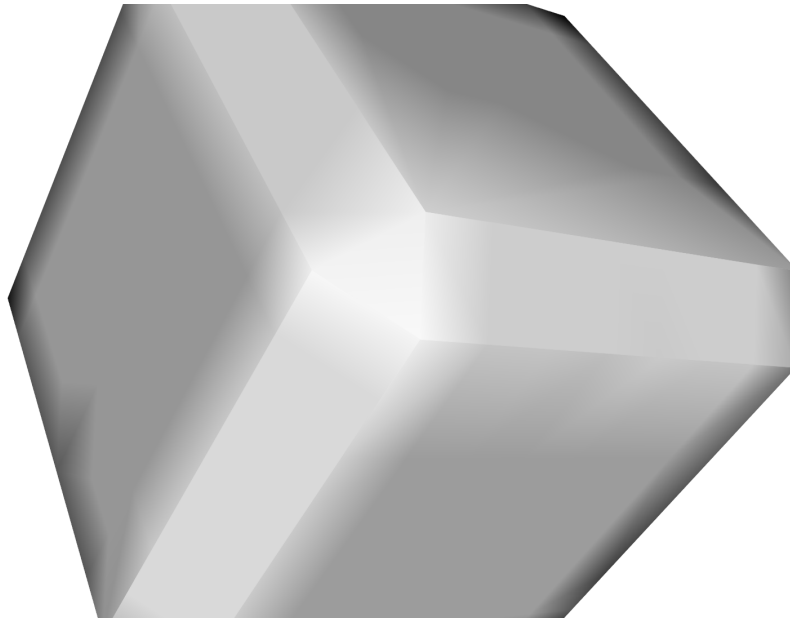


Abb. 2.8.: Ergebnis nach nicht-separierter dreidimensionaler Medianfilterung von Abb. 2.7.
MedianFilter-Tensorradius: $(1, 1, 1)$

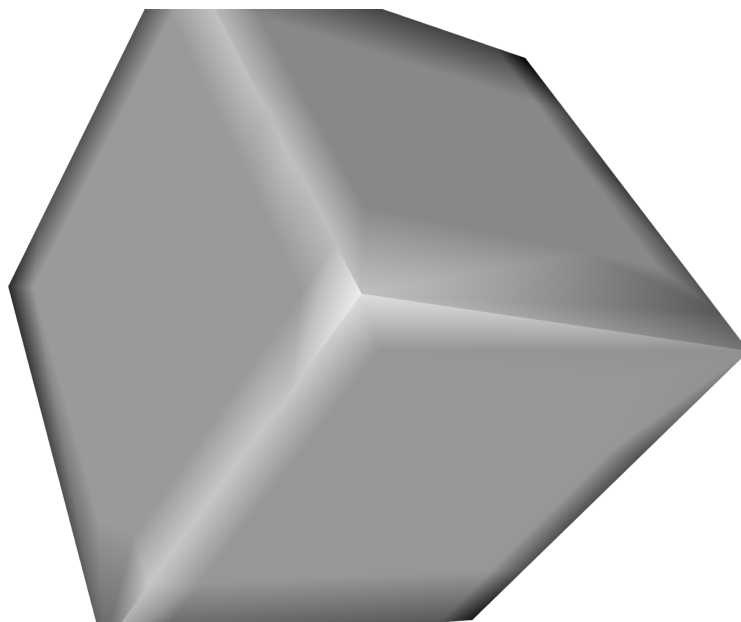


Abb. 2.9.: Ergebnis nach separierter dreidimensionaler Medianfilterung von Abb. 2.7.
MedianFilter-Tensorradien: $(1, 0, 0)$; $(0, 1, 0)$; $(0, 0, 1)$

2.4. Auswertung

Die Relevanz zur Bewältigung multidimensionaler Bilddaten kommt bei modernen biologischen Projekten immer mehr zur Geltung. Die in dem Abschnitt beschriebene Separierbarkeit multidimensionaler linearer lokaler Operatoren und dazugehörige Eigenüberlegungen (Abs. 2.2) bieten somit eine Möglichkeit – abhängig von der Dimension und den Kantenlängen – exponentiell wachsenden Rechenaufwand (Gl. 2.18) in linear wachsenden Rechenaufwand zu senken (Gl. 2.19).

Die Bearbeitung bzw. das Filtern multidimensionaler Bilddatensätze – wie z. B. Volumendaten (3D), Volumendaten über Zeit (4D) sowie Volumendaten über Zeit, mit unterschiedlichen Fluoreszenzfrequenzen (5D) – ist demzufolge mit linearem Berechnungsaufwand realisierbar.

Zum Resümee stellt sich die Frage, welche der bekannten oft genutzten Verfahren separierbar oder zumindest approximiert separierbar viel schneller zum selben Ergebnis führen. Aufgelistet sind eigene Überlegungen und entsprechende Fakten in folgender Tabelle (Tab. 2.3):

Bestimmte Filter	exakte Dekomposition	approximativ
Binomial	ja	ja
Gauß	ja	ja
Mean	ja	ja
Max	ja	ja
Min	ja	ja
Laplace	nein	ja
Marr-Hildrecht-Operator	nein	ja
Sobel	nein	ja
Prewitt	nein	ja
Robert	nein	ja
Varianz	nein	?
Canny	nein	?
Median	nein	ja
k-NN	nein	ja
δ -Nachbarschaft	nein	ja
Erosion	ja	ja
Dilatation	ja	ja
Opening	nein	ja
Closing	nein	ja

Tab. 2.3.: Filterverfahren und Überlegungen zu Möglichkeiten zur Separierung

Bei den *Binomial*-, *Gauß*-, und *Mean-Filter* handelt es sich um lineare lokale Operatoren, die separierbar sind [21].

Auch der *Maximal*- bzw. *Minimalwertfilter* kann als separierbar betrachtet werden. Schließlich gilt:

$$\text{Max}(\mathfrak{X}_{i_1, \dots, i_d}) = \text{Max} \left(\dots \left(\text{Max}(\mathfrak{X}_{i_1})_{i_2} \dots \right)_{i_d} \right),$$

ebenso wie:

$$\text{Min}(\mathfrak{X}_{i_1, \dots, i_d}) = \text{Min} \left(\dots \left(\text{Min}(\mathfrak{X}_{i_1})_{i_2} \dots \right)_{i_d} \right).$$

Laplace-, *Sobel*-, *Prewitt*-, *Robert-Filter* und auch der *Marr-Hildreth-Operator*²⁴ sind spezielle Filter zur Detektion von Kanten. Normalerweise handelt es sich um zweidimensionale Filtermasken, deren Prinzip jedoch auch auf multidimensionale Filtermasken übertragen werden kann [21, 64]. Die Summe von separierbaren Tensoren ist nicht direkt separierbar, aber eben mit einer Summe von separierbaren Tensoren exakt approximierbar [79].

Ein *Varianzfilter* dient zur nichtlinearen Kantendetektion. Auch er ist auf multidimensionale Räume erweiterbar. Die approximierbaren Möglichkeiten sind ein weiterer Forschungsansatz.

Bei dem *Canny-Filter* handelt es sich um einen Ablauf, der nicht einfach mit einem lokalen Operator durchführbar ist [23]. Es sind jedoch separierbare Operatoren (z. B. Gauß-Filter) enthalten, so dass Separierungen vorgenommen werden können. In weiteren Untersuchungen bietet es sich also an, mit Dekompositionen einen Canny-Filter zu approximieren.

Der *Median-Filter* ist ein nichtlinearer Filter, zum Minimieren von Rauschen, ohne gleichzeitig deutliche Kantenglättungen vorzunehmen. Der Filter gehört (wie Maximal- und Minimalwertfilter) zur Gruppe der Rangordnungsfiler. Im Gegensatz zu Maximal- und Minimalwertfilter ist der Median-Filter allerdings nicht direkt separierbar. Mittels einer Folge von eindimensionalen Median-Filtern für alle Dimensionen kann das Ergebnis eines normalen Median-Filters jedoch approximiert werden. Mehr Details sind hierzu in Abs. 2.3.6 beschrieben.

Der *k-NN-Filter* bzw. der Operator zur Bestimmung der δ -Nachbarschaft ist ebenso nicht separierbar – ist allerdings, wie in Form des *Median-Filters*, approximierbar, da auch hier Sortierungen je Dimension vorgenommen werden können.

²⁴*Laplacian of Gaussian (LoG); Mexican Hat; Sombrerofilter*

Morphologische Operatoren, wie *Erosion*, *Dilatation*, *Opening* und *Closing* sind nicht direkt separierbar wie ein normaler linearer lokaler Operator. Jedoch nutzen die *Erosion* und die *Dilatation* auf binärem Bilddatensatz ebenso Masken, die einem separierbaren Tensor entsprechen können [64]. Oft entspricht bei diskreten Datensätzen die Maske einem normalen Tensor mit fest vorgegebenen Radien. Der Operator entspricht daraufhin einer Verknüpfung, wie bei einem linearen lokalen Operator, nur, dass hier keine Summe, sondern eher ein Produkt vorliegt [21].

Erosion und *Dilatation* können also – je nach Maske – separiert werden [64]. *Opening* und *Closing* können daraufhin approximiert werden.

2.5. Anwendung multidimensionaler Filter

Mit Hilfe von multidimensionalen linearen lokalen Operatoren kann per Separierungsverfahren – bei selbem Ergebnis – exponentielle Bearbeitungszeit (Rechenaufwand) auf linearen Aufwand verringert werden (Kap. 2.2, Kap. 2.3).

Zum Vergleich wurde es an multidimensionalen Beispieldatensätzen durchgeführt (Kap. 2.3) [121, 122]. Wie sich an dem Beispiel zeigt, wurde der Zeitaufwand stark verringert, während das Ergebnis das Gleiche ist. Für entsprechenden Vergleich wurde die Differenz beider Ergebnisse betrachtet und ergab je Muxel Grauwertunterschiede von maximal 1, was auf Rundungsfehler zurückzuführen ist.

Weiterhin stellt sich die Frage, ob es überhaupt nötig ist – und wenn, warum – multidimensionale Filterungen vorzunehmen. Ist es nicht auch akzeptabel, einen 3D-Datensatz als Folge von 2D-Datensätzen zu betrachten und somit eine Reihe von 2D-Filterungen vorzunehmen?

Es wurde als Beispiel ein 3D-Datensatz von Zellen einmal mit einem 3D-Filter und einmal mit einer Folge von 2D-Filtern bearbeitet und anschließend das entsprechende Oberflächenmodell zur Visualisierung erstellt (Abb. 2.10).

Die 2D- und die 3D-Filterung kann separiert genutzt werden und deren Rechenaufwand befindet sich folglich in derselben Komplexitätsklasse [127] (linear). Jedoch, wie sich in Abb. 2.10 zeigt, ist die Qualität der Filterung bei der Folge von 2D-Filterungen stark verringert. Einzelne Artefakte und Störungen sind in Abb. 2.10 (ii), im Gegensatz zu Abb. 2.10 (i) noch enthalten. Hierbei handelt es sich um eine typische Gegebenheit, für die solche Glättungsfilter eingesetzt werden [64]. Die qualitative Verschlechterung wurzelt aus der Tatsache, dass in dem Fall bei den 2D-Filterungen Informationen aus der 3. Dimension vernachlässigt werden. Aus der Sicht der separierten Filterung kann eine 2D-Filterung bei einem 3D-Datensatz so betrachtet werden, dass lediglich noch der letzte separierte 1-D Filter fehlt.

Auch ist an der Oberfläche die fehlende Glättung in der 3. Dimension auf Grund des Gegensatzes von Abb. 2.10 (i) und Abb. 2.10 (ii) zu erkennen. Wie bei der Gegenüberstellung ersichtlich, fehlt somit der Bezug zwischen Punkten mit verschiedenen Werten in der 3. Dimension. Das erklärt die stufenartige Form in Abb. 2.10(ii) im Gegensatz zu Abb. 2.10(i) und die fehlerhaften Fragmente in Abb. 2.10(ii).

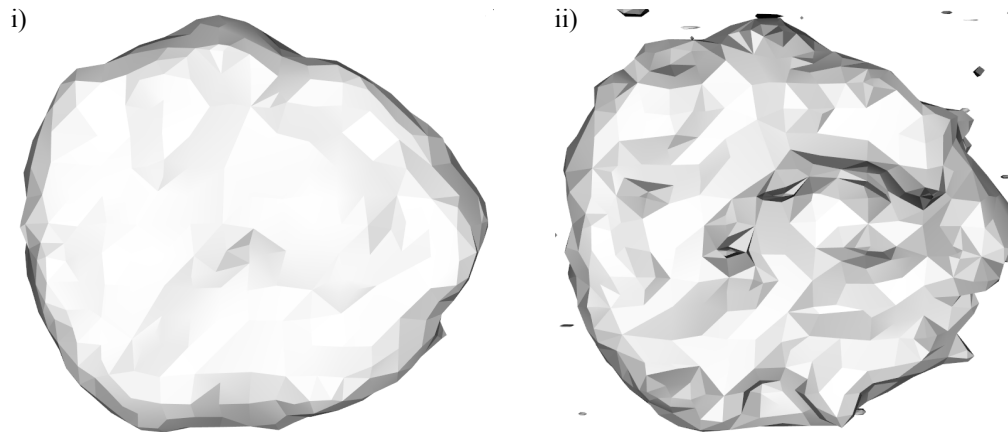


Abb. 2.10.: Abbild nach exakter Medianfilterung i.) 3D-Filter ii.) Folgen von 2D-Filtern

Realisiert sind multidimensionale Separierungen in *MATLAB* sowie auch in *C++*. Entsprechende Filter werden als Tensoren betrachtet. Von den Tensoren können Kantenlängen entsprechender Dimensionen festgelegt werden (Gl. 2.13).

Während ein d dimensionaler nicht-separierter Filter einer Faltung mit einem d dimensionalen k_1, \dots, k_d Tensor entspricht, kann die Nutzung des entsprechenden separierten Filters als Folge von 1dimensionalen k_1 bis k_d Tensoren betrachtet werden (Abb. 2.1). Dementsprechend wurde es im Quellcode realisiert.

3. Neue Verknüpfung multidimensionaler Segmentierungsverfahren

3.1. Problemdarstellung

Die Probleme der multidimensionalen Segmentierung biologischer Bilddatensätze liegen oft in der geringen Qualität der mikroskopischen Akquirierung (Abb. 3.1). Biologische Bilddatensätze sind relativ verrauscht und fehlerbehaftet. Die mikroskopischen Aufnahmen basieren auf optischen Verfahren, die erreichen sollen, biologisch relevante Objekte – möglichst *in vivo* – zu erkennen und auszuwerten (Abs. 1.2.1).

Trotz effizienter Bildbearbeitung bleiben vorhandene Fehler in der daran anschließenden Segmentierung häufig bestehen [142, 16, 20, 12, 29, 93, 117, 121, 145]. Es gibt bereits eine Vielzahl spezieller Techniken, mit denen es möglich ist, problem-spezifisch biologische Objekte (Zellen, Synapsen usw.) detailliert zu segmentieren [5, 27, 66, 143, 146, 152, 104]. Doch auch bei den hier genannten Techniken treten wiederum eine Reihe verschiedener Problemen auf (z. B. Rauschen, Über- und Untersegmentierung), die in der Dissertation erläutert werden.

In Abb. 3.1 sind mit Hilfe eines Fluoreszenzmikroskopes aufgenommene Originalbilder eines Zebrabärblings abgebildet. Die bereits beschriebenen Daten von sich entwickelnden Zebrabärblingen werden in dem Kapitel für Experimente genutzt. Abb. 3.1(i) verdeutlicht eine aufgenommene 2D-Schicht. Während Abb. 3.1(ii) der Projektion eines gesamten 3D-Datensatzes entspricht. In beiden Bildern ist das starke Rauschen ersichtlich. Auch die relativ geringe Deutlichkeit kommt zum Tragen, so dass es sogar manuell und erst recht (semi-)automatisch schwierig ist, zu entscheiden, wo genau und wie viele Zellkerne bei dem Bilddatensatz vorliegen.

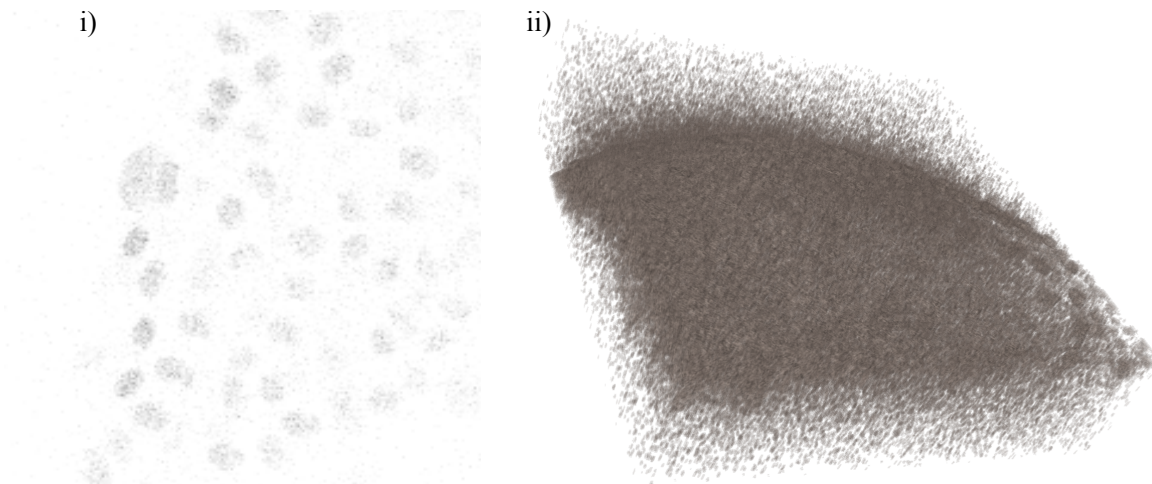


Abb. 3.1.: Stark verrauschtes Ursprungsbild i) 2D-Abbild einer Schicht ii) 3D-Projektion einer Menge von Schichten

Bereits existierende Verfahren haben ihre individuellen Nachteile:

- Es können Segmente verloren gegangen sein (Untersegmentierung).
- Mehrere Segmente können fälschlicherweise miteinander fusionieren (Übersegmentierung).
- Die Form der Segmente kann sich deutlich von der des Originals unterscheiden.
- Die Kontraste, Helligkeiten können sich deutlich geändert haben.

In der vorliegenden Dissertation sollen unterschiedliche Segmentierungsverfahren verknüpft und die jeweils unterschiedlichen Nachteile gemindert werden. Unterschiedlich segmentierte Abbildungen des Ursprungsbildes sollen also parallel genutzt werden, um bestimmte biologische Objekte besser zu segmentieren und die Vorzüge einzelner Segmentierungsverfahren zu vereinen (Abs. 3.2.2) [112, 28].

Es handelt sich hierbei nicht um eine typische Bildfusion, in der verschieden akquirierte Bilder eines Objektes miteinander verknüpft werden [81]. Vielmehr liegt hier jeweils ein einziges Originalbild vor, das mit mehreren, unterschiedlich durchgeführten Verarbeitungsabläufen kombiniert wird.

In den kommenden Abschnitten wird infolgedessen ein eigenes Lösungsverfahren zur Verknüpfung mehrerer Segmentierungsverfahren beschrieben (s. Abs. 3.2). Anschließend wird eine Reihe verschiedener Segmentierungsverfahren einzeln vorgestellt und getestet (vgl. Abs. 3.2.1). Der daran anschließende Abschnitt beschreibt schließlich, wie die vorliegenden Verfahren experimentell miteinander zu verknüpfen sind und die Verknüpfung auszuwerten ist (vgl. Abs. 3.2.2).

3.2. Lösungsverfahren

Jedes Segmentierungsverfahren hat Vor- und Nachteile. In dem vorliegenden Abschnitt der Dissertation wird versucht, die jeweiligen Vorteile vier verschiedener Segmentierungsverfahren (s. Abs. 3.2.1) [*Otsu-Schwelwertfilter* (geschichtet), *Wasserscheidentransformation*, *Confidence-Connected Regionenwachstum* und *Laplacian Level-Set*] miteinander zu verknüpfen und deren Nachteile gering zu halten. Dafür werden spezifische Verfahren ausgewählt, verglichen und unterschiedlich miteinander verknüpft. Vier unterschiedliche Verknüpfungsverfahren [*Arithmetisches Mittel*, *Geometrisches Mittel*, *Median-Verknüpfung*, *Fuzzy C-Means (FCM)*] werden zur Fusionierung entsprechender Segmentierungen getestet und evaluiert. Getestet wird das Konzept an einem biologischen Bilddatensatz von Zebrabärblingen. In einem Experiment sollen daraufhin Zellkerne automatisch ermittelt werden (s. Abs. 1.2.4).

3.2.1. Evaluierung verschiedener Segmentierungen für Zellkerne

Im vorliegenden Abbild (s. Abb. 3.2) befinden sich Ergebnisse von vier verschiedenen Segmentierungsverfahren, der Mikroskopbilder eines Zebrabärblinges. Die Abläufe der Segmentierungsverfahren, sowie deren Besonderheiten, werden später in eigenen Abschnitten beschrieben.

Otsu-Schwelwertfilter

Der *Otsu-Schwelwertfilter* [103]¹ ist ein schnell zu berechnender und vollautomatischer Filter. Der entsprechende Schwellwert wird in einem festdefinierten Verfahren ermittelt und angewendet. Das Verfahren ist relativ schnell (s. Tab. 3.3, S. 67), da komplexe Berechnungen nicht notwendig sind. Lediglich der Schwellwert – abhängig von der minimalen Varianz der jeweils entsprechenden Segmente – wird berechnet und angewandt.

Die Qualität der Ergebnisse hängt allerdings stark von den vorherigen Glättungen ab. Starke Glättungen können nach der *Otsu-Schwelwertfilterung* zu weniger Segmenten führen, weil weniger kleine Artefakte und somit weniger kleine Segmente durch Rauschen entstehen. Auch ist es wahrscheinlicher, dass infolgedessen mehr Segmente miteinander verschmelzen.

Außerdem kann es vorkommen, dass bei niedrigen Schwellwerten Objekte bzw. Segmente verloren gehen. Zu hoch gewählte Schwellwerte führen dagegen dazu, dass nahe aneinander liegende Zellkerne des Segments miteinander verschmelzen.

¹<engl> Otsu threshold filter

Bedingt durch das genutzte Akquisitionsverfahren (Konfokale Mikroskopie [108, 132, 62, 22, 114, 98]) haben Abbildungen in unteren Schichten eine schwächere Intensität, als Abbildungen in oberen Schichten. Automatisierte Schwellwertfilter – wie z. B. der *Otsu-Schwellwertfilter (global)* (Abs. 3.2.1), mit globalem Schwellwert – erzielen folglich keine guten Ergebnisse.

Bei niedrigen Schwellwerten werden schwach sichtbare Zellkerne in unteren Schichten gut bestimmt. In den oberen Ebenen werden Zellkerne jedoch eher miteinander verschmolzen oder sogar ignoriert. Bei zu hohen Schwellwerten gehen hingegen schwach sichtbare Zellkerne eher verloren.

Um die Probleme zu umgehen, wird für jede Ebene ein eigener Schwellwert bestimmt. Die Absicht liegt darin, weiter unten liegende, schwach sichtbare Zellkerne zu bestimmen und trotzdem in den oberen Ebenen eine akzeptable Segmentierung der einzelnen Zellkerne zu erreichen.

In jeder einzelnen Ebene kann also ein normaler *Otsu-Schwellwertfilter* angewandt werden. Allerdings werden so, in Ebenen, in denen eigentlich keine Zellkerne vorhanden sind, Segmente fehlerhaft zugeordnet (Übersegmentierung). Der Grund liegt darin, dass eine Entscheidung für einen Schwellwert getroffen werden muss, was dazu führt, dass eine Menge Muxel als Element eines Segmentes zugeordnet wird, obwohl gar kein Segment vorliegt. Umgangen wird das Problem, indem die minimale und maximale Intensität auf jeder Ebene ermittelt wird. So kann zuverlässiger abgeschätzt werden, ob tatsächlich Zellkerne vorliegen.

Trotz alledem besteht auch beim vorliegenden Verfahren immer noch das Problem, dass die Mehrzahl der Zellkerne nicht korrekt voneinander getrennt und folglich zu einem einzelnen Segment verbunden werden, wie in Abb. 3.2(i) ersichtlich (Untersegmentierung) [121].

Wasserscheidentransformation

Die *Wasserscheidentransformation* wurde gewählt, um Kanten hervorzuheben Abb. 3.2(ii). In dem Verfahren werden Gradienten bestimmt, um die entsprechenden Intensitätsänderungen zu ermitteln [92, 86, 64, 51, 73].

Jedoch treten hierbei folgende Probleme auf (s. Abb. 3.2(ii)):

- schwach sichtbare Zellkerne
- Übersegmentierung
- große, inhomogene Zellkerne werden als mehrere Segmente erkannt

- aneinanderliegende Zellkerne verschmelzen

Darüber hinaus besteht das Problem der Parametrisierung. Entweder gehen schwach sichtbare Zellkerne verloren oder es kommt zu einer Übersegmentierung (s. Abb. 3.2(ii)) [121].

Regionenwachstum

*Regionenwachstum*² basiert auf dem Verfahren von Saatpunkten ausgehend, mit vorgegebenen Eigenschaften zu wachsen und so Segmente zu bestimmen [4].

Confidence-Connected Regionenwachstum ist ein derartiges *Regionenwachstum*-Verfahren. Iterativ werden die Muxel, die sich am äußeren Rand des bisherigen Segmentes befinden, untersucht und je nach Eigenschaft dem Segment beigefügt. Es wird bestimmt, ob sich der Wert des Muxels innerhalb eines festgelegten Bereiches befindet. Die Spanne entspricht dem mittleren Grauwert aller anliegenden Muxel und der Differenz zu einer festgelegten Gewichtung der entsprechenden Standardabweichung. Das Verfahren wird mit drei Parametern festgelegt:

- Anzahl der Wiederholungen des Verfahrens
- Multiplikator zur Standardabweichung
- Größe der Umgebung der anliegenden zu bestimmenden Muxel

Einzelne Segmente können mit derartigen Verfahren relativ einfach bestimmt werden. Werden anschließend manuelle Korrekturen verschiedener Parameter vorgenommen (z. B. Saatpunkte, Schwellwerte, Varianzen), sind gute Segmentierungen möglich [64].

Da unterschiedliche Segmente jedoch unterschiedliche Intensitäten aufweisen, ist es schwer, für alle zu ermittelnden Segmente die gleichen Parameter zu verwenden. Zudem tritt häufig das Problem auf, dass sich nahe beieinanderliegende Segmente berühren und schwer zu unterteilen sind. Außerdem kommt es zu Segmenten mit unerwünschten Hohlräumen. Parametrisierungen, die für einzelne Segmente geeignet sind, können bei anderen Segmenten zum Auslaufen führen – was bedeutet, dass sich das Segment zu sehr ausweitet oder im Bildhintergrund unerwünscht weiterentwickelt.

Zur Verbesserung können mehr Parameter beigefügt werden. Z. B. können die minimalen und die maximalen Segmentgrößen, oder aber Eigenschaften, die *Active Contour Models (ACM)* [67, 33, 155] oder denen spezieller *Level-Sets* [99] entsprechen, als Parameter festgelegt werden. Das ist wegen der steigenden Parametrisierung mit einem hohen Rechenaufwand verbunden.

²<engl> region growing – <dt> Regionenwachstum

Auf Grund oft vorliegender Abweichungen (z. B. Helligkeit oder Rauschen) ist es problematisch, geeignete Kriterien bzw. Parameter für das *Regionenwachstum* festzulegen. Ein einfacher Schwellwert als Kriterium für die iterative Integration der Muxel zur segmentierten Region ist somit nicht ausreichend. Derartige Werte manuell zu ermitteln, ist ebenso nicht praktikabel, da der Einstellungsaufwand exponentiell steigen würde (s. Abb. 3.2(iii)) [121, 113].

Laplacian Level-Set

Laplacian Level-Set ist ein *Level-Set*-Verfahren, das sich – von einem Startsegment ausgehend – auf einem kantenbetonten Abbild mit gegebenen Parametern iterativ entwickelt und anpasst [63, 99]. Wie beim *Regionenwachstum* oder beim *Active Contour Model (ACM)* werden Saatpunkte bzw. Startsegmente als Segmentierungsstartpunkte benötigt. Im optimalen Fall wird jedem Zellkern ein Saatpunkt zugeordnet. Um jeden der Saatpunkte wird nun ein Bereich definiert, der als jeweiliges Startsegment dient. Ausgehend von allen Startsegmenten wird jeweils eine *Level-Set*-Segmentierung durchgeführt, die sich entsprechend den Kanten und Formen bildet. Saatpunkte, die innerhalb eines anderen schon gefundenen Segmentes liegen, werden ignoriert. Die am Ende vorliegenden Segmente werden wieder zu einem einzigen Datensatz verknüpft.

Das *Laplacian Level-Set* ist eine Erweiterung des *Regionenwachstums* Abs. 3.2.1 und wurde ursprünglich entwickelt, um Gesichter durch *Active Contour Model (ACM)* zu erkennen [67]. In der vorliegenden Arbeit wird daraufhin ein Ansatz entwickelt, um multidimensionale biologische Bilddaten zu segmentieren [121].

Die Weiterentwicklung kann als rekursiver Prozess implementiert werden. Das Segment wird als diskrete Menge von Elementen betrachtet, deren Verhalten bezogen auf Modelle und physikalische Eigenschaften immer wieder neu bestimmt wird [63].

Der klare Vorteil des Verfahrens liegt in der Erweiterung und Ermöglichung der problemspezifischen und modellbasierten Parametrisierung. Die Parametrisierung steht in Bezug zur Physik (Kräfte, Energie) und deren Einwirkung (s. Abb. 3.2).

Probleme können jedoch in vorliegenden biologischen Experimenten trotzdem auftreten, besonders wenn nicht einzelne Segmente, sondern eher eine große Menge von multidimensionalen Segmenten ermittelt werden. Hier können zwar nicht so leicht und ausführlich, wie bei der Wasserscheidentransformation Abs. 3.2.1, beim *Regionenwachstum* Abs. 3.2.1 oder beim *Otsu-Schwellwertfilter* Abs. 3.2.1, Untersegmentierungen auftreten. Dennoch können solche falschen Segmentierungen vereinzelt auch hier vorkommen. Mehrere Segmente können so als ein einzelnes Objekt gedeutet und auch die Form (z. B. Löcher oder Heterogenität) zu untersuchender Segmente kann fehlerhaft bestimmt werden (Abb. 3.2).

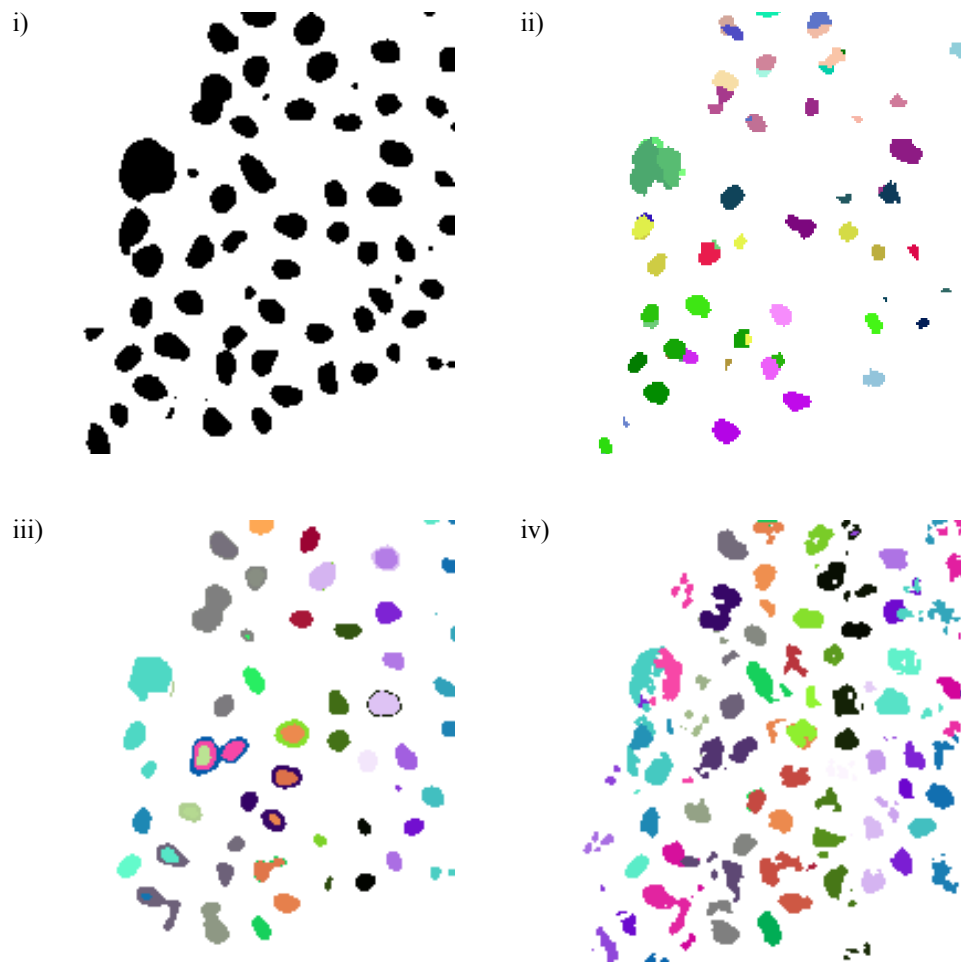


Abb. 3.2.: Gegenüberstellung verschiedener Segmentierungsverfahren am Beispiel der Verarbeitung von Zellkernbildern [121]. Die verschiedenen Farben in der Abbildung dienen lediglich der Visualisierung, sind ansonsten allerdings zu vernachlässigen.
i) Schwellwertfilter ii) *Wasserscheidentransformation* iii) *Confidence-Connected Regionenwachstum* iv) *Laplacian Level-Set*

3.2.2. Verknüpfung mehrerer Segmentierungsansätze

Die h folgenden Filterungen und Segmentierungen führen, basierend auf dem Originalbild $O(X)$, zu entsprechenden Zwischenergebnissen $O'_1(X), \dots, O'_h(X)$. Die Zwischenergebnisse und das Originalbild können anschließend miteinander verknüpft werden. Verknüpft werden hier allerdings nicht Segmente oder Abschnitte, sondern jeder einzelne Muxel, um schließlich eine Klassifikation der einzelnen Elemente eines Bildes vorzunehmen und danach ein neu segmentiertes Bild aufzubauen.

Ähnliche Ansätze zur Fusionierung verschiedener Bilddaten sind in [59, 119, 52, 80] enthalten. Der Neuheitswert in der Dissertation liegt allerdings in der speziellen Wahl und dem Vergleich der verschiedenen Verfahren, sowie der Anwendung zur direkten Unterscheidung von Segment und Nicht-Segment, angewandt auf multidimensionale biologische Bilddatensätze.

Damit die verschiedenen Abbildungen auch einheitlich verknüpft werden können, werden die Muxel einheitlich normiert. Als Wertebereich wird der maximale Wertebereich der vorliegenden Abbildungen gewählt. Bei normiertem Grauwertbereich $[0, \dots, 255]$ entspricht z. B. ein angepasster Binärdatensatz den Werten 0 und 255.

Hier sei betont, dass das Verfahren nicht nur für die Segmentierungen binärer Datensätze geeignet ist, sondern auch für nicht-binäre. Lediglich die Klassifikation der einzelnen Muxel ist binär. Jedem Muxel wird zugeordnet, ob es dem Hintergrund oder einem Segment (z. B. einer Zelle) angehört.

Das erlangte Ergebnis kann darüber hinaus als Träger genutzt und mit zuvor gefilterten Bilddaten gefaltet werden. Fehler und Grauwerte aus dem Hintergrund werden somit beseitigt.

Es werden vier verschiedene Verknüpfungen mit den vier in Abs. 3.2.1 behandelten Segmentierungsansätzen und dem Originalbild getestet und ausgewertet. Bei den Segmentierungsansätzen handelt es sich um Verfahren mit ihren eigenen Vor- und Nachteilen. Das Originalbild wurde ergänzend mit beigefügt, um zu zeigen, dass sich, trotz zusätzlicher Verknüpfung mit gering einwirkenden Verfahren (im Extremfall, eben das Originalbild selbst), eine akzeptable Gesamtverknüpfung ergeben kann.

Bei den hier hinterfragten und getesteten Verknüpfungen (s. Abb. 3.3) handelt es sich um:

- das arithmetische Mittel (s. Abs. 3.2.2),
- das geometrische Mittel (s. Abs. 3.2.2),
- den Median (s. Abs. 3.2.2) und
- den *Fuzzy C-Means (FCM)* (s. Abs. 3.2.2).

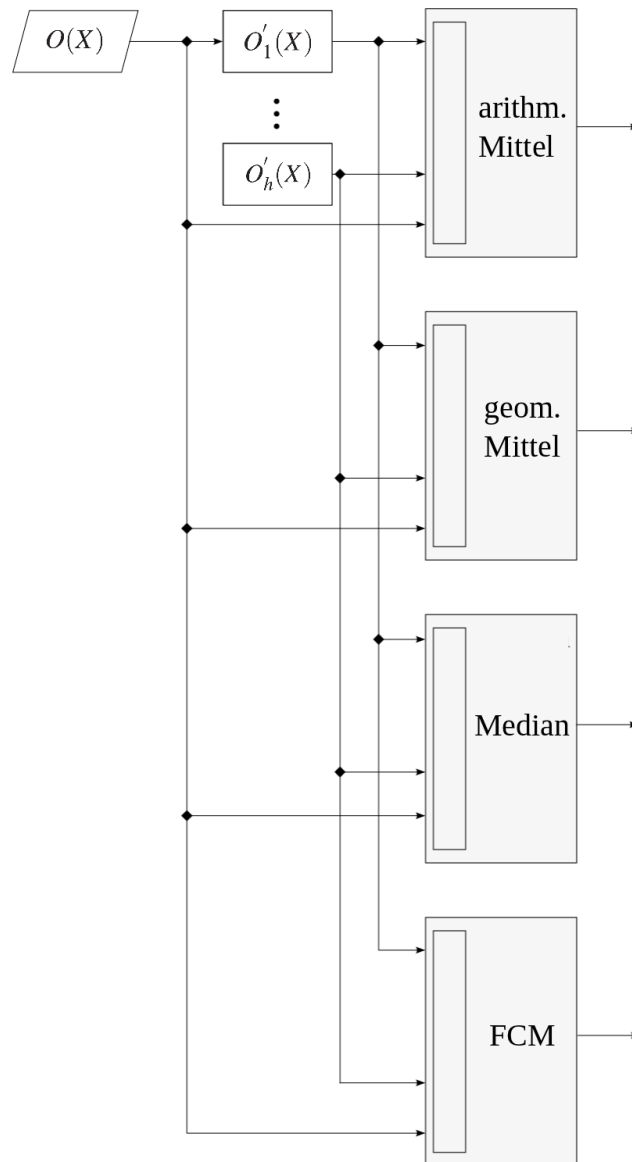


Abb. 3.3.: Verschiedene Verknüpfungsabläufe (Arithmetisches Mittel, Geometrisches Mittel, Median, *Fuzzy C-Means (FCM)*); O. B. d. A.: $O'_1(X)$ - Originalbild, $O'_2(X)$ - Otsu-Schwellwertfilter, $O'_3(X)$ - Wasserscheidentransformation, $O'_4(X)$ - Regionenwachstum, $O'_5(X)$ - Laplacian Level-Set

Arithmetisches Mittel

Jedes Muxel i_1, \dots, i_d eines Bildes wird mit den gleich positionierten Muxeln aller anderen Bilder addiert und normiert. Es handelt sich hier also um ein arithmetisches Mittel zwischen den gleichpositionierten zueinandergehörigen Muxeln.

Ein arithmetisches Mittel wird für einen typischen Durchschnittswert genutzt und kann mit Hilfe der Addition relativ schnell berechnet werden. Auch negative Werte sind anwendbar. Das arithmetische Mittel ist jedoch nicht robust und kann mit einzelnen Ausreißern stark beeinflusst werden.

Geometrisches Mittel

Jedes Muxel i_1, \dots, i_d eines Bildes wird mit den gleich positionierten Muxeln aller anderen Bilder multipliziert. Im Anschluss wird daraus die d te Wurzel gezogen. Es handelt sich hier um ein geometrisches Mittel zwischen den gleichpositionierten zueinandergehörigen Muxeln.

Mit einem derartigen geometrischen Mittel kann ein Wachstumsfaktor bestimmt werden. Der Rechenaufwand ist jedoch auf Grund der Rechenoperatoren (Multiplizieren und Radizieren) deutlich höher, als der des arithmetischen Mittels (Abs. 3.2.2). Des Weiteren kann hier kein negativer Wert, wie beim arithmetischen Mittel, genutzt werden. Bei einem einzelnen Nullwert wird das gesamte geometrische Mittel auch Null.

Median-Verknüpfung

Zwischen den Muxeln i_1, \dots, i_d jeden Bildes wird der, sich aus einer Sortierung ergebende, Median bestimmt. Der Median ist, im Gegensatz zu arithmetischen oder geometrischen Mittelwerten, relativ robust gegenüber Ausreißern (extrem abweichenden Werten). Weiterhin können negative Werte eingesetzt werden. Ein großer Nachteil liegt allerdings in dem deutlich höheren Rechenaufwand, gegenüber dem des geometrischen und erst recht des arithmetischen Mittelwertes.

Fuzzy C-Means (FCM)

In Zusammenarbeit mit der *Information School, Jingnan University* (Wuxi, China) wurde ein Verfahren konzipiert, welches mittels eines Clustering-Verfahrens – dem *Fuzzy C-Means (FCM)*-Algorithmus [40, 11, 94, 96] – jedem einzelnen Muxel ein entsprechendes Segment zuordnet. *Fuzzy C-Means (FCM)* ist eigentlich ein Verfahren der unüberwachten Klassifikation. Es wird in der Dissertation in entsprechenden Segmenten zur Klassifikation jedes einzelnen Muxels und somit zur Segmentierung genutzt.

Wie in Abs. 1.2.3 beschrieben, dient die Segmentierung der Unterteilung von Segmenten nach bestimmten Kriterien. Um jeweils verschiedene Vorzüge zu nutzen, werden eine Menge von $b \in \mathbb{N}$ unterschiedlich bearbeiteten Abbildungen erstellt. Derartige Abbildungen ergeben sich durch z. B. Filter oder Eigenschaftstensoren.

Die Menge von b bearbeiteten multidimensionalen Bilddaten wird jeweils in einem Vektor abgebildet. Die Länge solch eines einzelnen Vektors $v_i(X), i \in \mathbb{N}, 1 \leq i \leq b$ entspricht folglich der Anzahl von $C(X)$ Muxel der ursprünglichen Abbildung X .

Die b Vektoren $(v_1(X), \dots, v_b(X))$ ergeben eine Menge von $C(X)$ Punkten im b dimensionalen Raum.

Aus den $C(X)$ Elementen werden nun, mittels des *Fuzzy C-Means (FCM)* Algorithmus, zwei Cluster erstellt, die allgemein den Segmenten bzw. dem Hintergrund entsprechen. Die Elemente werden den Clustern so zugeordnet, dass die Summe der Distanzen zwischen den Clustern und den ihnen zugeordneten Elementen minimal wird. Als Fuzzifier wurde der Wert zwei, für die euklidische Distanz, gewählt [93].

Eine Besonderheit des *Fuzzy C-Means (FCM)* Algorithmus liegt darin, dass einzelne Elemente gewichtet werden – d. h., es wird nicht nur direkt festgelegt, zu welchem Segment ein entsprechendes Element gehört (wie z. B. beim K-Means-Algorithmus), sondern außerdem die fuzzifizierte Gewichtung beigefügt. Der *Fuzzy C-Means (FCM)* Algorithmus dient im vorliegenden Fall einer besseren Approximation. Die Clustereinstellung selbst ist wiederum scharf [112, 94, 28, 11].

Der Ablauf kann nun zu binären Segmentklassifikationen genutzt werden. Eine Klasse steht für existierende Bildsegmente, die andere Klasse für nicht vorhandene Bildsegmente, die dem Hintergrund entsprechen. Es sollen also zwei Cluster ermittelt werden, aus denen sich die Klassenzuordnung ergibt. Um welches Bildsegment es sich folglich handelt, wird zunächst nicht weiter beachtet.

Defuzzifiziert wird mit dem größeren Zugehörigkeitswert. Auf weitere Filterungen und mögliche Verbesserungen wird nicht weiter eingegangen. Das bietet allerdings eine Basis, um weiteren Forschungsansätzen nachzugehen.

Mit der beschriebenen Methode können Bildrauschen und mit Artefakten bestückte biologische Bilddatensätze verarbeitet werden. Artefakte und Rauschen bei verschiedenen Intensitäten konnte im Vergleich zu den anderen hier beschriebenen Verfahren am Besten bearbeitet werden. Hierzu dient zwar oft die Bildbearbeitung (Abs. 1.2.3), doch das allein ist nicht immer ausreichend, um einen Bilddatensatz in akzeptabler Qualität zu analysieren.

Dank des *Fuzzy C-Means (FCM)* Algorithmus können die Vorteile verschiedener Verfahren nun verknüpft und Nachteile verschiedener Verfahren reduziert werden.

Im Gegensatz zum *K-Means* Algorithmus, der vom Grundprinzip durchaus dem *Fuzzy C-Means (FCM)* Algorithmus entspricht, können beim *Fuzzy C-Means (FCM)* Algorithmus Unterschiede darüber hinaus in Form von Distanzen (z. B. euklidischen Distanzen) auf verschiedene Weise gewichtet werden. Praktisch angewandt wird hierfür das Matlabtool *fuzzy c-means clustering* [28, 112].

3.3. Auswertung

Zur Auswertung des beschriebenen Segmentierungsverfahrens wurde der in Abs. 1.2.4 bereits beschriebene – mittels Fluoreszenzmikroskopie akquirierte – Bilddatensatz von einem Fischbärbling gewählt. In der ausgeschnittenen 2D-Abbildung Abb. 3.4 sind einzelne fluoreszierende Zellkerne zu erkennen. Aber auch die starken Kontrastunterschiede sowie das Rauschen und Fehler kommen in dem Bild gut zur Geltung. Mit dem Beispieldatensatz sollen die verschiedenen, beschriebenen Segmentierungsverfahren direkt gegenübergestellt und getestet werden.

Bei der Segmentierung muss stets die Balance zwischen Qualität, Quantität und Zeit gehalten werden. In den letzten Jahren wurden viele Forschungsaufgaben zum Thema Segmentierungsverfahren in Angriff genommen [104, 142, 20, 29, 93, 117, 145, 5, 146]. Als Basis zum Vergleich werden hier stets manuelle Segmentierungen mit (semi-)automatischen Verfahren verglichen.

Bei der Auswertung in der vorliegenden Arbeit werden manuelle Segmentierungen (s. Abb. 3.5(ii)) beim Originalbild (s. Abb. 3.5(i)) vorgenommen. Hierbei werden allerdings nicht direkt ganze Segmente, sondern jeweils ein in der Mitte befindlicher Punkt gewählt. Beim Vergleich der Segmentierungsverfahren stellen Anzahl und Positionen der Segmente die beiden Eigenschaften dar, die bei den vorgestellten Segmentierungsverfahren gegenübergestellt werden (s. Tab. 3.3).

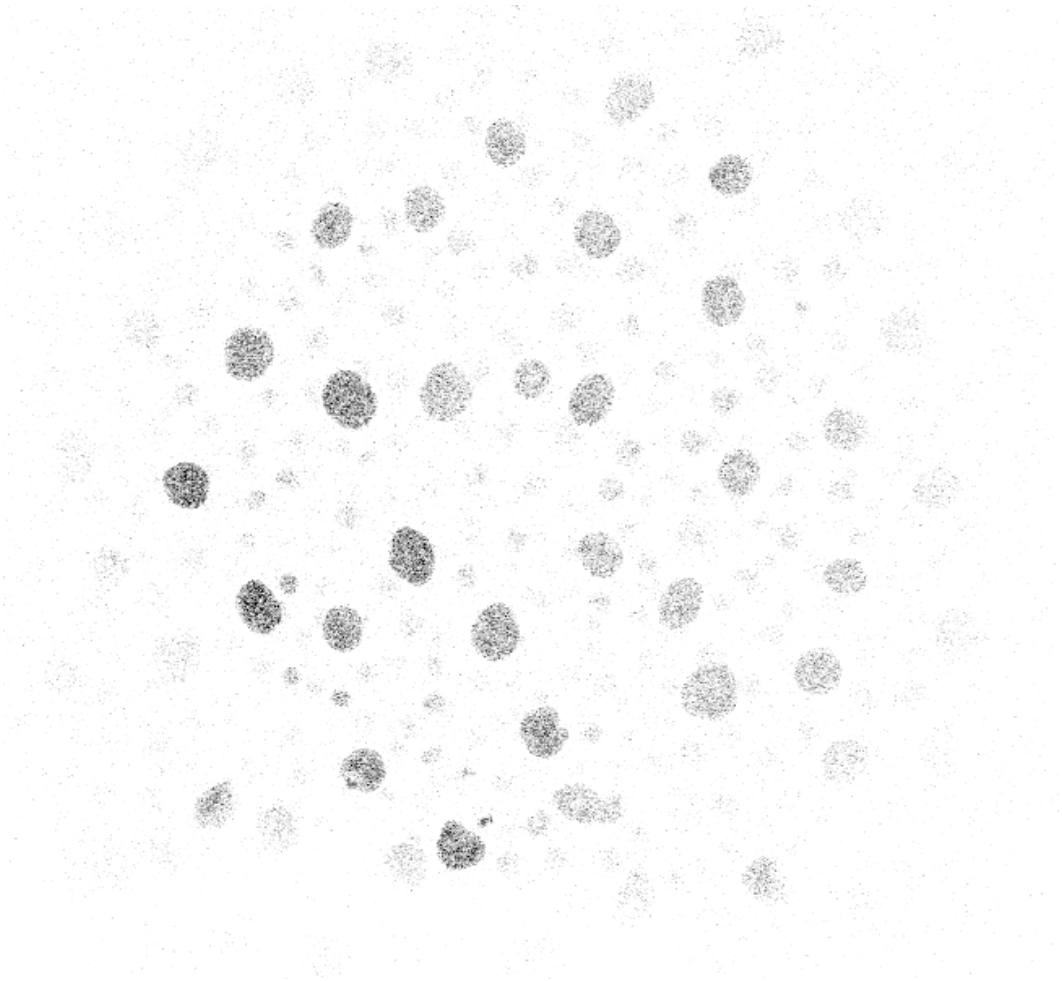


Abb. 3.4.: 2D-Abbildung aus der 75. Ebene vom Originaldatensatz

So können die Anzahl der bestimmten Segmente und deren Positionen miteinander verglichen werden. Das Ergebnis vom geschichteten *Otsu-Schwelwertfilter* (s. Abb. 3.5(iii)) ist, im Gegensatz zu den anderen Verfahren, relativ uneinheitlich und fehlerhaft. Während die *Wasserscheidentransformation* (s. Abb. 3.5(vi)) selbst bei Übersegmentierungen das Erhalten von Kanten ermöglicht, kann mit dem *Confidence-Connected Regionenwachstum* (s. Abb. 3.5(v)) oder dem *Laplacian Level-Set* (s. Abb. 3.5(iv)) eine relativ gute Form einzelner Segmente bewahrt werden (Abb. 3.4 [121]).

Die in der vorliegenden Arbeit angewandten Verknüpfungen der Segmentierungsverfahren (s. Abs. 3.2.2) weisen unterschiedliche Vor- und Nachteile auf. In den Ergebnissen der hier durchgeführten Experimente werden die Vorzüge der verschiedenen Verknüpfungsmöglichkeiten einzelner Segmentierungsverfahren ersichtlich.

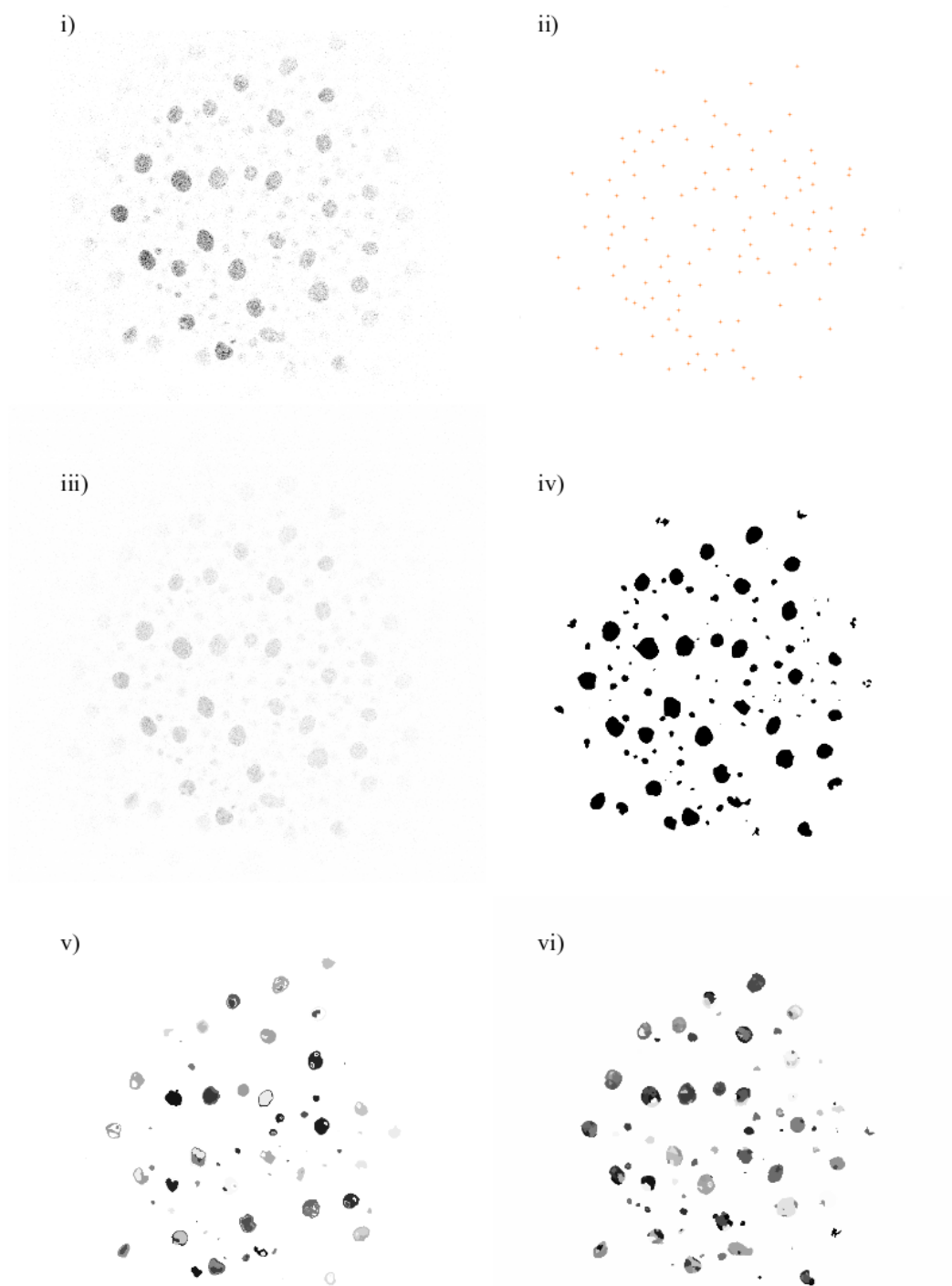


Abb. 3.5.: 2D-Abbildung aus der 75. Ebene verschiedener segmentierter Datensätze. i) Originalbild ii) manuelle Punktauswahl aus erkanntem Segment (d. h., ein Punkt je Segment) iii) *Otsu-Schwelwertfilter* (geschichtet) iv) *Laplacian Level-Set* v) *Confidence-Connected Regionenwachstum* vi) *Wasserscheidentransformation*

Mittels der hier vorliegenden Verknüpfungen der Ergebnisse (Abs. 3.2.2) kann nicht nur die Parametrisierung einzelner Segmentierungsverfahren verbessert werden, sondern auch die Gewichtung der einzelnen Verfahren zueinander.

Das Originalabbild wurde mit den Abbildungen der vier zuvor beschriebenen Segmentierungsverfahren (Abs. 3.2.1), wie in Abb. 3.3 dargestellt, verknüpft. In Tab. 3.2 wird ersichtlich, dass die Vorteile der arithmetischen (Abb. 3.6(ii)) und geometrischen Verknüpfung (Abb. 3.6(iii)) in der kürzeren Berechnungszeit liegen. Auch die Median-Berechnung ist soweit akzeptabel (Abb. 3.6(iv)). Die qualitativ beste Lösung bietet allerdings – trotz hohen Rechenaufwands – das *Fuzzy C-Means (FCM)*-Verfahren (Abb. 3.6(i)).

Die daraufhin erstellte Tabelle zur Auswertung der vier Verfahren wurde manuell ausgewertet (s. Tab. 3.1 und Tab. 3.2). Die Segmentierungsverfahren wurden auf Effizienz, Robustheit, Detailliertheit und Zuverlässigkeit untersucht:

Effizienz (*effizient*) steht hierbei für den Berechnungsaufwand pro Berechnungsdauer Tab. 3.3.

Robustheit (*robust*) steht für die Fähigkeit, Veränderungen, ohne Abänderungen von Parametern, standzuhalten.

Detailliertheit (*detailliert*) ist zu assoziieren mit der Genauigkeit, auf die der entsprechende Algorithmus auf das vorliegende Bild einwirkt.

Zuverlässigkeit (*zuverlässig*) wiederum bezieht sich nicht auf die hier existierende Zuverlässigkeit der Software, sondern eher auf die Qualität der Ergebnisse. Je höher die Zuverlässigkeit, um so höher auch die Trefferquote (s. Tab. 3.3).

Für die Bewertungen (Tab. 3.1 und Tab. 3.2) gelten hierbei die Unterteilungen *sehr schlecht* (–), *schlecht* (–), *befriedigend* (o), *gut* (+) und *sehr gut* (++)

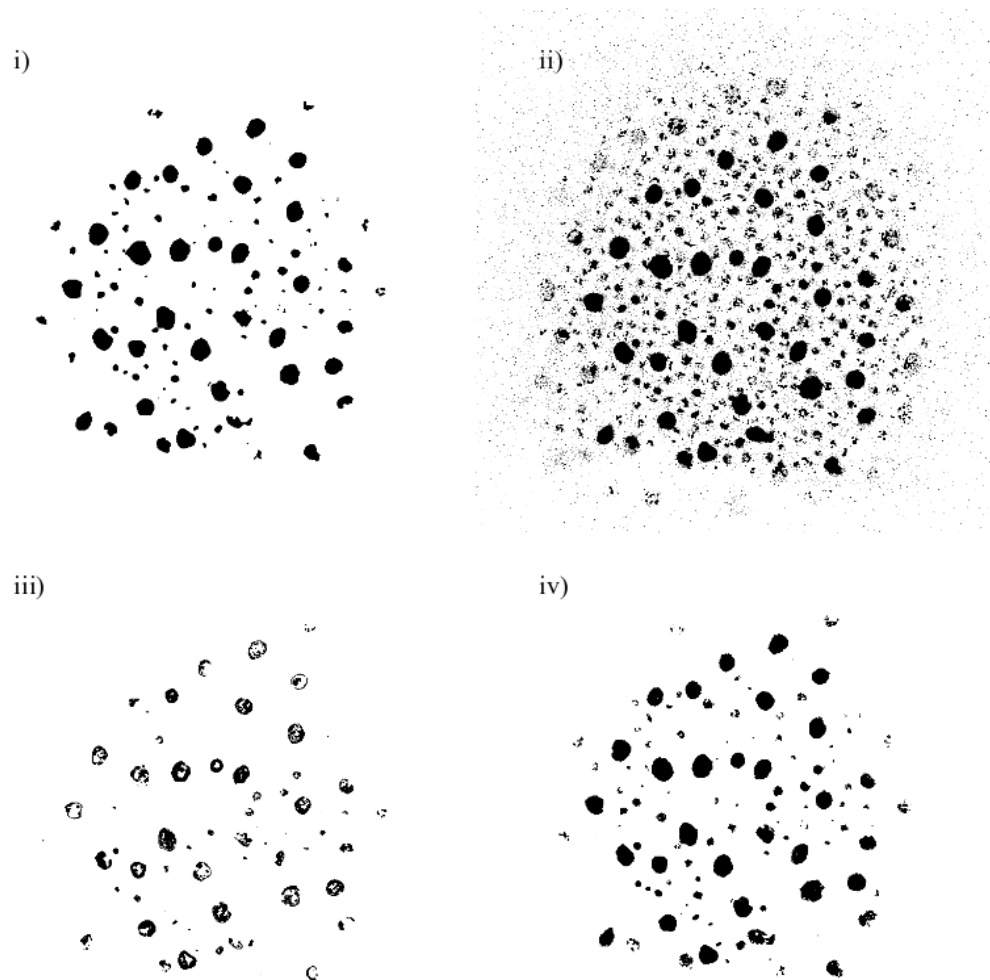


Abb. 3.6.: 2D-Abbildung aus verschiedenen Verknüpfungen von Segmentierungsverfahren: i) *Fuzzy C-Means (FCM)*, ii) Arithmetischer Mittelwert, iii) Geometrischer Mittelwert und iv) Median

Segmentierungsverfahren	<i>effizient</i>	<i>robust</i>	<i>detailliert</i>	<i>zuverlässig</i>
<i>Otsu-Schwelwertfilter (geschichtet)</i>	++	0	--	-
<i>Laplacian Level-Set</i>	--	+	++	+
<i>Confidence-Connected Regionenwachstum</i>	-	-	+	0
<i>Wasserscheidentransformation</i>	0	--	+	--

Tab. 3.1.: Auswertung verschiedener Segmentierungsverfahren

Verknüpfung der bisherigen Ergebnisse	<i>effizient</i>	<i>robust</i>	<i>detailliert</i>	<i>zuverlässig</i>
<i>Fuzzy C-Means (FCM)</i>	-	++	++	+
Arithmetisches Mittel	++	0	-	-
Geometrisches Mittel	+	-	0	0
Median	0	+	+	+

Tab. 3.2.: Auswertung verschiedener Verknüpfungen

In der vorliegenden Dissertation konnte festgestellt werden, dass bei der jeweiligen Ermittlung der Segmente vier mögliche Fälle auftreten können:

- Richtig positiv (r_p): Ein Segment wurde an einer bestimmten Stelle manuell festgelegt und automatisch wurde dort ein Segment erkannt.
- Falsch negativ (f_n): Ein Segment wurde an einer bestimmten Stelle manuell festgelegt, aber es wurde fälschlicherweise nicht automatisch als Segment erkannt..
- Falsch positiv (f_p): Ein Segment wurde an einer bestimmten Stelle manuell nicht festgelegt, aber es wurde dort fälschlicherweise automatisch ein Segment erkannt.
- Richtig negativ (r_n): Ein Segment wurde an einer bestimmten Stelle nicht manuell festgelegt und automatisch wurde dort auch kein Segment erkannt.

Die Segmentierungsverfahren wurden manuell verglichen und kontrolliert. Sobald genau ein manuell festgelegter Segmentmittelpunkt sich innerhalb genau eines automatisch bestimmten Segmentes befindet, so gilt er als existent und erkannt (richtig positiv). Liegt ein manuell festgelegter Segmentmittelpunkt außerhalb der automatisch bestimmten Segmente, so gilt er als existent und nicht erkannt (falsch negativ). Automatisch ermittelte Segmente, in denen sich allerdings kein manuell festgelegter Segmentmittelpunkt befindet, gelten als nicht existent, aber als irrtümllich zusammengefasstes Segment (falsch positiv). Die Größen und Positionen der Segmente werden in der vorliegenden Dissertationsschrift nicht weiter miteinbezogen.

Zur detaillierten Auswertung der einzelnen Segmentierungsverfahren wurden die Anzahl der in dem Experiment ermittelten Segmente, die dazu benötigte Zeit, die Genauigkeit und die Trefferquote verglichen. D. h. es wurden die benötigte Berechnungszeit, die Genauigkeit (s. Def. 7) und die Trefferquote (s. Def. 8) verglichen. Zunächst wurden die durchgeführten Segmentierungsabläufe (Abs. 3.2) gegenübergestellt (Abb. 3.7, Abb. 3.9, Abb. 3.8, Abb. 3.10, Abb. 3.11).

Definition 7 Die Genauigkeit³ (s. Gl. 3.1) steht für den Anteil der korrekt als positiv erkannten Ergebnisse an der Gesamtheit der als positiv erkannten Ergebnisse:

$$\frac{r_p}{r_p + f_p}. \quad (3.1)$$

Definition 8 Die Trefferquote⁴ (s. Gl. 3.2) steht hingegen für den Anteil der korrekt als Segment ermittelten Objekte an der Gesamtheit der tatsächlich vorliegenden, manuell festgelegten, also relevanten Segmenten:

$$\frac{r_p}{r_p + f_n}. \quad (3.2)$$

In Tab. 3.3 werden die Rechenzeit, die Genauigkeit (s. Def. 7) und die Trefferquote (s. Def. 8) gegenübergestellt. Hierfür wird die Anzahl der richtig positiv (r_p)⁵ und die Anzahl der falsch negativ (f_n)⁶ bestimmten Segmente dargestellt. Als optimaler Vergleich wird hier die manuelle Auswertung gewählt, so dass die falsch negative Segmentierung selbstverständlich bei 0 liegt.

³auch Positiver Vorhersagewert, Relevanz oder Wirksamkeit genannt

⁴auch Richtig-Positiv-Rate, Empfindlichkeit oder Sensitivität genannt

⁵ r_p – richtig positiv

⁶ f_p – falsch positiv

	r_p	f_n	f_p	Rechenzeit	Genauigkeit $\left(\frac{r_p}{r_p+f_p}\right)$	Trefferquote $\left(\frac{r_p}{r_p+f_n}\right)$
Manuell	109	0	0	—	100,0%	100,0%
<i>Otsu-Schwelwertfilter (geschichtet)</i>	88	21	636	1019,8s	12,5%	80,7%
Laplacian Level-Set	107	2	13	29216,6s	90,7%	98,2%
<i>Confidence-Connected Regionenwachstum</i>	57	52	36	1214,8s	78,1%	52,3%
<i>Wasserscheidentransformation</i>	66	43	119	1149,7s	46,5%	60,6%
Verknüpfung der bisherigen Ergebnisse				$\Sigma =$ 32600,9s		
Fuzzy C-Means (FCM)	109	0	11	53,6s	90,8%	100,0%
Arithmetisches Mittel (geglättet)	109	0	780	0,2s	12,3%	100,0%
Geometrisches Mittel	62	37	3	0,1s	56,5%	56,9%
Median	106	3	31	0,3s	79,1%	97,3%

Tab. 3.3.: Anzahl der ermittelten Segmente, Größe der Rechenzeit, Genauigkeit und Trefferquote verschiedener Verfahren und der sich daraus ergebenden Verknüpfungen

Wie in Tab. 3.3, Abb. 3.7 und Abb. 3.8 ersichtlich, ist das *Laplacian Level-Set*-Verfahren mit einer Genauigkeit von 90,7% und einer Trefferquote von 98,2% das qualitativ eindeutig beste Verfahren, benötigt aber auch mit 29216,6s die meiste Zeit. Der Grund für die lange Rechenzeit liegt in der Komplexität des Level-Set-Verfahrens und der Tatsache, dass das Level-Set-Verfahren wiederum für alle Saatpunkte, die möglich sind, angewendet wird.

Der *Otsu-Schwelwertfilter* hingegen hat die kürzeste Rechendauer von ca.1000s für einen z-Stapel, hat jedoch eine ziemlich niedrige Genauigkeit von 12,5%, bei einer Trefferquote von 80,7% (s. Tab. 3.3, Abb. 3.7 und Abb. 3.8).

Eine höhere Genauigkeit von 78,1% bietet das *Confidence-Connected Regionenwachstum*, wobei die Trefferquote bei lediglich 52,3% liegt und die Rechenzeit knapp 20% mehr beträgt (s. Tab. 3.3, Abb. 3.7 und Abb. 3.8).

Die *Wasserscheidentransformation* benötigt ebenso eine längere Rechendauer – allerdings nur 15% mehr. Die Trefferquote liegt zwar bei 60,5%, aber die Genauigkeit ist mit 46,5% deutlich niedriger (s. Tab. 3.3, Abb. 3.7 und Abb. 3.8).

Die geschilderten Feststellungen lassen sich in den Diagrammen Abb. 3.7, Abb. 3.8, Abb. 3.9, Abb. 3.10 und Abb. 3.11 zusammenfassen.

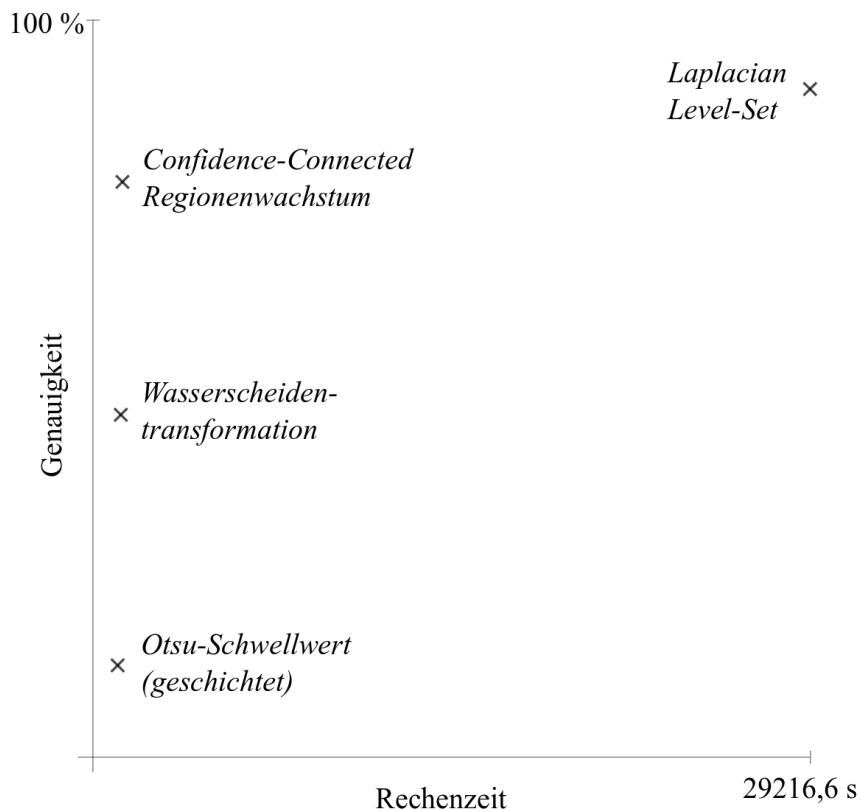


Abb. 3.7.: Genauigkeit-Rechenzeit-Diagramm 1

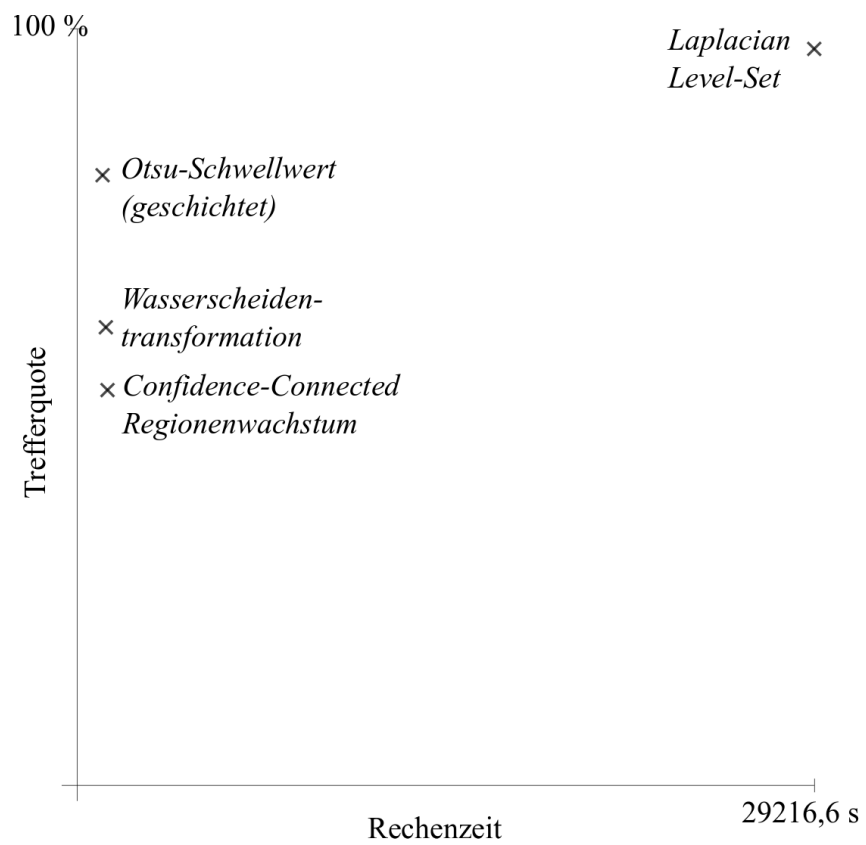


Abb. 3.8.: Trefferquote-Rechenzeit-Diagramm 1

In einem nächsten Schritt wurden die verschiedenen Verknüpfungen getestet (s. Tab. 3.3, Abb. 3.9 und Abb. 3.10). Weil bei der Verknüpfung per *Arithmetischem Mittel* eindeutig ein zu verrauschtes Abbild entstand (Abb. 3.6), wurde zusätzlich ein Glättungsfilter (Median) angewandt und das Ergebnis ausgewertet (Tab. 3.3). Trotz allem blieben beim *Arithmetischen Mittel (geglättet)* 889 Segmente erhalten, was zwar zu einer Trefferquote von 100,0%, aber auch zu einer Genauigkeit von lediglich 12,3% führte. Die Rechenzeit lag lediglich bei 0,2s - wobei der Glättungsfilter nicht mitgezählt wurde.

Das *Geometrische Mittel*, das – wie auch die anderen Verknüpfungsverfahren – zusätzlich zu den Basis-Operatoren anfällt, hatte mit nur 0,1s die kürzeste Rechenzeit. Die Genauigkeit war mit 56,6% zwar deutlich höher, als die des *Arithmetischen Mittels (geglättet)*, war aber mit einer Trefferquote von nur 56,9% trotzdem nicht ausreichend.

Auch der *Median* hat mit 0,3s eine kurze Rechenzeit. Die Trefferquote von immerhin 97,3% ist auch relativ gut. Sogar die Genauigkeit ist mit 79,1% akzeptabel, könnte aber noch optimiert werden.

Das *Fuzzy C-Means (FCM)*-Verfahren hebt sich qualitativ von den anderen zuvor genutzten Segmentierungsverfahren ab. Es ist vom Ergebnis her sogar etwas besser als das *Laplacian Level-Set*-Verfahren (s. Tab. 3.3, Abb. 3.8 und Abb. 3.9). Die Rechenzeit von ca. 54s ist zwar nicht mit der des *Arithmetischen Mittels (geglättet)*, des *Geometrischen Mittels* oder des *Medians* vergleichbar, aber doch deutlich niedriger, als die der vorherigen Segmentierungsverfahren. Die Genauigkeit liegt allerdings bei 90,8%, während die Trefferquote bei ganzen 100,0% liegt.

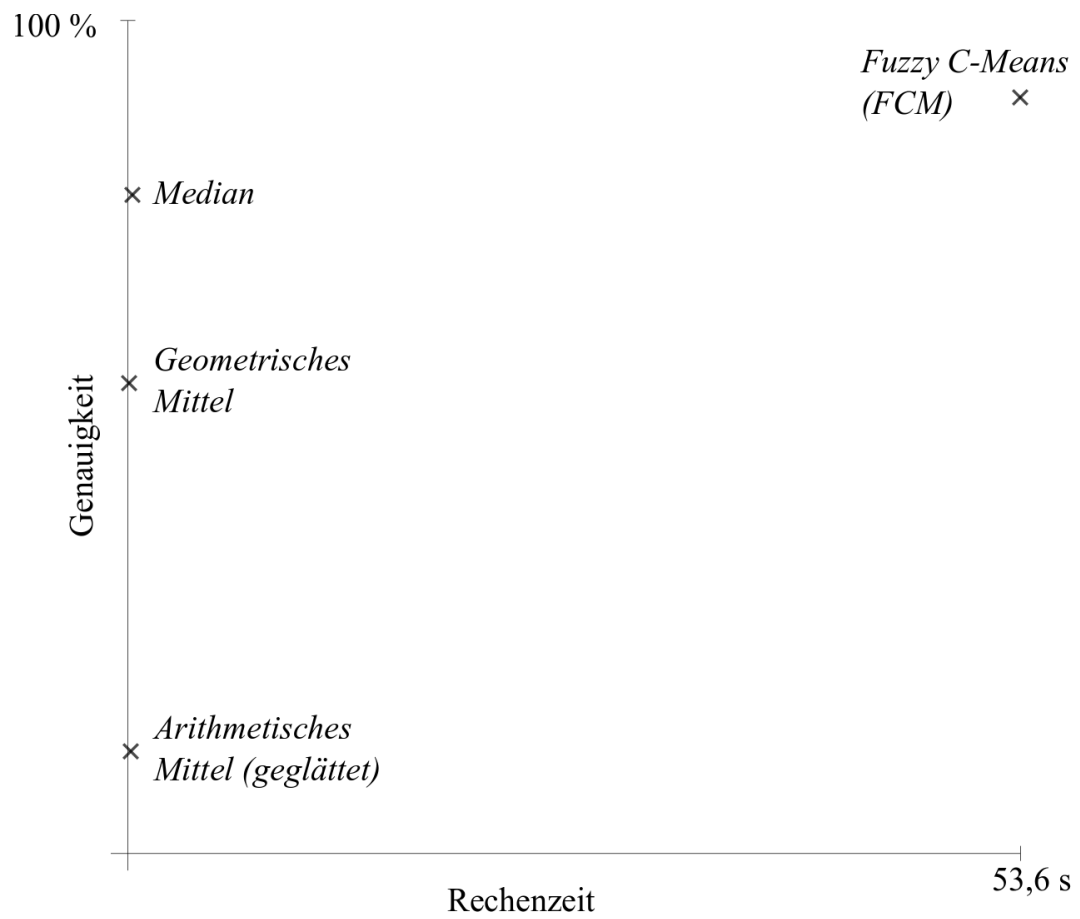


Abb. 3.9.: Genauigkeit-Rechenzeit-Diagramm 2

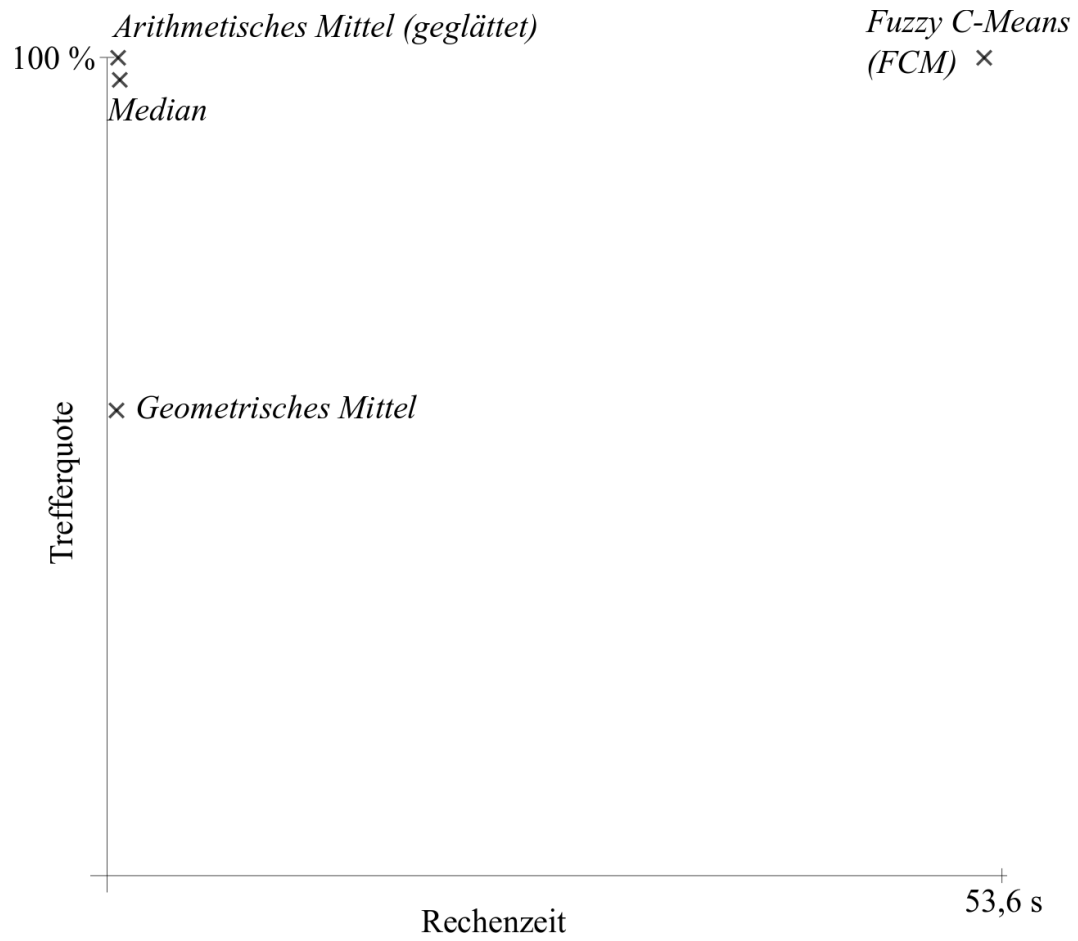


Abb. 3.10.: Trefferquote-Rechenzeit-Diagramm 2

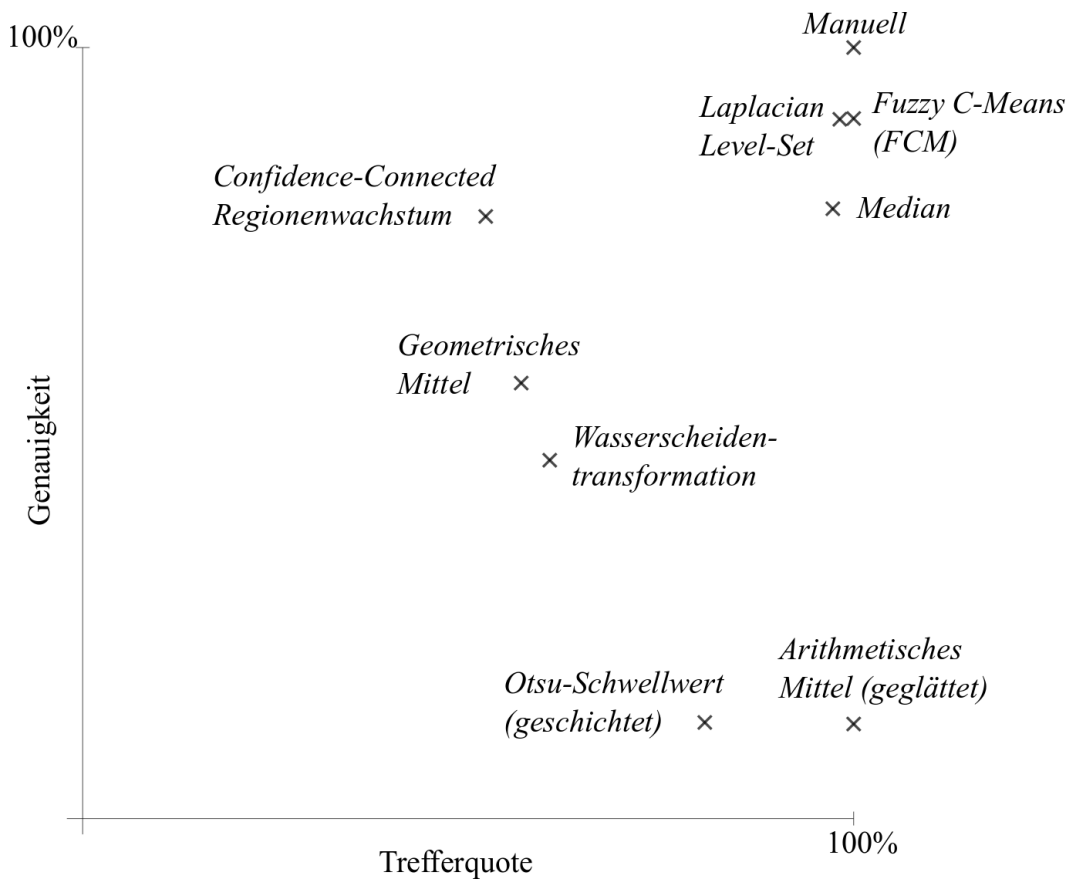


Abb. 3.11.: Genauigkeit-Trefferquote-Diagramm

Trotz allem ist der effektive Unterschied zwischen dem *Laplacian Level-Set* und dem anschließend verglichenen *Fuzzy C-Means (FCM)* statistisch nicht signifikant. Die Relevanz der leichten Verbesserung (bei Berechnungsdauer, Trefferquote und Genauigkeit) muss in späteren Studien mit größeren Datensätzen noch umfassender evaluiert werden.

Bei den getesteten Verfahren wurden die einzelnen Vorzüge genutzt und eine Verknüpfung mittels *Fuzzy C-Means (FCM)* vorgenommen, um eine qualitative Steigerung zu erzielen. Wird hingegen z. B. nur das *Laplacian Level-Set*-Verfahren durchgeführt und besteht lediglich ein begrenztes Zeitlimit, muss abgewogen werden, ob die Anwendung weiterer Verfahren und die im Anschluss durchgeführte qualitativ gute *Fuzzy C-Means (FCM)*-Verknüpfung lohnenswert und effektiv ist.

Eine qualitative Steigerung (bei Berechnungsdauer, Trefferquote und Genauigkeit) von wenigen Prozenten (nahe dem Optimum) ist sehr zeitintensiv. Hierbei muss stets problemspezifisch bedacht werden, was mehr gewichtet sein soll: die Qualität, die Quantität oder der Zeitaufwand (Abb. 1.1).

4. Neuartige Merkmalsgewinnung multidimensionaler Segmente

Wie in Abs. 1.2.1 und in Abs. 1.2.4 beschrieben, handelt es sich bei biologischen Datengewinnungen verfahrensbedingt um schwer zu analysierende Datensätze. Die dazugehörige mikroskopisch kleinen, biologischen Objekte sind z. B. Zellen, Synapsen oder DNA-Strukturen. Die Gestalt ist allerdings sehr uneinheitlich und auch versuchsabhängig. Kleine toxikologische Einwirkungen und die Beschaffenheiten der jeweiligen Experimente wirken auf die Segmente ein.

In den im Abs. 4 gegebenen Datensätzen handelt es sich um mikroskopisch gewonnene Volumendaten von Synapsen. In Abb. 4.1 sind unterschiedliche Synapsen abgebildet. Die Synapsen befanden sich in Diaphragmen (Lachmuskeln) von Mäusen [116]. Eine relativ homogene Struktur ist im rechten Abbild Abb. 4.1(ii) zu erkennen. Hier sind keine Löcher oder heterogene Strukturen zu erkennen. Bei dem Abbild handelt es sich um eine Synapse einer Maus, die vom Griscelli-Syndrom befallen ist. Hingegen eine deutlich inhomogene Struktur ist im linken Abbild Abb. 4.1(i) zu erkennen. Die dort dargestellte Synapse ist nicht vom Griscelli-Syndrom befallen und relativ inhomogen.

Wie in Abb. 4.1 zu erkennen, liegt die Besonderheit darin, dass bei solchen biologischen Objekten oft die Form, die Morphologie eine wichtige Rolle spielt [66, 85, 143, 150, 49, 54, 17, 41, 139].

Die für das Kapitel gewonnenen und weiter analysierten Datensätze sind 3D-Aufnahmen zu einem festen Zeitzustand. Allerdings wird bei den hier in dem Kapitel erstellten Verfahren zur Merkmalsextraktion die Multidimensionalität bewahrt.

Bewegungen im dreidimensionalen Raum entsprechen hier z. B. einem vierdimensionalen Segment. Die Bewegungsanalyse hat somit die Aufgabe, die Form bzw. die morphologischen Eigenschaften eines derartigen multidimensionalen Segmentes auszuwerten. Aber auch Zustandsaufnahmen – wie bei den Auswertungen von toxikologischen Einwirkungen (Abb. 4.1) – müssen anhand entsprechender Merkmale ausgewertet werden.

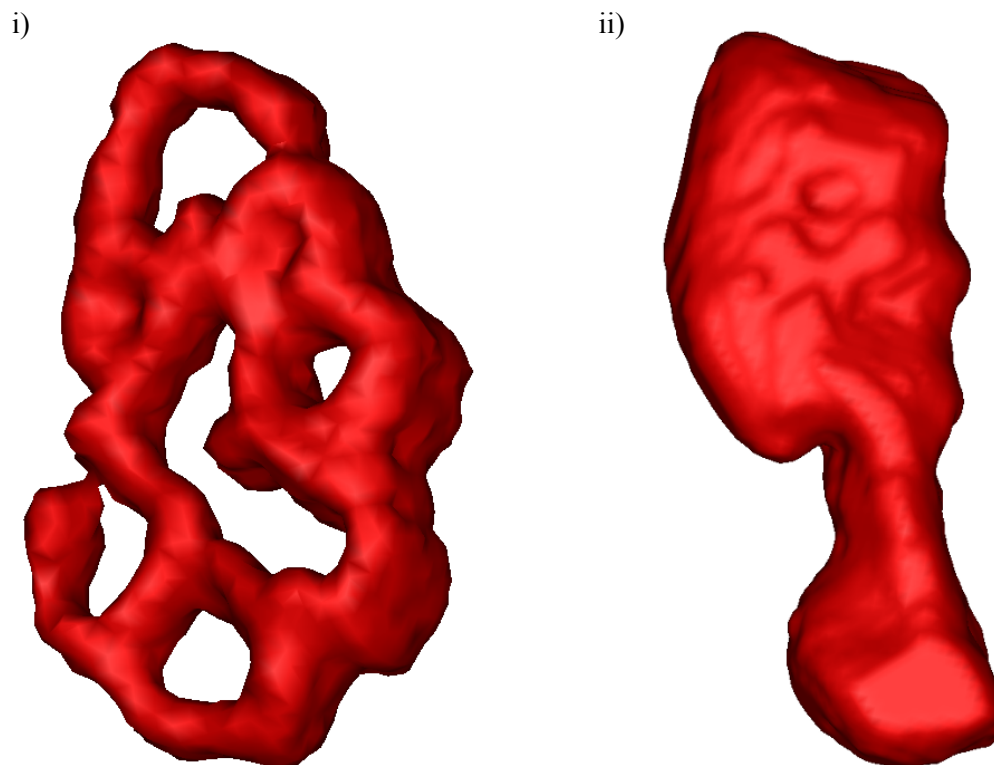


Abb. 4.1.: Dreidimensionale Abbildungen von Synapsen: i) vom Griscelli-Syndrom nicht-befallen ii) vom Griscelli-Syndrom befallen

Toxikologische Einwirkungen sind – aus Sicht des Biologen – anhand von speziellen Formen, Entwicklungen und morphologischen Eigenschaften zu erkennen. In der Arbeit wird folglich versucht, neue Ideen zur automatischen Bestimmung morphologischer Eigenschaften zu verwirklichen. Hierzu werden zwei Ansätze vertieft. Zuerst wird die Bestimmung und Auswertung von Netzstrukturen erarbeitet. Und als Zweites wird die Isoperimetrie angewandt, um das Verhältnis zur optimal homogenen Struktur zu betrachten.

Bestimmung und Auswertung von Netzstrukturen: Eine neuartige Methode ist es, Segmente mit dazugehörigen Graphen (s. Abs. 4.1.1 und Abs. 4.1.2) in Verbindung zu bringen und – mittels der Graphentheorie, der geometrischen netzbasierten Modellierung und topologischen Gegebenheiten – auszuwerten (s. Abs. 4.1.3).

- Daraus ergibt sich die Gelegenheit, Segmente – abhängig von Zyklen, Winkeln und ortsspezifischen Weiten – zu extrahieren (s. Abs. 4.1). Ein inhomogenes Objekt, wie in Abb. 4.1(i) zu erkennen, hat eben mehr Zyklen, als ein homogenes. Zu beachten ist, dass nicht immer Segmente fehlerfrei ermittelt werden können. Bei der Bestimmung von soeben erwähnten For-

men (z. B. Zyklen) soll weiterhin mit einer neuartigen Herangehensweise ein Prinzip geboten werden, mit dem bestimmte Formen geschätzt und abermals korrigiert werden.

- Weiterhin wird das Verhältnis von euklidischen Distanzen, zu geodätischen Distanzen hinterfragt und getestet (s. Abs. 4.1.4). Schließlich können geodätische Distanzen nicht größer sein als euklidische Distanzen. Und deren Unterschied ist umso größer, je inhomogener die entsprechenden Netzstrukturen sind.

Isoperimetrische Ungleichung: Morphologische Eigenschaften entsprechen z. B. dem Verhältnis zwischen Umfang und Fläche. Im Gegensatz hierzu wird in der Dissertation auf multidimensionale Objekte Bezug genommen und die entsprechende isoperimetrische Ungleichung [136, 37] betrachtet (s. Abs. 4.2).

4.1. Graphenbasierte morphologische Merkmalsextraktion

Von gewonnenen multidimensionalen Segmenten sollen praxisorientiert morphologische Eigenschaften ermittelt werden. Im Gegensatz zu z. B. menschlicher Objekterkennung [42, 115] können allerdings bei vielen biologischen Objekten nur schwer Formen vordefiniert und festgelegt werden. Biologische Objekte sind im Allgemeinen zu unterschiedlich. Die Auswertung von Skeletten entsprechender Segmente ist eine umsetzbare Alternative.

Abb. 4.2 stellt von Schritt (i) bis (iv) mögliche Zwischenzustände dar. In Abb. 4.2(i) ist ein Originalbild einer Mikroskopaufnahme von einer Synapse zu erkennen. Die Synapse ist relativ heterogen, deren Aufnahme ist verrauscht und von der Intensität her uneinheitlich.

Ein einfaches Segmentierungsverfahren, mit Hilfe eines Otsu-Schwellwertfilters, hinterlässt ein Binärbild, in dem die grobe Struktur der aufgenommenen Synapse noch vorgegeben ist (s. Abb. 4.2(ii)). Daraufhin wird ein in Abs. 4.1.1 beschriebenes Verfahren angewandt, um ein Skelett zu ermitteln (s. Abb. 4.2(iii)). Das Skelett wird anschließend einem Graphen zugeordnet (Abb. 4.2(iv)), der dann für weitere Berechnungen genutzt werden kann.

Experimentell wird auf einen Datensatz mit Abbildungen von 50 vom Griscelli-Syndrom befallenen¹ und 50 nicht-befallenen² Synapsen zugegriffen. Neuartige Verfahren werden in Abs. 4.1.3, Abs. 4.1.4 und Abs. 4.2 ausgewertet.

¹Vom Griscelli-Syndrom befallen, wird weiterhin als "dilute" bezeichnet

²Vom Griscelli-Syndrom nicht-befallen, wird weiterhin als "nicht-dilute" bezeichnet.

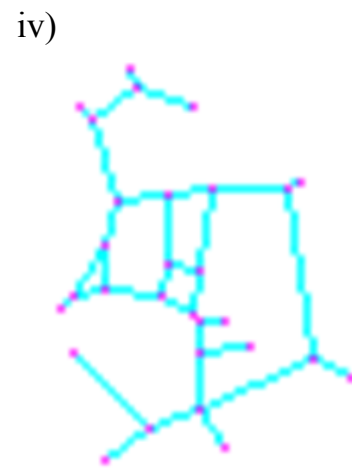
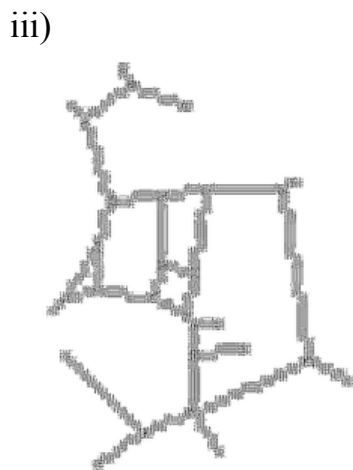


Abb. 4.2.: Phasen zur Erstellung von Graphen i) Original ii) Segment iii) Skelett iv) Graph

4.1.1. Berechnung eines Skeletts aus einem Segment

Eine Möglichkeit zur Erstellung eines Skeletts ist, von einem gegebenen Segment zunächst die Mediale Achse (s. Abb. 4.2(iii)) zu ermitteln.

Definition 9 *Eine Mediale Achse d -ter-Dimension entspricht der geometrischen Mitte eines Segmentes bzw. einer Menge von Punkten. Sie wurde festgelegt als die Menge von Punkten, die als Mittelpunkte von d -dimensionalen Kugeln³ mit maximalem Radius betrachtet werden kann - Kugeln, die sich innerhalb des Segmentes befinden und mindestens d Punkte am Rand, an der Oberfläche des Segmentes berühren [13].*

Mit einer Reihe von Skelettierungs-Algorithmen können Mediale Achsen bestimmt werden [43]. Kritisch wird es allerdings in höheren Dimensionen. Zum Beispiel können sich in einem Quader bei der Ermittlung *Medialer Achsen* ganze Flächen ergeben. Hierzu werden jedoch schon gegebene Ansätze genutzt [83, 89], um derartige Fälle zu umgehen. Eine so gewonnene Mediale Achse kann weiterbearbeitet und als Skelett des anfänglichen Segmentes betrachtet werden [65].

4.1.2. Berechnung eines Graphen aus einem Skelett

Um ein Skelett eines Objektes auswerten zu können, soll es zuerst in ein höheres Abstraktionsniveau, also einen Graphen transformiert werden [137]. Ein derartiges Skelett enthält somit Informationen der generellen morphologischen Eigenschaft, die für die Merkmalsextraktion nutzbar ist. Daraus folgend wird aus einem Skelett der entsprechend dazugehörige Graph bestimmt. Ein Graph besteht aus einer Menge von Knoten und entsprechenden Verknüpfungen [128]. Ein Muxel des Skeletts, das am Ende eines Bogens steht, entspricht einem Knoten. Ein derartiger Knoten steht entweder am Ende eines Zweiges des Skeletts, oder es befindet sich auf einem Schnitt mehrerer Zweige.

Nachdem alle Knoten gefunden wurden, müssen sie verknüpft werden, um entsprechende Bögen zu erstellen. Ein Bogen befindet sich zwischen zwei Knoten, wenn es einen Weg von Muxeln gibt, der zum Skelett zwischen den zwei Knoten gehört, ohne dass der Weg andere Knoten enthält. Das sei klar zu unterscheiden von einer Kante, die der direkten kürzesten Verknüpfung zwischen zwei Knoten entspricht. In Abb. 4.2 wird die Umwandlung eines Skeletts in einen Graphen illustriert. Der folgende Schritt besteht darin, dem Graphen weitere Informationen zuzuordnen [26, 15]. Für die Bögen werden die Besonderheiten der euklidischen Längen und der Orientierungen der Bögen gewählt. Für die Knoten wiederum dienen direkt deren Koordinaten.

³Die d -dimensionalen Kugeln entsprechen Hyperkugeln mit entsprechenden Sphären und Volumen im d -dimensionalen Raum.

4.1.3. Inhomogenität im Verhältnis zur Anzahl von Zyklen

Definition 10 Ein Graph ist genau dann planar, wenn er in einer Ebene durch Punkte und Linien überkreuzungsfrei dargestellt werden kann [128].

Definition 11 Nach dem Eulerschen Polyedersatz [37]⁴ gilt für planare Graphen G ⁵
⁶ ⁷

$$\text{Anz}_N(G) + \text{Anz}_F(G) - \text{Anz}_E(G) = 2. \quad (4.1)$$

Nun kann mit ermittelten planaren ungerichteten Graphen (s. Abb. 4.3) die Anzahl der gegebenen Flächen bestimmt werden, wobei eine innere Fläche eines Graphen einem Zyklus entspricht [97]. Nicht-planare Graphen, die bei höherdimensionalen Räumen durchaus auftreten können, werden zunächst nicht weiter betrachtet. Je mehr Zyklen vorliegen, umso inhomogener ist der entsprechende Graph sowie das dazugehörige Segment.

Ein Beispiel liegt in Abb. 4.3 vor. In Abb. 4.3(i) ist ein Graph mit homogener Struktur zu erkennen, der keinen einzigen Zyklus besitzt. Als Gegensatz ist in Abb. 4.3(ii) ein Graph mit zumindest zwei Zyklen abgebildet. Die Struktur ist eher inhomogen.

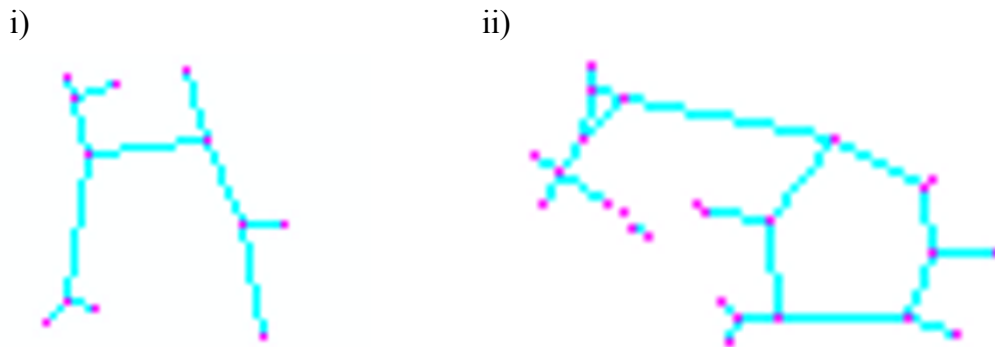


Abb. 4.3.: Planare Graphen i) homogen ii) inhomogen

⁴Bewiesen wurde der Satz induktiv.

⁵ $\text{Anz}_N(G)$ - Anzahl der Knoten / Nodes von G

⁶ $\text{Anz}_F(G)$ - Anzahl der Flächen / Faces von G

⁷ $\text{Anz}_E(G)$ - Anzahl der Kanten / Edges von G

Experimenteller Vergleich mit der Anzahl von Zyklen

Im Weiteren werden planare ungerichtete Graphen und direkte Zyklen (nicht Fuzzy-Zyklen) betrachtet. Von gegebenen Synapsen (50 dilute und 50 nicht-dilute) wurden Graphen erstellt und daraufhin die Anzahl der Zyklen berechnet.

Wie in Tab. 4.1 und Abb. 4.4 ersichtlich, liegt die mittlere Anzahl von Zyklen der 50 nicht-diluten Synapsen bei knapp 17 Stück. Die dazugehörige Standardabweichung liegt allerdings bei 10,76. Hingegen enthalten die 50 diluten Synapsen im Durchschnitt knapp 3 Zyklen. Und die Standardabweichung liegt bei 2,87. Trotz der relativ großen Standardabweichung der Zyklenanzahl der nicht-diluten Synapsen kann gesagt werden, dass in dem durchgeführten Experiment die kleinste Anzahl von Zyklen der nicht-diluten Synapsen (Minimalwert = 3) größer ist, als der Mittelwert der diluten Synapsen (Mittelwert = 2,75). Zyklen können hier durchaus als ein Merkmal, als eine Eigenschaft – zur Unterscheidung von homogen und nicht-homogen geformten Synapsen – genutzt werden.

Ein Zyklus entspricht einem direkt geschlossenen Verlauf innerhalb eines Graphen. Die Koordinaten oder die Dimension entsprechender Knoten sind in dem einfachen Fall nicht weiter relevant. Lediglich die Verknüpfungen⁸ haben hierfür eine Bedeutung. Für den Versuch wurden die zu Beginn erwähnten 50 diluten und 50 nicht-diluten Synapsen ausgewertet.

	Zyklen bei	
	nicht-diluten Synapsen	diluten Synapsen
Minimalwert	3	1
Maximalwert	47	11
Mittelwert	16,70	2,75
Standardabweichung	10,76	2,87

Tab. 4.1.: Experiment zur Anzahl der Zyklen als Merkmal von gegebenen Synapsen in Bezug zur Infektion mit dem Griscelli-Syndrom

⁸Verknüpfungen auch dargestellt als Inzidenz- bzw. Adjazenzmatrizen

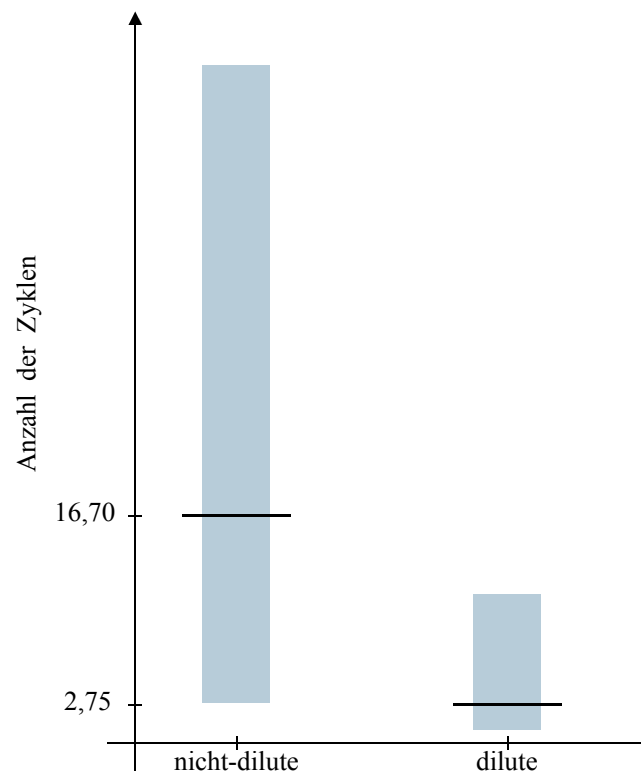


Abb. 4.4.: Diagramm zur Anzahl der Zyklen in gegebenen Synapsen

4.1.4. Gegensatz von euklidischen Distanzen zu geodätischen Wegen

Im Folgenden wird der Gegensatz von euklidischen Distanzen Def. 12 zu geodätischen Wegen Def. 13 hinterfragt, um ihn daraufhin für eine neuwertige Beschreibung morphologischer Eigenschaften zu nutzen. Denn, je homogener ein vorliegendes Netzwerk bzw. der entsprechende Graph ist, umso kleiner ist die durchschnittliche Differenz von geodätischer Weglänge zu euklidischer Distanz zwischen zwei unterschiedlichen Punkten.

Definition 12 *Euklidische Distanz steht für den Abstand zweier Punkte des gegebenen d -dimensionalen euklidischen Raumes \mathbb{R}^d [18]. Die Distanz ist für zwei Punkte X_i und X_j ⁹ durch die euklidische Norm $\|X_i - X_j\|_2$ des Differenzvektors zwischen den beiden Punkten festgelegt und wird weitergehend wie folgt genannt:*

$$l_{Eukl}(X_i, X_j). \quad (4.2)$$

Definition 13 *Eine Geodäte¹⁰ ist die lokal kürzeste Verbindungskurve zweier Punkte eines topologischen Raumes bzw. einer Mannigfaltigkeit im \mathbb{R}^d [18]. Im gekrümmten d -dimensionalen Raum wird die Bogenlänge zwischen Punkt X_i und X_j durch $s = \int_{X_i}^{X_j} ds$ ¹¹ beschrieben.*

Auf einer Netzwerkstruktur im multidimensionalen Raum (z. B. ungerichteter Graph) entspricht eine Geodäte also einer kürzesten Distanz zwischen Knoten X_i und X_j :

$$l_{Geod}(X_i, X_j). \quad (4.3)$$

⁹ $i, j \in \mathbb{N}, i \neq j$

¹⁰auch geodätischer Weg genannt

¹¹ $i, j \in \mathbb{N}, i \neq j$

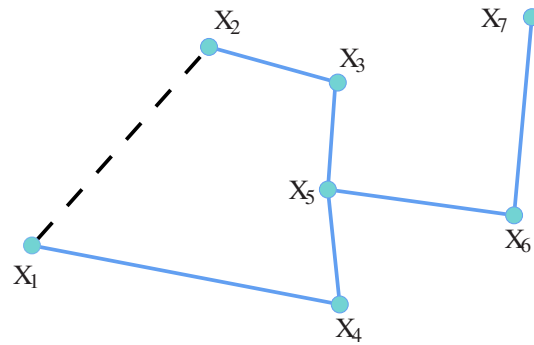


Abb. 4.5.: Euklidische Distanz zwischen X_1 und X_2 und Geodäte über X_1, X_4, X_5, X_3, X_2

In einer Netzwerkstruktur entspricht eine kürzeste Distanz einem kürzesten Pfad zwischen zwei Knoten. D. h., eine Geodäte zwischen Punkt X_1 und X_2 in einer Netzwerkstruktur entspricht der Summe aller euklidischen Distanzen der Zwischenwege. Im gegebenen Beispiel (s. Abb. 4.5) entspricht das dem Weg $[X_1, X_4, X_5, X_3, X_2]$ und somit der Länge $l_{Geod}(X_1, X_2) = l_{Eukl}(X_1, X_4) + l_{Eukl}(X_4, X_5) + l_{Eukl}(X_5, X_3) + l_{Eukl}(X_3, X_2)$.

Die Bestimmung kürzester Distanzen kann in folgende Gruppen unterteilt werden:

- der kürzeste Pfad zwischen zwei unterschiedlichen Knoten
- der kürzeste Pfad zwischen einem Startknoten und allen anderen Knoten des Graphen
- der kürzeste Pfad von jedem Knoten im Graphen zu jedem anderen Knoten

Zum Ermitteln der kürzesten Pfade stehen hier eine Reihe von Algorithmen zur Verfügung [128, 14]. Für das weitere Vorgehen wurde der Dijkstra-Algorithmus [128, 127] für jeden einzelnen Knoten gewählt. Bestimmt werden somit für alle gegebenen Knoten bzw. Punkte die kürzesten Pfade bzw. Geodäten zu jedem anderen Knoten.

Von Interesse ist, wie anfangs beschrieben, das Verhältnis aller euklidischen Distanzen zu allen Längen der Geodäten (Def. 14). Beim weiteren Vorgehen wird dafür die Funktion

$$\mathbf{v}(l_{Geod}(X_i, X_j), l_{Eukl}(X_i, X_j)) = l_{Geod}(X_i, X_j) - l_{Eukl}(X_i, X_j) \quad (4.4)$$

in der vorliegenden Dissertationsschrift festgelegt¹², wobei:

$$l_{Geod}(X_i, X_j) \geq l_{Eukl}(X_i, X_j). \quad (4.5)$$

Definition 14 $\Upsilon(\mathbf{v}, X)$ steht für das durchschnittliche Verhältnis aller euklidischen Distanzen zu allen geodätischen Längen eines gegebenen nicht-gerichteten Graphen mit vorgegebener Funktion $\mathbf{v}(l_{Geod}(X_i, X_j), l_{Eukl}(X_i, X_j))$. $\left(\frac{\sum_{\forall j \neq i} \mathbf{v}(l_{Geod}(X_i, X_j), l_{Eukl}(X_i, X_j))}{Anz_N(G) - 1} \right)$ ist hierbei das durchschnittliche Verhältnis $\mathbf{v}(l_{Geod}(X_i, X_j), l_{Eukl}(X_i, X_j))$ aller Pfade von einem Knoten X_i ausgehend. $\Upsilon(\mathbf{v}, X)$ lässt sich daraufhin wie folgt mittels arithmetischem Mittel beschreiben, wobei $i, j \in \mathbb{N}; 1 \leq i, j \leq Anz_N(G)$:

$$\Upsilon(\mathbf{v}, X) = \frac{\sum_{\forall i} \left(\frac{\sum_{\forall j \neq i} \mathbf{v}(l_{Geod}(X_i, X_j), l_{Eukl}(X_i, X_j))}{Anz_N(G) - 1} \right)}{Anz_N(G)} \quad (4.6)$$

¹² $\mathbf{v}(l_{Geod}(X_i, X_j), l_{Eukl}(X_i, X_j)) \in [0, \infty]$

Experimenteller Vergleich von Netzstrukturen

Wie in Tab. 4.2 und Abb. 4.6 ersichtlich, können in dem Experiment deutliche Differenzierungen vorgenommen werden. Für den Versuch wurden 50 dilute (homogene) und 50 nicht-dilute (inhomogene) Synapsen ausgewertet. Während bei diluten Synapsen der Mittelwert für $\Upsilon(v, X)$ bei 0,33, zwischen 0,01 und 0,67 liegt, beträgt der Mittelwert von nicht-diluten Synapsen 1,03, zwischen 0,37 und 3,66. Der minimale Wert der nicht diluten Synapsen (0,37) entspricht somit ungefähr dem Mittelwert der diluten Synapsen. Und der Maximalwert der diluten Synapsen (0,67) liegt deutlich unter dem Mittelwert der nicht-diluten Synapsen. Das Verfahren ist also nutzbar zur Unterscheidung homogener und inhomogener morphologischer Eigenschaften.

	$\Upsilon(v, X)$ bei	
	nicht-diluten Synapsen	diluten Synapsen
Minimalwert	0,37	0,01
Maximalwert	3,66	0,67
Mittelwert	1,03	0,33
Standardabweichung	0,67	0,18

Tab. 4.2.: Experiment zum Verhältnis euklidischer und geodätischer Distanzen als Merkmal von gegebenen Synapsen, in Bezug zum Befall mit dem Griscelli-Syndrom

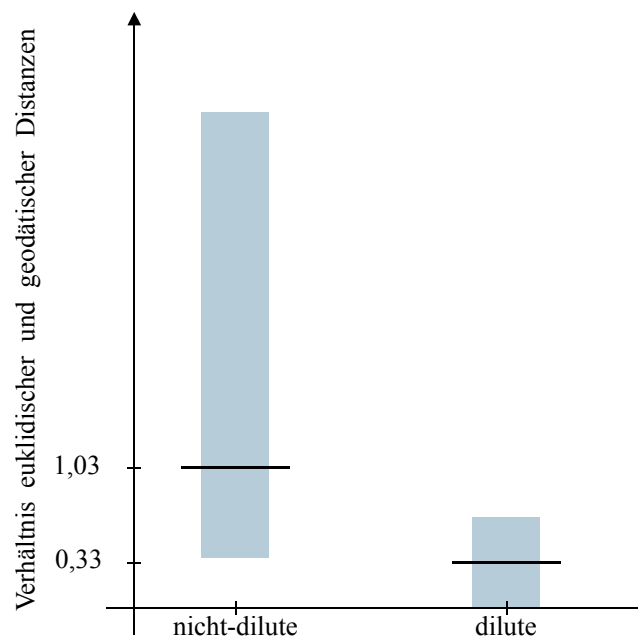


Abb. 4.6.: Diagramm zum Verhältnis euklidischer und geodätischer Distanzen gegebener Synapsen

4.2. Isoperimetrisch morphologische Merkmalsextraktion

Bei Untersuchungen von Einwirkungen chemischer sowie toxikologischer Wirkstoffe auf biologische Zellstrukturen werden oft Entwicklungen und Wachstum evaluiert. Das Griscelli-Syndrom z. B. lässt sich u. a. anhand der Form entsprechender Synapsen ermitteln (Abs. 1.2.4). Ebenso können Einwirkungen chemischer Stoffe mittels des Wachstums von Zellstrukturen analysiert werden (Abs. 1.2.4). Betrachtet wird hierbei die (In)Homogenität entsprechender Trajektorien mehrerer Zellen.

Auf Grund der Multidimensionalität bietet es sich an, die stetige Bestimmung der (In)Homogenität abstrakt zu betrachten. (In)Homogenität – und somit wichtige morphologische Eigenschaften – multidimensionaler Segmente kann mit dem Verhältnis zwischen Oberflächen und Volumen bestimmt werden. Um abstrakt das multidimensional zu betrachten, wird die Isoperimetrische Ungleichung genutzt [136]. Die Isoperimetrische Ungleichung¹³ ist eine mathematische Ungleichung, die für das Verhältnis von Rand¹⁴ zu Volumen¹⁵ eines multidimensionalen Körpers steht [136, 37].

Definition 15 (multidimensionale Kugel) Die Menge aller Punkte des d -dimensionalen euklidischen Raumes, deren Abstand zu einem gegebenen Punkt (dem Mittelpunkt) kleiner gleich einer positiven reellen Zahl r (dem Radius) ist, entspricht der Hyperkugel bzw. der d -dimensionalen Kugel K im \mathbb{R}^d [18].

Definition 16 (Gammafunktion) Bei der Gammafunktion handelt es sich um eine transzendente analytische Funktion $\Gamma(x) = \int_0^{\infty} t^{x-1} \cdot e^{-t} dt$, für $x \in \mathbb{C}$ [18]. Infolgedessen gilt für alle $n \in \mathbb{N}$: $\Gamma(n) = (n-1)!$

Definition 17 (Lebesgue-Maß) Ein d -dimensionales Lebesgue-Maß L_d ordnet jedem d -dimensionalen Segment S im \mathbb{R}^d sein d -dimensionales Volumen $L_d(S)$ zu [82].

Definition 18 (Hausdorff-Maß) Ein $(d-1)$ -dimensionales Hausdorff-Maß H_{d-1} , welches dem topologischen Rand ∂S vom Segment S ein Maß $H_{d-1}(\partial S)$ zuordnet, entspricht – bei einem $(d-1)$ -dimensionalen Rand – einem bestimmbareren $(d-1)$ -dimensionalen Oberflächeninhalt [58].

¹³Das Wort *isoperimetrisch* entstammt dem Griechischen: *iso* steht für "gleich", und *perimeter* bedeutet "Umfang".

¹⁴Der Rand steht für die Menge des gegebenen d -dimensionalen Raumes zwischen Abschluss und Innerem eines gegebenen d -dimensionalen Objektes.

¹⁵Das Volumen steht für die Menge aller Elemente innerhalb eines d -dimensionalen Objektes im d -dimensionalen Raum.

Das d -dimensionale Lebesgue-Maß bzw. das Volumen einer d -dimensionalen Kugel K mit Radius r beträgt [82]:

$$L_d(K) = r^d \cdot \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2} + 1)} \quad (4.7)$$

$L_d(K)$ erhält folgende Werte:

d	1	2	3	4	5	6	7	8	...	∞
$L_d(K)$	2	π	$\frac{4}{3}\pi$	$\frac{1}{2}\pi^2$	$\frac{8}{15}\pi^2$	$\frac{1}{6}\pi^3$	$\frac{16}{105}\pi^3$	$\frac{1}{24}\pi^4$...	0
	2,00	3,14	4,19	4,93	5,26	5,17	4,72	4,06	...	0

Tab. 4.3.: Verschiedene Werte vom Lebesgue-Maß einer Kugel mit Radius 1 [18]

Das $(d - 1)$ -dimensionale Hausdorff-Maß bzw. die Sphäre einer d -dimensionalen Kugel K mit Radius r beträgt [58]:

$$H_{d-1}(\partial K) = \frac{\partial L_d(K)}{\partial r} = d \cdot r^{d-1} \cdot \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2} + 1)} = 2 \cdot r^{d-1} \cdot \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2})} \quad (4.8)$$

$H_{d-1}(\partial K)$ erhält folgende Wert:

d	1	2	3	4	5	6	7	8	...	∞
$H_{d-1}(\partial K)$	2	2π	4π	$2\pi^2$	$\frac{8}{3}\pi^2$	π^3	$\frac{16}{15}\pi^3$	$\frac{1}{3}\pi^4$...	0
	2,00	6,28	12,57	19,74	26,32	31,01	33,07	32,47	...	0

Tab. 4.4.: Verschiedene Werte vom Hausdorff-Maß einer Kugel mit Radius 1 [18]

Für jedes nichtleere beschränkte Gebiet S im \mathbb{R}^d mit rektifizierbarem¹⁶ Rand und einer d -dimensionalen Kugel K ergibt sich aus Def. 17 und Def. 18:

$$\frac{L_d(S)^{\frac{1}{d}}}{H_{d-1}(\partial S)^{\frac{1}{d-1}}} \leq \frac{L_d(K)^{\frac{1}{d}}}{H_{d-1}(\partial K)^{\frac{1}{d-1}}} \quad (4.9)$$

$\frac{L_d(K)^{\frac{1}{d}}}{H_{d-1}(\partial K)^{\frac{1}{d-1}}}$ bei der d -dimensionalen Kugel K ist unabhängig vom Radius r und der Position im \mathbb{R}^d .

¹⁶rektifizierbar bedeutet, eine z. B. dreidimensionale Ebene so gerade ziehen zu können, dass die Oberflächengröße direkt an der geraden Fläche zu ermitteln ist.

Aus Gl. 4.7, Gl. 4.8 und Gl. 4.9 ergibt sich:

$$\frac{L_d(S)^{\frac{1}{d}}}{H_{d-1}(\partial S)^{\frac{1}{d-1}}} \leq \frac{\left(r^d \cdot \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2}+1)}\right)^{\frac{1}{d}}}{\left(d \cdot r^{d-1} \cdot \frac{\pi^{\frac{d}{2}}}{\Gamma(\frac{d}{2}+1)}\right)^{\frac{1}{d-1}}} = \frac{\frac{\pi^{\frac{1}{2}}}{\Gamma(\frac{d}{2}+1)^{\frac{1}{d}}}}{d^{\frac{1}{d-1}} \cdot \frac{\pi^{\frac{1}{2}} \cdot \pi^{\frac{1}{2} \cdot \frac{1}{d-1}}}{\Gamma(\frac{d}{2}+1)^{\frac{1}{d-1}}}} = \frac{\Gamma(\frac{d}{2}+1)^{\frac{1}{(d-1)d}}}{d^{\frac{1}{d-1}} \cdot \pi^{\frac{1}{2 \cdot (d-1)}}}. \quad (4.10)$$

Und für $2 \leq d \in \mathbb{N}$ ergeben sich somit folgende Ungleichungen:

$$\begin{aligned} d = 2: & \quad \frac{L_2(S)^{\frac{1}{2}}}{H_1(\partial S)^1} \leq \frac{\pi^{\frac{1}{2}}}{2\pi} \approx 0,282\dots \\ d = 3: & \quad \frac{L_3(S)^{\frac{1}{3}}}{H_2(\partial S)^{\frac{1}{2}}} \leq \frac{\left(\frac{4}{3}\pi\right)^{\frac{1}{3}}}{(4\pi)^{\frac{1}{2}}} \approx 0,454\dots \\ d = 4: & \quad \frac{L_4(S)^{\frac{1}{4}}}{H_3(\partial S)^{\frac{1}{3}}} \leq \frac{\left(\frac{\pi^2}{2}\right)^{\frac{1}{4}}}{(2 \cdot \pi^2)^{\frac{1}{3}}} \approx 0,551\dots \\ d = 5: & \quad \frac{L_5(S)^{\frac{1}{5}}}{H_4(\partial S)^{\frac{1}{4}}} \leq \frac{\left(\frac{8 \cdot \pi^2}{15}\right)^{\frac{1}{5}}}{\left(\frac{8 \cdot \pi^2}{3}\right)^{\frac{1}{4}}} \approx 0,615\dots \\ d = 6: & \quad \frac{L_6(S)^{\frac{1}{6}}}{H_5(\partial S)^{\frac{1}{5}}} \leq \frac{\left(\frac{\pi^3}{6}\right)^{\frac{1}{6}}}{(\pi^3)^{\frac{1}{5}}} \approx 0,661\dots \\ & \quad \vdots \\ d \rightarrow \infty: & \quad \lim_{d \rightarrow \infty} \left(\frac{L_d(S)^{\frac{1}{d}}}{H_{d-1}(\partial S)^{\frac{1}{d-1}}} \right) \leq 1 \end{aligned}$$

Bei der Merkmalsextraktion zweidimensionaler Segmente ist die Nutzung des Verhältnis zwischen Flächeninhalt und entsprechendem Flächenrand bekannt. Im dreidimensionalen Raum entspricht das wiederum dem Verhältnis zwischen Volumen und Oberfläche.

Die Isoperimetrische Ungleichung wurde bereits zum Teil zur Beschreibung von dreidimensionalen Objekten genutzt [90]. Jedoch kann nun erstmals in der vorliegenden Arbeit das Maß der Isoperimetrie abstrakt für multidimensionale Objekte festgelegt werden. Mit Hilfe des Maßes kann bestimmt werden, wie groß die Ähnlichkeit der Morphologie eines multidimensionalen Objektes zu der einer Kugel ist.

Das bedeutet, dass unter allen Körpern im multidimensionalen Raum mit gleicher Größe des Randes, die Sphäre¹⁷ das größte Volumen aufweist. Auch im d -dimensionalen euklidischen Raum \mathbb{R}^d gilt die analoge Aussage [136]:

Unter allen Körpern mit gleichem $(d-1)$ -dimensionalem Rand besitzt die d -dimensionale Sphäre das größte d -dimensionale Volumen.

¹⁷Eine Sphäre steht für die Verallgemeinerung des Kugelrandes auf beliebig vielen Dimensionen.

Aus Gl. 4.9 und Gl. 4.10 wird hiermit die Funktion ϖ festgelegt:

$$\varpi(S) = \frac{L_d(S)^{\frac{1}{d}}}{H_{d-1}(\partial S)^{\frac{1}{d-1}}} \cdot \frac{H_{d-1}(\partial K)^{\frac{1}{d-1}}}{L_d(K)^{\frac{1}{d}}} = \frac{L_d(S)^{\frac{1}{d}}}{H_{d-1}(\partial S)^{\frac{1}{d-1}}} \cdot \frac{\Gamma(\frac{d}{2} + 1)^{\frac{1}{(d-1)d}}}{d^{\frac{1}{d-1}} \cdot \pi^{\frac{1}{2(d-1)}}} \quad (4.11)$$

Des Weiteren besteht nun die Möglichkeit, Krümmungen, Streckungen, Bewegungen, Entwicklungen und andere morphologische Einwirkungen eines gesamten multidimensionalen Objektes mit Isoperimetrie zu verknüpfen und somit indirekt zu bestimmen. Z. B. eine Entwicklung im dreidimensionalen Raum entspricht einem vierdimensionalen Objekt. Eine gleichmäßige, homogene Entwicklung wird einer einheitlichen Sphäre viel mehr ähneln, als eine uneinheitliche, heterogene Entwicklung. Gerade bei der Bestimmung von biologischen Abläufen können derartige morphologische Eigenschaften zur Auswertung genutzt werden.

Dreidimensionaler Versuch zum Deuten der Isoperimetrie

Auf Grund der Diskretisierung akquirierter Daten handelt es sich bei den zu analysierenden Daten um diskrete Datensätze. Das Lebesgue-Maß (Volumen) kann daraufhin als Menge aller im entsprechenden Segment befindlichen Muxel betrachtet werden. Das Hausdorff-Maß wiederum entspricht der Menge der am Rand zum Segment liegenden Muxel.

In einem Experiment soll nun dreidimensional die Relevanz zur praktischen Anwendung getestet werden. Bei den Datensätzen handelt es sich um Synapsen von Diaphragmen von Mäusen. 50 der untersuchten Mäuse waren vom Griscelli-Syndrom befallen. Die anderen 50 Mäuse waren nicht befallen.

Die Inhomogenität – was durchaus auch als Nicht-Glattheit, mit bestimmtem Grad der Steigungen gedeutet werden kann – entspricht einer Korrelation zum Griscelli-Syndrom. Im Gegensatz zu einer befallenen Synapse, erhält eine gesunde Synapse einen möglichst großen Rand auf einem kleinen Volumen und eine dazugehörige Stabilität (d. h. beim Erlangen der Oberfläche darf sie nicht zu dünn werden).

Die Nicht-Glattheit kann nun mittels des Verhältnisses von Lebesgue-Maß und Hausdorff-Maß mit ϖ ermittelt werden, wird experimentell getestet und dokumentiert (Tab. 4.5).

Experimenteller isoperimetrischer Vergleich

Der gleiche Datensatz zur Untersuchung von Graphen (Abs. 4.1.3, Abs. 4.1.4) wird nun zur Auswertung der entsprechenden Isoperimetrie (Gl. 4.11) genutzt. Weitergehend wird praktisch bedingt der Kehrwert der isoperimetrischen Ungleichung (s. Gl. 4.9) betrachtet. Wie in Tab. 4.5 und in Abb. 4.7 ersichtlich ist, haben vom Griscelli-Syndrom befallene, also homogen geformte Synapsen eine deutlich kleinere Isoperimetrie bzw. kleinere Homogenität als nicht-befallene, inhomogene Synapsen (Abb. 4.1). Auch hier ist der Mittelwert der diluten-Synapsen (8,74) nicht viel größer als der Minimalwert nicht-diluter Synapsen (8,59). Und der Maximalwert der diluten-Synapsen (17,56) ist kleiner, als der Mittelwert der nicht-diluten Synapsen (23,37). Das Merkmal der Isoperimetrie kann in dem biologischen Experiment somit multidimensional gut genutzt werden, um in diskursiver Deutlichkeit Ergebnisse zu interpretieren. Für den Versuch wurden die zu Beginn erwähnten 50 diluten und 50 nicht-diluten Synapsen ausgewertet.

	ϖ bei	
	nicht-diluten Synapsen	diluten Synapsen
Minimalwert	8,59	1,67
Maximalwert	45,81	17,56
Mittelwert	23,37	8,74
Standardabweichung	7,57	3,72

Tab. 4.5.: Experiment zur Isoperimetrie als Merkmal von gegebenen Synapsen, in Bezug zur Infektion mit dem Griscelli-Syndrom

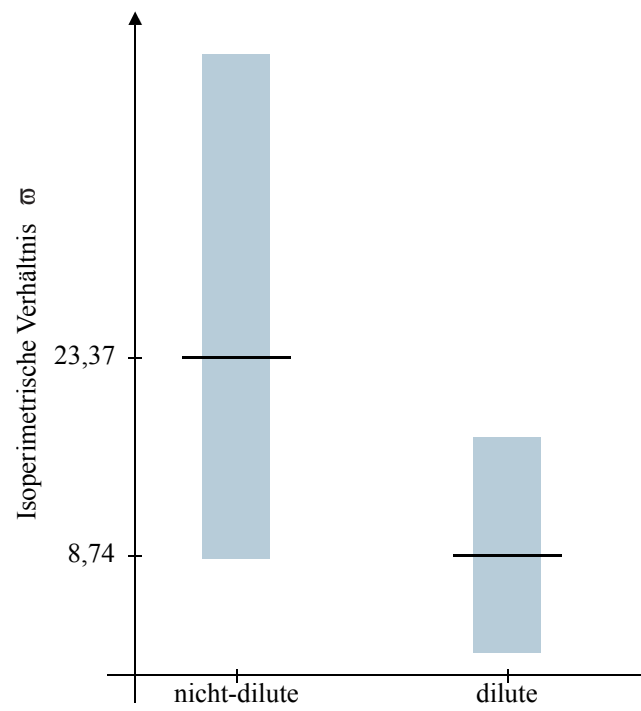


Abb. 4.7.: Diagramm zu isoperimetrischen Verhältnissen gegebener Synapsen

4.3. Auswertung

Es wurden drei neuartige Ansätze zur Beschreibung und Deutung morphologischer Eigenschaften bzw. (in)homogener Strukturen erarbeitet. Hierbei handelt es sich um die Herangehensweise zur Bestimmung und Interpretation von:

- Zyklen innerhalb von bestimmten Graphen, die sich aus Skelettierungs-Algorithmen ergeben (Abs. 4.1.3),
- Verhältnissen aller euklidischer Distanzen und geodätischer Wege in Graphen, die sich aus Skelettierungs-Algorithmen ergeben (Abs. 4.1.4),
- isoperimetrisch morphologischen Merkmalen (Abs. 4.2).

Bei Experimenten an existierenden biologischen Versuchsdaten (Abs. 1.2.4, Abs. 1.2.4) hat sich deren Nutzbarkeit bestätigt. Multidimensionale Daten und deren Formen werden bei naturwissenschaftlichen Themen hinterfragt und können nun mathematisch ausgewertet werden.

Selbstverständlich gibt es eine Vielzahl von Ansätzen, mit denen die hier vorgeschlagenen Verfahren zur Bestimmung und Auswertung morphologischer Eigenschaften verbessert werden können. Ein möglicher Ansatz – die Ermittlung und Deutung von Fuzzy-Zyklen – wurde kurz erwähnt.

Die vorliegenden Verfahren bieten neuartige Möglichkeiten für die Bestimmung von Formen multidimensionaler biologischer Bilddaten und die Basis zur Hinterfragung und Erforschung weiterer Ansätze.

5. Implementierung

5.1. Übersicht

Auf Grund der zur Verfügung stehenden multidimensionalen Akquisitionsverfahren (z. B. Fluoreszenzmikroskopie, Spim) können heutzutage Aufnahmen von Bewegungen, biologischen Entwicklungen und Wachstum mit einer Vielzahl von Algorithmen analysiert werden.

Die vorliegende Arbeit befasst sich mit dem Problem der Analyse großer Mengen biologischer multidimensionaler Bilddaten von mehreren Terabyte (Stand 2012). Hierbei handelt es sich um eine Vorbereitung nachfolgender Data-Mining-Schritte [93], die in der Arbeit aber nicht umfassend behandelt werden. Es wurde ein entsprechendes Konzept zur Analyse multidimensionaler Bilddaten erarbeitet, um ein exponentielles Wachstum des Rechenaufwands zu bewältigen und mit entsprechenden Verfahren zu verknüpfen.

Das in Kap. 1.3 beschriebene Konzept wird mit *C++* programmiert und mit Hilfe von *CMake* erstellt. Zur Visualisierung dient die *VTK*-Bibliothek¹ [125]. Die *Boost C++*-Bibliothek wird zur Auswertung von Graphen genutzt. Zur Konstruktion der Widgets der entsprechend neu entwickelten GUIs (*ImagingProjectGui*, *KnowledgeBaseGui* etc.) kommt *QT*² zum Einsatz.

Die Plugins zur Bilddatenanalyse werden mit hierfür nutzbaren Bibliotheken (*ITK*³, *MATLAB*, direkt per *C++*) konstruiert. Durch einfache Erweiterungen neuer Plugins (Abs. 5.3) können neue Operatoren, neue Bausteine eingebunden werden. Die Struktur von *ImagingProjectGui* bewahrt also ebenso die Philosophie, möglichst einfach mit Plugins erweitert zu werden, Operatoren interaktiv zu ergänzen, zu vergleichen und Ergebnisse verschiedener Art abzuspeichern, um die Analyse großer Datensätze zu bewältigen [104].

Ferner dient Xerces zum Erstellen und Generieren von XML-Dateien, in denen Projekte und Abläufe abgespeichert werden.

¹ *VTK* – Visualization Toolkit

² *QT* – Quasar Technologies

³ *ITK* – InsightToolkit

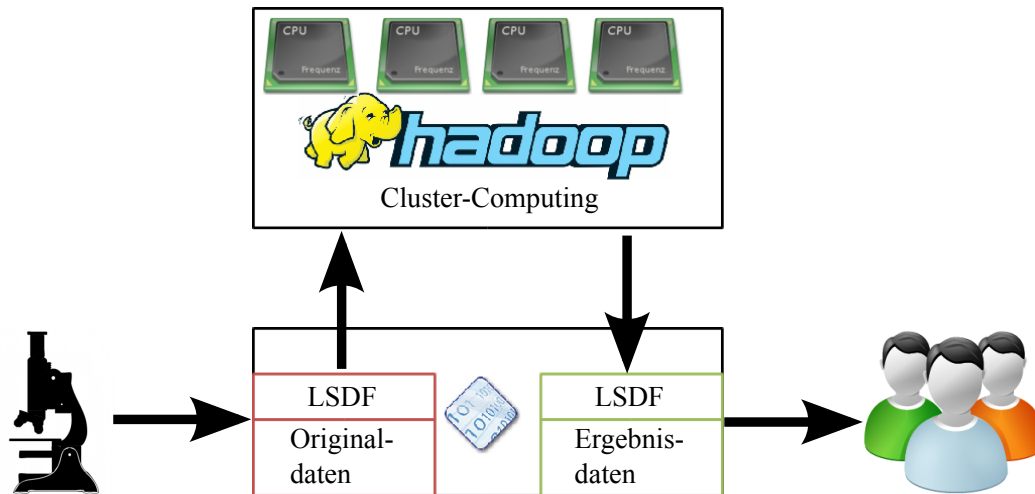


Abb. 5.1.: Abstraktes Konzept des Datenmanagements unter Verwendung des Cluster-Computings [124]

Mittels *Hadoop* und Rechnernetzen können große Datensätze parallel berechnet werden (s. Abb. 5.1). Die berechneten Endergebnisse werden daraufhin in einer *LSDF*-Datenbank eingefügt (Abb. 5.2). Zudem dienen einzelne Operatoren zur Merkmalsextraktion. Die Endergebnisse der Merkmalsextraktionen können anschließend in einer *Ergebnis-XML* abgespeichert werden.

KnowledgeBaseGui ist ein Programm mit dem spezielle Informationen zu bestimmten in Datenbanken abgespeicherten Objekten, vom Nutzer bzw. Experten hinzugefügt werden können. Es besteht so die Möglichkeit, eine Wissensbasis aufzubauen und / oder zu ergänzen.

Alle gewonnenen und in die dazugehörige Datei *Ergebnis-XML* gesicherten Merkmale können anschließend mit *Gait-CAD* ausgewertet und zur Klassifikation genutzt werden. *Gait-CAD* ist eine *MATLAB*-Toolbox für Data-Mining [95].

Zum Datenmanagement, bei den immensen Mengen an Daten des *LSDF*, steht der *DataBrowser* zur Verfügung. So sind schnelle, sichere und informative Zugriffe auf große Bilddaten gewährt [138].

5.2. Datenbank

Neuartige Akquisitionsmöglichkeiten für biologische Untersuchungen führen schnell zu immensen Datenmengen. Das in [104] beschriebene Experiment, in dem das Wachstumsverhalten unter speziellen toxikologischen Einflüssen untersucht wird (Abs. 1.2.4), ist hierfür ein Beispiel.

Um aufschlussreiche Daten entsprechender Zebrabärblinge zu erhalten, werden alle 20 Sekunden Volumendatensätze über ca. 6 Stunden aufgenommen. Zusammengefasst ergibt das $6 \cdot 60 \cdot 60 \cdot \frac{1}{20}$ Zustände mit je 128 MB. Der gesamte Speicherbedarf für einen einzelnen Zebrabärbling beträgt folglich ca. 135 GB⁴. Angesichts den großen Datenmengen und der in der Dissertationsschrift konzipierten Verfahren wird ein Speicherungssystem benötigt, das schnell, sicher und immer verfügbar ist. *LSDF* soll das realisieren [46, 138, 123, 124].

Die Datenbank kommt in jedem Abschnitt der Bildanalyse zur Geltung. Die akquirierten Daten werden über das Datensystem *Large Scale Data Facility (LSDF)* gewartet und in einer individuellen Datenbank abgespeichert [138, 111].

Im Rahmen des Programms "Wissenschaftliches Rechnen" wurden im Jahr 2009 erste Schritte für die Umsetzung von *LSDF* eingeleitet. Mit einer initialen Investition wurde eine prototypische Datenspeicherung aufgebaut, in der erste Daten vom Institut für Toxikologie und Genetik (ITG) abgelegt wurden. Das Arbeitsgebiet *Large Scale Data Management & Analysis* bildet einen Schwerpunkt der wissenschaftlichen Arbeiten im SCC. Und hierfür ist die *LSDF* auch im Hinblick auf **Data Intensive Computing (DIC)** und Datenanalyse – beides beinhaltet Schwerpunkte der Arbeiten in den neu gegründeten Simulation Labs des SCC – die ideale Plattform. Folgende Themen stehen hauptsächlich im Fokus:

- Langzeitarchivierung und Integrität,
- Effektive Übertragung und Verwaltung von Milliarden von Daten (unified namespace),
- Information Life Cycle der Daten,
- Entwicklung von Schnittstellen zur *LSDF* und Anzeige der Datenstrukturen im Browser,
- Datenanalyse und Entwicklung von Datenanalyse-Tools, effektive Bildverarbeitung,
- Entwicklung von Methoden und Werkzeugen im Hinblick auf Data Intensive Computing.

⁴ $128 \text{ MB} \cdot 6 \cdot 60 \cdot 60 \cdot \frac{1}{20} = 135 \text{ GB}$

5.3. ImagingProjectGui

Die Struktur von *ImagingProjectGui* ermöglicht die anwenderfreundliche Erstellung von Projekten zur Be- und Verarbeitung von Bilddaten. Sie bietet die möglichst einfache Erweiterung mit Plugins bzw. Operatoren jeglicher Bibliotheken und die anschließende Berechnung entsprechender Projekte mittels Grid-Computing bzw. *Hadoop*-Clustern.

Mit dem Programm *ImagingProjectGui* können relativ erwartungskonform und anwenderfreundlich Analyseabläufe bzw. Projekte erstellt, weitergeleitet und angewendet werden. Auf einer Oberfläche können einzelne Operatoren ausgewählt und miteinander verknüpft werden.

Außerdem können Zwischenschritte im eigenen Terminal nachverfolgt und mittels *VTK* und eigenen Fenstern entsprechende 2D- und 3D-Daten visualisiert werden.

Ein Beispiel dafür ist die Erstellung eines neuartigen *FCM*-Verarbeitungsverfahrens (Kap. 3). Es dient dazu, die Vorzüge verschiedener Filter mittels *FCM* miteinander zu verknüpfen. Das oben genannte Verfahren wurde mittels *MATLAB* erstellt und an biologischen Volumendaten getestet sowie als Plugin in *ImagingProjectGui* eingebunden.

Bedingt durch den benötigten Speicherplatz waren Berechnungen an einem einzelnen Rechner nicht möglich (*Out of Memory*). Die oben genannten Berechnungen können mit den bereits erwähnten *Hadoop*-Clustern bewältigt werden.

Die graphische Übersicht der Filter bzw. Objekte wird durch eine *QGraphicsScene*⁵ (Overview-GraphicsScene) und ein *QGraphicsView*⁶ erzeugt. Die sich darauf ergebende *ObjectPixmap* stellt ein *IOObject* als Kasten mit Beschriftung dar. Wenn solch ein *ObjectPixmap* zu *QGraphicsScene* hinzugefügt wird, kommt ein *QGraphicsPixmapItem* zurück, über das sich u. a. die Positionen, an die das *ObjectPixmap* gezeichnet wird, festlegen und auslesen lassen.

Wegen des Qualitätsmanagements und der Weiterentwicklung solcher Bibliotheken wurden entsprechende Softwareversionen festgelegt. Somit besteht die Möglichkeit, in Datenbanken die Versionen zu ermitteln, mit den momentanen Programmen zu vergleichen und deren Funktionalität abzuwägen.

Erstellt wurde *ImagingProjectGui* unter dem Betriebssystem Linux Ubuntu. Dadurch wurde die Weiterentwicklung, sowie die Nutzung des Cluster-Computings vereinfacht. Es besteht durchaus die Möglichkeit, *ImagingProjectGui* unter Windows oder Mac zu erweitern.

⁵<http://doc.trolltech.com/4.4/qgraphicsview.html>

⁶<http://doc.trolltech.com/4.4/qgraphicsscene.html>

5.4. KnowledgeBaseGui

KnowledgeBaseGui dient zur Erstellung von Basen mit Expertenwissen. Als Expertenwissen dienen Einschätzungen und Zuordnungen mit menschlichen biologischen Erfahrungen über Formen und morphologischen Eigenschaften in Bezug auf Krankheiten und toxikologische Einwirkungen. Per *KnowledgeBaseGuiWriter* können Datenbanken aus einer Vielzahl von kleineren Bilddaten erweitert werden. Die Daten sind in einer Datenbank per *SQLite* gespeichert und im *LSDF* als Text-Datei integriert⁷. Des Weiteren soll es ebenso im XML-Format gesichert werden. Das entspricht dem Konzept beim *ImagingProjectGui* (Abs. 5.2). Die Datenstruktur sieht folgendermaßen aus:

In der Datenbank ist vor allem der Hauptordner wichtig. Seine innere Struktur bleibt möglichst immer gleich⁸. Die Grundstruktur gewährleistet die Stabilität und stetige Bewahrung von Expertenwissen. Außerdem befinden sich toxikologische Einflüsse und entsprechende Fuzzy-Zuordnungen in der Datenbank (z. B.: Objekt 1, ist bei einem Wert von 70% mit Einfluss A und zu 5% mit Einfluss B befallen, wobei A und B unabhängig voneinander sind.). Der Nutzer kann durch das vorliegende GUI (Abb. 5.3, Abb. 5.4, Abb. 5.5, Abb. 5.6) bei der manuellen Bewertung Aussagen darüber treffen, ob er entsprechendes Objekt zu *dilute* oder *not-dilute* zuordnen kann oder nicht (Alg. 1).

Wenn ein Segment manuell als bekannt gedeutet wird, so kann der entsprechende Fuzzywert für z. B. von Griscelli-Syndrom befallen⁹ bzw. nicht befallenen¹⁰ zugeordnet werden. Sollte dem Nutzer das vorliegende Segment allerdings unbekannt vorkommen, so sind die zugeordneten Werte für die Dilatation irrelevant und der Wert für die Dilatation wird auf *NIL* gesetzt (Alg. 1). Die Einbindung der manuell bestimmten Werte erfolgt direkt in der entsprechenden Datenbank.

Algorithmus 1 *dilute* oder *not-dilute*

Eingabe: bekannt

Ausgabe: dilute

- 1: bekannt $\in [0, 1]$
- 2: **if** *bekannt* < 1 **then**
- 3: dilute $\in (0, \dots, 1)$
- 4: **else**
- 5: dilute $\in (NIL)$
- 6: **end if**

11

⁷im CSV-Format

⁸Hauptordner: /volume ; /surface ; /graph

⁹befallen – *dilute*

¹⁰nicht befallen – *not-dilute*

Es wird somit der Grad des entsprechenden Befalls (s. Abs. 1.2.4) zugeordnet. Wie in Abb. 5.3 ersichtlich, werden zu Beginn alle Objekte als "unbekannt" deklariert. Das bedeutet, dass bei jeglichem manuellen "Überspringen", Werte einfach "unbekannt" bleiben. Wird ein Objekt als "bekannt" eingestuft, muss es mit "bekannt bzw. unbekannt" wieder neu bestückt werden. Weitere Werte bleiben erhalten.

Eine Oberfläche ermöglicht es, morphologische Eigenschaften einzelner zu untersuchender Objekte nutzerfreundlich an spezielle Gegebenheiten anzupassen.

KnowledgeBaseGui ermöglicht es bei den hier zu untersuchenden Datensätzen:

- den Volumendatensatz (s. Abb. 5.3),
- das Oberflächenmodell oder (s. Abb. 5.4),
- das entsprechende Skelettmodell bzw. Netzwerk (s. Abb. 5.5) oder
- alles gleichzeitig mit unterschiedlichen Gewichtungsfaktoren in Bezug auf die Opazität (s. Abb. 5.6)

zu visualisieren.

Erstellt wurde *KnowledgeBaseGui* mit *C++*. Für die Oberfläche kam *QT4* zum Einsatz. Als weitergehende Bibliothek zur 3D-Visualisierung von Bilddaten¹² wurde *VTK* (Visualization Toolkit) angewendet.

Im Folgenden werden Abbildungen einer Synapse, und zwar als Volumendatensatz (Abb. 5.3), Oberflächenmodell (Abb. 5.4), Skelettmodell bzw. Netzwerk (Abb. 5.5) und ein aus allen Faktoren kombiniertes Modell (Abb. 5.6).

¹²mittels *ImagingProjectGui* oder *ProjectProcessor* erstellt und im *LSDF* abgespeichert

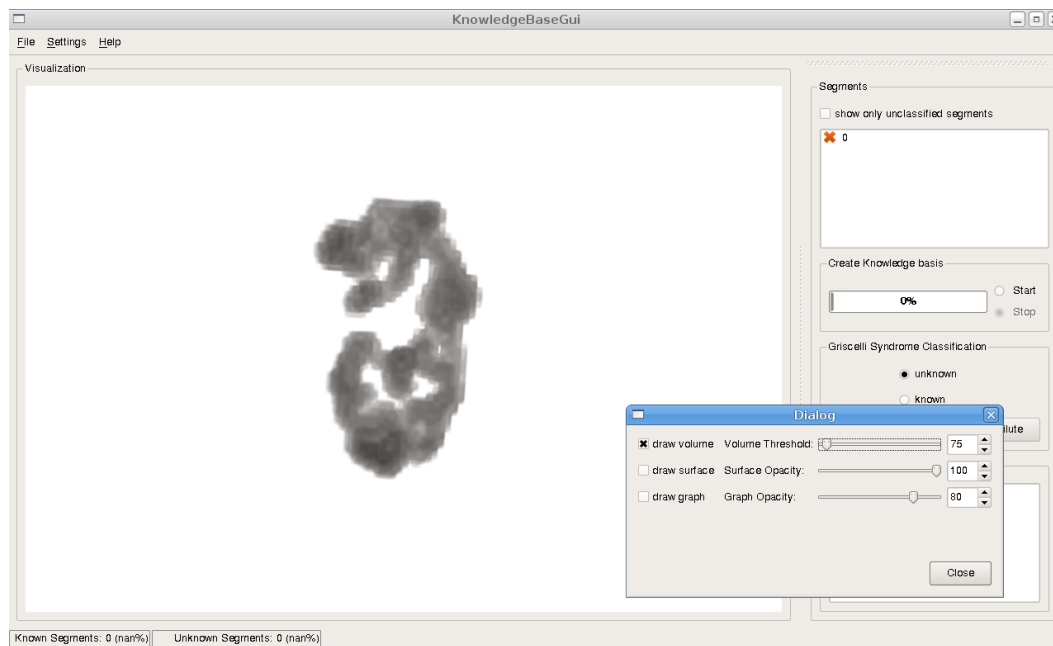


Abb. 5.3.: Volumenansicht unter *KnowledgeBaseGui*

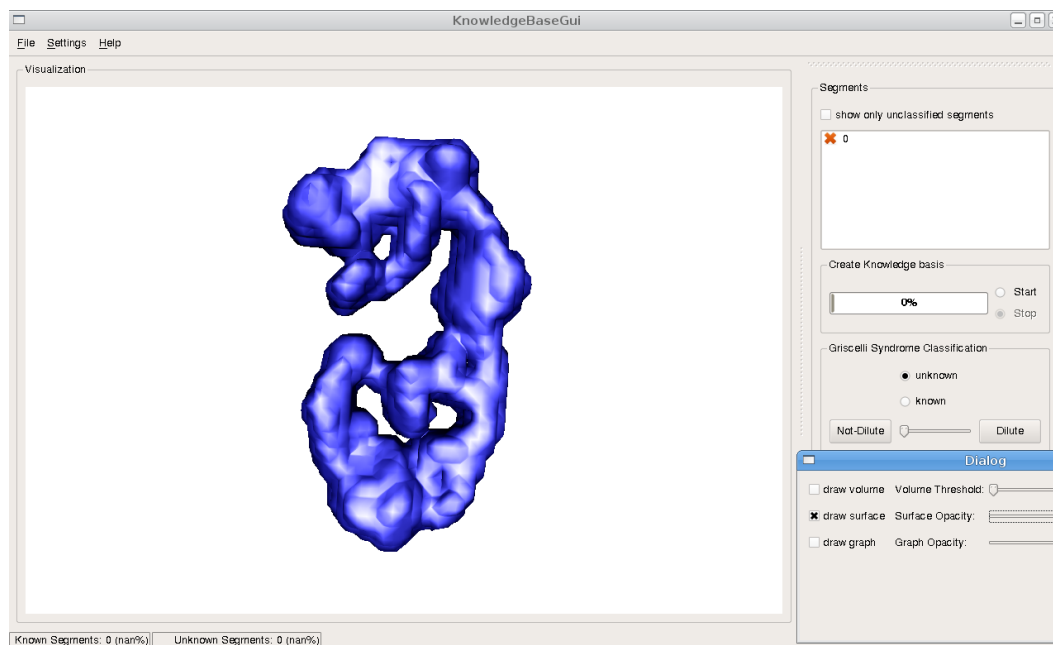


Abb. 5.4.: Ansicht eines Oberflächenmodells unter *KnowledgeBaseGui*

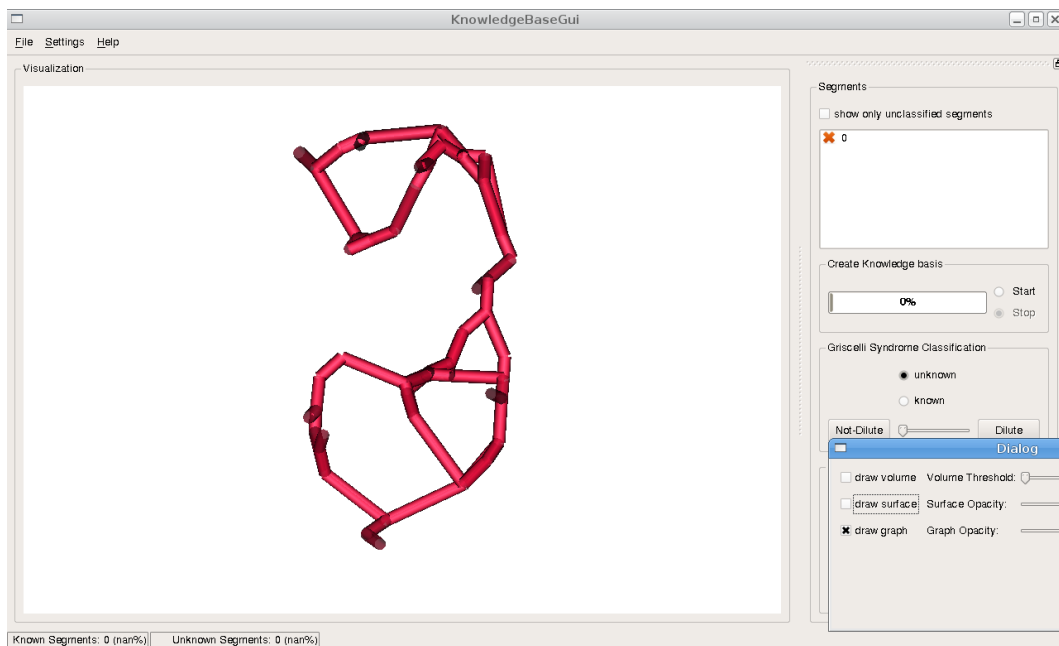


Abb. 5.5.: Ansicht eines Graphen bzw. Netzwerkes unter *KnowledgeBaseGui*

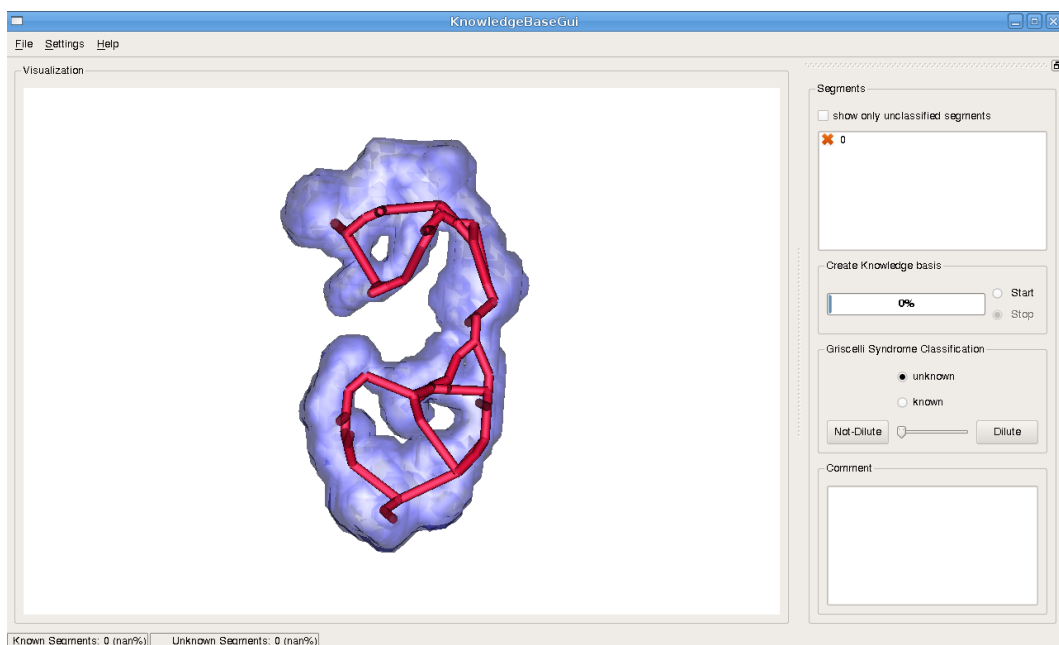


Abb. 5.6.: Volumen-, Oberflächenmodelle und Graphen unter *KnowledgeBaseGui*

5.5. Verknüpfung zu Gait-CAD

Gait-CAD (Abb. 5.7) ist ein Programm zur Klassifikation biomedizinischer Datensätze [95, 93]. Über die gegebenen Daten (Abs. 5.2) im CSV-Typ kann *Gait-CAD* auf entsprechend akquirierte und erarbeitete Bilddaten zugreifen. Dabei können extrahierte Merkmale (Position, Größe, Form, Umfang, Verhältnisse etc.) vorliegen, die zur Evaluation bzw. Klassifikation genutzt werden können. Die Merkmale können allerdings auch, nach entsprechenden Evaluationen, rückwirkend neu bestimmt und z. B. per *Gait-CAD* beigefügt bzw. korrigiert werden.

Im Folgenden wird abgebildet, wie so eine *Gait-CAD*-Oberfläche aussieht (Abb. 5.7).

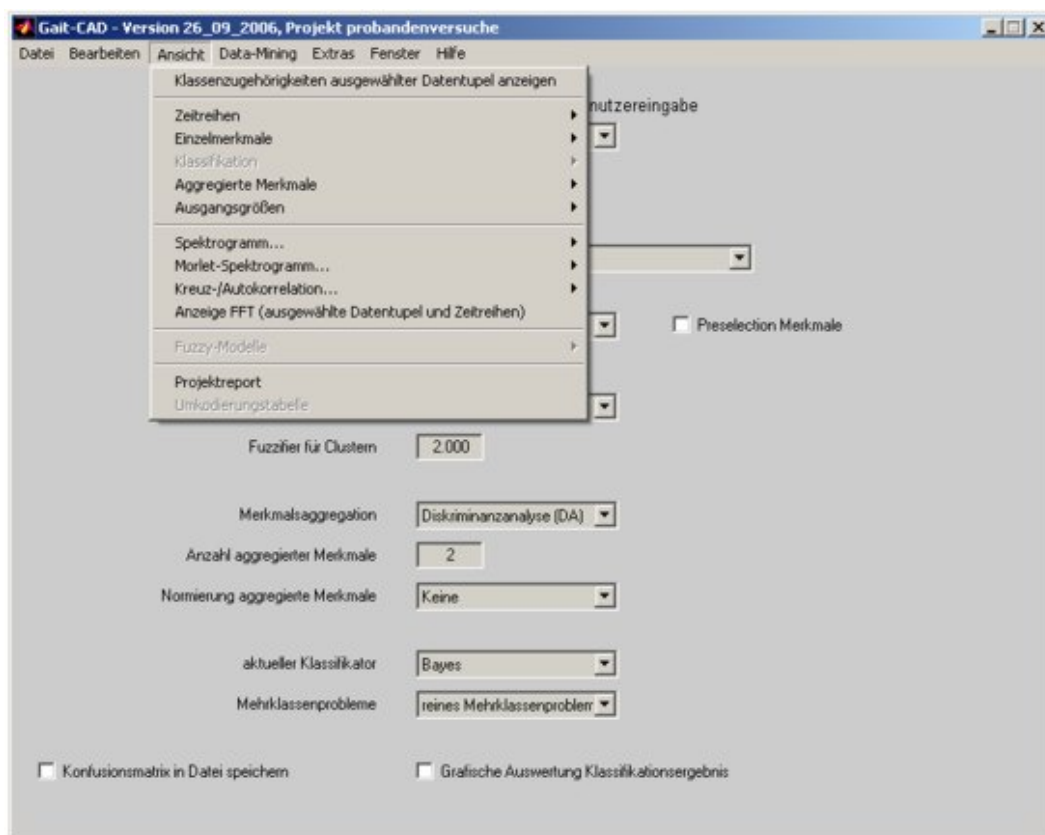


Abb. 5.7.: Ein Abbild der Oberfläche von *Gait-CAD* [95, 93]

Gait-CAD steht als Open-Source zur Verfügung. Ursprünglich diente *Gait-CAD* – wie der Name schon sagt – zur Ganganalyse. Die sich entwickelnden Konzepte und implementierten Algorithmen zum Data-Mining konnten aber auch für andere Probleme genutzt werden. Abstrakt betrachtet kann *Gait-CAD* zur Auswertung von Bewegungen, Trajektorien angewandt werden. Dabei handelt es sich um eine Software, die ebenso für die Evaluation von biologischen Zuständen und Entwicklungen eingesetzt werden kann. Da *Gait-CAD* – wie bereits erwähnt – unter *MATLAB* erstellt wurde, kann die Software auf das hier vorgestellte Datensystem *LSDF* zugreifen.

Die mit den geplanten Abläufen und mit z. B. *Hadoop* ermittelten Merkmale je Zustand, wurden in CSV- oder XML-Dateien abgespeichert. Die CSV- oder XML-Dateien können daraufhin in *MATLAB* bzw. *Gait-CAD* aufgelistet und mit vielen bereits integrierten Verfahren zu weiterführenden Klassifikationen oder auch für das Tracking genutzt werden.

6. Zusammenfassung

Das Ziel der vorliegenden Dissertationsschrift bestand darin, ein neues Konzept zur interaktiven multidimensionalen Bildanalyse zu entwickeln.

Dazu werden in Kap. 2 Verbesserungsvorschläge für multidimensionale Filterverfahren entworfen. Bei den Filterverfahren werden hauptsächlich multidimensionale lokale Operatoren betrachtet. Ohne Qualitätsverlust oder Einschränkungen in der Quantität hinnehmen zu müssen, können mit Hilfe von Separierungen Berechnungen deutlich schneller vollzogen werden. In dem Kapitel werden entsprechende Verfahren hergeleitet, berechnet und experimentell getestet. Wie zu erkennen, sind Nachteile in Form von intensiver Nutzung des Speichers und der notwendigen Berechnungsdauer, gegeben. Das muss bei Entscheidungen der Berechnungsschritte stets mit eingeplant werden.

Kap. 3 beschreibt Lösungsansätze zur Segmentierung mit dynamischen Konturen und Zuordnungen. Vorteile und Nachteile verschiedener schon vorliegender Algorithmen werden beschrieben und an biologischen Bilddaten überprüft. Anschließend werden verschiedene Verfahren getestet und mit den Ergebnissen der gewählten Algorithmen verknüpft.

In Kap. 4 werden mithilfe der Graphentheorie und der Isoperimetrie morphologische Merkmale bestimmt. Durch Filterung und Segmentierung multidimensionaler Bilddaten können Segmente berechnet werden. Mit den berechneten Segmenten können neue Verfahren zur Analyse morphologischer Eigenschaften genutzt werden. Besonders bei biologischen, schwer modellbasiert zu beschreibende Strukturen soll das Verfahren angewendet werden.

Die wesentlichen Ergebnisse der Arbeit sind:

1. Das Ermitteln und das Beweisen des linearen Rechenaufwands von sonst exponentiell rechenaufwändigen multidimensionalen Filterungen ermöglicht eine deutliche Beschleunigung entsprechender separierbarer lokaler Operatoren und somit auch eine enorme, exponentielle Effizienzsteigerung. In der Arbeit wurde die Effizienz in Abs. 2.2 berechnet und dargelegt. Z. B. beim Radius 4 und der Dimension 4 ist in einem Spezialfall die separierte Filterung um über 18.000 % schneller als die nicht-separierte.

2. Die theoretisch ermittelte Effizienzsteigerung wurde experimentell an umfassenden biologischen Bilddaten bestätigt (s. Abs. 2.3).
3. Es konnten neuartige Verfahren zur Segmentierung biologischer, multidimensionaler und uneinheitlicher Daten praktisch angewandt werden. Bei den erarbeiteten Verfahren handelt es sich um Ansätze, die heutzutage der zweidimensionalen Gesichtserkennung dienen, und die, mit Modifikationen bezüglich der höherdimensionalen Elemente, in Qualität, Quantität und Zeitaufwand nun auch in der multidimensionalen Zellsegmentierung eingesetzt werden können (s. Abs. 3.2).
4. Zur Bestimmung der multidimensionalen morphologischen Homogenität wurde die Isoperimetrische Ungleichung (s. Gl. 4.10 und Gl. 4.10) hergeleitet und getestet (s. Abs. 4.2).
5. Mediale Achsen von dreidimensionalen Segmenten (s. Def. 9) biologischer Bilddaten wurden auf neuartige Weise zur Bestimmung gewichteter Graphen eingesetzt (s. Abs. 4.1.1).
6. Anhand der vorliegenden Graphen wurden Zyklen bestimmt, um biologisch morphologische Eigenschaften automatisch zu bestimmen (s. Abs. 4.1.3).
7. Die Relation zwischen den kürzesten euklidischen Verknüpfungen (s. Def. 12) und den kürzesten geodätischen Distanzen (s. Def. 13) wurde neuartig angewandt, um Merkmale bezüglich der Homogenität zu bestimmen (s. Abs. 4.1.4).

Die vorgestellten Algorithmen wurden in C++ programmiert, kompiliert und können zur Bildanalyse genutzt werden.

Derzeit finden Bemühungen statt, die Ergebnisse der Arbeit zur Analyse toxikologischer Effekte bei Zebrafärblingen zu verwenden. Hierfür können nun, auf das jeweilige Problem bezogen, relativ einfache Prozesse erstellt, visualisiert und ausgewertet werden.

Als weitere Schritte sind zu nennen:

- die Einbindung von Parallelisierungs- / Schedulingverfahren,
- die Einbindung separierbarer Operatoren,
- die Einbindung von Segmentierungsverfahren,
- die Einbindung graphentheoretischer Merkmalsextraktoren,
- die Einbindung von Fuzzy-Graphen,
- die Einbindung von Fuzzy-Zyklen,
- die Einbindung von Wissensbasen,

-
- die Implementierung weiterer Algorithmen sowie weiterer Vorgehensweisen,
 - die Bereitstellung und Erweiterung der gewonnenen Wissensdatenbank,
 - die Verbindung gegebener Merkmale und
 - die erweiterte multidimensionale Mensch-Maschine-Interaktion.

7. Literaturverzeichnis

- [1] A. Abbott. Cell Culture: Biology's New Dimension. *Nature*, 424:870 – 872, 2003.
- [2] A. Abbott. Microscopy Marvels: Seeing the System. *Nature*, 459:630 – 631, 2009.
- [3] S. Abeysinghe, T. Ju, M. L. Baker, and W. Chiu. Shape Modeling and Matching in Identifying 3D Protein Structures. *Computer-Aided Design*, 40(6):708 – 720, 2008.
- [4] R. Adams and L. Bischof. Seeded Region Growing. *IEEE Computer Society*, 16:641 – 647, 1994.
- [5] D. Anoraganingrum. Cell Segmentation with Median Filter and Mathematical Morphology Operation. *10th international Conference on Image Analysis and Processing (ICIAP'99)*, 3:1043 – 1046, 1999.
- [6] G. R. Arce and M. P. McLoughlin. Theoretical Analysis of the Max/Median Filter. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 1:60 – 69, 1987.
- [7] I. N. Bankman. *Handbook of Medical Imaging*. Academic Press, 2000.
- [8] H. Bay, T. Tuytelaars, and L. van Gool. SURF: Speeded Up Robust Features. *Computer Vision*, 3954:401 – 417, 2006.
- [9] M. M. Bayer and R. Sinnott. Distributed BLAST in a Grid Computing Context. *Lecture Notes in Computer Science*, 3695:241 – 252, 2005.
- [10] M. Beller, R. Stotzka, and T. Müller. Application of an Interactive Feature-Driven Segmentation. *Biomedizinische Technik*, 49:210 – 211, 2004.
- [11] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, 1981.
- [12] A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr. Interactive Image Segmentation Using an Adaptive GMMRF Model. *Computer Vision - ECCV*, 3021:428 – 441, 2004.

- [13] H. Blum. A Transformation for Extraction new Descriptors of Shape. In *Models for the Perception of Speech and Visual Form*, pages 362 – 380, Cambridge, 1967.
- [14] J. A. Bondy and U. S. R. Murty. *Graph Theory*. Springer, 2008.
- [15] C. Borgelt. Graph Mining: An Overview. In *Proc., 19. Workshop Computational Intelligence*, pages 189 – 203. KIT Scientific Publishing, 2009.
- [16] Y. Y. Boykov and M.-P. Jolly. Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images. *IEEE International Conference on Computer Vision (ICCV) 2001*, 1:105 – 112, 2001.
- [17] U.-D. Braumann, H. Franke, J. Hengstler, J.-P. Kuska, and M. Weber. Graph-Based Quantification of Astrocytes. In *Bildverarbeitung für die Medizin*, pages 379 – 383, 2006.
- [18] I. N. Bronstein and K. A. Semendjajew. *Taschenbuch der Mathematik*. Harri, 2000.
- [19] M. Brown, R. Szeliski, and S. Winder. Multi-Scale Oriented Patches. Technical report, Microsoft Corporation, 2004.
- [20] M. S. Brown, L. S. Wilson, B. D. Dousi, R. W. Gill, and C. Sun. Knowledge-based Method for Segmentation and Analysis of Lung Boundaries in Chest X-Ray Images. *Computerized Medical Imaging and Graphics*, 22:463 – 477, 1998.
- [21] J. Calliess, M. Mai, and S. Pfeiffer. On the Computational Benefit of Tensor Separation for High-Dimensional Discrete Convolutions. *Multidimensional Systems and Signal Processing*, 31:1 – 25, 2010.
- [22] M. B. Cannel, A. McMorland, and C. Soeller. Image Enhancement by Deconvolution. In J. Pawley, editor, *Handbook of Biological Confocal Microscopy*, pages 488 – 500. Springer, 2006.
- [23] J. Canny. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679 – 698, 1986.
- [24] J.-F. Cardoso and P. Comon. Independent Component Analysis, a Survey of Some Algebraic Methods. *IEEE International Symposium on Circuits and Systems*, 2:93 – 96, 1996.
- [25] J. D. Carroll and J. J. Chang. Analysis of Individual Differences in Multidimensional Scaling via an N-way Generalization of "Eckart-Young" Decomposition. In *Psychometrika*, pages 283 – 319, 1970.

-
- [26] D. Chakrabarti and C. Faloutsos. Graph Mining: Laws, Generators, and Algorithms. *ACM Computing Surveys (CSUR)*, 38(1):1 – 69, 2006.
- [27] X. Chen, X. Zhou, and S. T. C. Wong. Automated Segmentation, Classification, and Tracking of Cancer Cell Nuclei in Time-Lapse Microscopy. In *IEEE Transactions on Biomedical Engineering*, volume 4, pages 762–766, 2006.
- [28] K.-S. Chuang, H.-L. Tzeng, S. Chen, J. Wu, and T. J. Chen. Fuzzy C-Means Clustering with Spatial Information for Image Segmentation. *Computerized Medical Imaging and Graphics*, 30:9 – 15, 2008.
- [29] W. F. Clocksin. Automatic Segmentation of Overlapping Nuclei with High Background Variation Using Robust Estimation and Flexible Contour Models. In *ICIAP - Proceedings of the 12th International Conference on Image Analysis and Processing*, pages 682 – 687, 2003.
- [30] G. Cochrane, R. Akhtar, J. Bonfield, L. Bower, F. Demiralp, N. Faruque, R. Gibson, G. Hoad, T. Hubbard, C. Hunter, M. Jang, and S. Juhos. Petabyte-Scale Innovations at the European Nucleotide Archive. *Nucleic Acids Research*, 37:19 – 25, 2008.
- [31] P. Comon. *Mathematics in Signal Processing V*, chapter Tensor Decompositions, pages 1 – 24. Clarendon Press, Oxford, UK, 2002.
- [32] P. Comon. Canonical Tensor Decompositions. *American Institute of Mathematics (AIM) Research Conference Center (ARCC) – Tensor Decompositions Workshop*, pages 1 – 16, 2004.
- [33] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active Shape Models – Their Training and Application. *Computer Vision and Image Understanding*, 61:38 – 59, 1995.
- [34] L. de Lathauwer, B. de Moor, and J. Vandewalle. On the Best Rank-1 and Rank- (R_1, R_2, \dots, R_n) Approximation and Applications of Higher-Order Tensors. In *SIAM, Journal on MATRIX Analysis and Applications*, pages 1324 – 1342, 2000.
- [35] L. de Lathauwer and J. Vandewalle. Dimensionality Reduction in Higher-order Signal Processing and Rank- (R_1, R_2, \dots, R_N) Reduction in Multilinear Algebra. In *Linear Algebra and its Applications*, pages 31 – 55, 2004.

- [36] T. M. DeChiara, D. C. Bowen, D. M. Valenzuela, M. V. Simmons, W. T. Poueymirou, S. Thomas, E. Kinetz, D. L. Compton, E. Rojas, J. S. Park, C. Smith, P. S. DiStefano, D. J. Glass, S. J. Burden, and G. D. Yancopoulos. The Receptor Tyrosine Kinase MuSK Is Required for Neuromuscular Junction Formation In Vivo. *Cell*, 85:501 – 512, 1996.
- [37] M. Dehn. Die Eulersche Formel im Zusammenhang mit dem Inhalt in der Nicht-Euklidischen Geometrie. In *Mathematische Annalen*, 1906.
- [38] T. Deng, M. Kawamata, and T. Higuchi. Design of Two-Dimensional Recursive Digital Filters Based on the Optimal Decomposition of Magnitude Specifications. *Circuits, Systems and Computers*, 3:733 – 756, 1993.
- [39] D. Dunlavy, T. G. Kolda, and P. Kegelmeyer. Tensor Decomposition for Analyzing Multi-Link Graphs. In *SIAM, Parallel Processing for Scientific Computing*, 2008.
- [40] J. C. Dunn. A Fuzzy Relativ of the ISODATA Process and Its Use in Detecting Compact Well-Separated Cluster. *Journal of Cybernetics*, 3:32 – 57, 1972.
- [41] J. Fehr, C. Sauer, H. Kurz, O. Ronneberger, and H. Burkhardt. Identifikation von Zellen in Intaktem Gewebe. In *Bildverarbeitung für die Medizin*, pages 369 – 373, 2006.
- [42] T. Feldmann, I. Mihailidis, D. Paulus, and A. Wörner. Online Full Body Human Motion Tracking Based on Dense Volumetric 3D Reconstructions from Multi Camera Setups. *KI 2010: Advances in Artificial Intelligence*, 6359:74 – 81, 2010.
- [43] M. Foskey, C. L. Ming, and D. Manocha. Efficient Computation of a Simplified Medial Axis. *Journal of Computing and Information Science in Engineering*, 3:274 – 285, 2003.
- [44] M. Frigo and S. Johnson. FFTW: An Adaptive Software Architecture for the FFT. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 3:1381 – 1384, 1998.
- [45] M. Frigo and S. Johnson. FFTW: Fastest Fourier Transform in the West. Technical report, High Energy Astrophysics Division at Harvard-Smithsonian Center for Astrophysics, 2012.
- [46] A. O. García, S. Bourov, A. Hammad, T. Jejkal, J. C. Otte, S. Pfeiffer, T. Schenker, C. Schmidt, J. van Wezel, B. Neumair, and A. Streit. Data Man-

- agement and Analysis at the Large Scale Data Facility. In *6th International Conference on Digital Information Management*, pages 235 – 241, 2011.
- [47] A. O. García, S. Bourov, A. Hammad, J. van Wezel, B. Neumair, A. Streit, J. C. Otte, V. Hartmann, T. Jejkal, P. Neuberger, R. Stotzka, S. Pfeiffer, T. Schenker, and C. Schmidt. *Data-Intensive Analysis for Scientific Experiments at the Large Scale Data Facility*, 2011.
- [48] S. F. Gilbert. *Developmental Biology*. Elsevier, 2010.
- [49] E. Gladilin, R. Eils, and K. Rohr. 3D Formnormalisierung von Zellkernen mit Hilfe einer Elastischen Kugelabbildung. In *Bildverarbeitung für die Medizin*, pages 384 – 388, 2006.
- [50] Y. Gong, C. Mo, and S. E. Fraser. Planar Cell Polarity Signalling Controls Cell Division Orientation During Zebrafish Gastrulation. *Nature*, 430:689 – 693, 2004.
- [51] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice Hall International, 2002.
- [52] I. S. Gousias, D. Rueckert, R. A. Heckemann, L. E. Dyet, J. P. Boardman, A. D. Edwards, and A. Hammers. Automatic Segmentation of Brain MRIs of 2-year-olds Into 83 Regions of Interest. *Neuroimage*, 40(2):672 – 684, 2008.
- [53] A. Haibel, I. Manke, A. Melzer, and J. Banhart. In Situ Microtomographic Monitoring of Discharging Processes in Alkaline Cells. *Journal of Electrochemical Society*, 157:387 – 391, 2010.
- [54] N. Harder, F. Mora-Bermudez, and W. J. Godinez. Automated Analysis of Mitotic Phases of Human Cells in 3D Fluorescence Microscopy Image Sequences. *Medical Image Computing and Computer-Assisted Intervention*, 4190:840 – 848, 2006.
- [55] K. Haris, S. N. Efstratiadis, N. Maglaveras, C. Pappas, J. Gourassas, and G. Louridas. Model-Based Morphological Segmentation and Labeling of Coronary Angiograms. *IEEE Transactions on Medical Imaging*, 18(10):1003 – 1015, 1999.
- [56] R. A. Harshman. Foundations of the PARAFAC Procedure: Models and Conditions for an "Explanatory" Multi-Modal Factor Analysis. In *UCLA Working Papers in Phonetics*, pages 1 – 84, 1970.

- [57] R. A. Harshman and M. E. Lundy. PARAFAC: Parallel Factor Analysis. *Computational Statistics & Data Analysis*, 18:39 – 72, 1994.
- [58] F. Hausdorff. Dimension und äußeres Maß. *Mathematische Annalen*, 79:157 – 179, 1918.
- [59] R. A. Heckemann, J. V. Hajnal, P. Aljabar, D. Rueckert, and A. Hammers. Automatic Anatomical Brain MRI Segmentation Combining Label Propagation and Decision Fusion. *NeuroImage*, 33(1):115 – 126, 2006.
- [60] J. R. Helliwell, P. R. Strickland, and B. McMahon. The Role of Quality in Providing Seamless Access to Information and Data in E-Science; the Experience Gained in Crystallography. *Information Services & Use*, 26:45 – 55, 2006.
- [61] J. Huisken and D. Y. R. Stainier. Selective Plane Illumination Microscopy Techniques in Developmental Biology. *Development*, 136:1963 – 1975, 2009.
- [62] J. Huisken, J. Swoger, S. Lindek, and E. H. K. Stelzer. Selective Plane Illumination Microscopy. In *Handbook of Biological Confocal Microscopy*, pages 672 – 679, 2006.
- [63] L. Ibanez, W. Schroeder, L. Ng, and J. Cates. *The ITK Software Guide*. Kitware Inc., 2005.
- [64] B. Jähne. *Digital Image Processing*. Springer, 2005.
- [65] B. K. Jang and R. T. Chin. Analysis of Thinning Algorithms Using Mathematical Morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:541 – 551, 1990.
- [66] F. Jetzek, R. C.-D., and L. Dreschler-Fischer. Ein Geometrisches Modell für die Zellsegmentierung. *Bildverarbeitung für die Medizin*, 4:121 – 125, 2006.
- [67] M. Kass, A. Witkin, and D. Terzopoulos. *Snake: Active Contour Models*, 1988.
- [68] P. J. Keller, A. D. Schmidt, J. Wittbrodt, and E. H. K. Stelzer. Reconstruction of Zebrafish Early Embryonic Development by Scanned Light Sheet Microscopy. *Science*, 322:1064 – 1069, 2008.
- [69] B. N. Khoromskij. Fast and Accurate Tensor Approximation of a Multivariate Convolution with Linear Scaling in Dimension. *Journal of Computation and Applied Mathematics*, 234:3122 – 3139, 2010.
- [70] R. Knippers. *Molekulare Genetik*. Thieme, 2006.

- [71] T. G. Kolda and B. W. Bader. Tensor Decompositions and Applications. In *SIAM Review*, pages 455 – 500, 2009.
- [72] E. Komatsu, J. Dunkley, M. R.olta, C. L. Bennett, B. Gold, G. Hinshaw, N. Jarosik, D. Larson, M. Limon, L. Page, D. N. Spergel, M. Halpern, R. S. Hill, A. Kogut, S. S. Meyer, G. S. Tucker, J. L. Weiland, E. Wollack, and E. L. Wright. Five-Year Wilkinson Microwave Anisotropy Probe Observations: Cosmological Interpretation. *The Astrophysical Journal*, 180:1 – 49, 2009.
- [73] H. Kopp. *Bildverarbeitung Interaktiv*. B.G. Teubner, Stuttgart, 1997.
- [74] T. Krämer. *Digitisierung und Klassifikation von Signalen*. Universität Karlsruhe; Fakultät für Informatik, 1985.
- [75] M. Krings, A. H. Mahnken, C. Schroer, J. Patommel, W. Kalender, and B. Glasmacher. CT, μ -CT und μ -Tomographie (Synchrotron) der In Vitro Kalzifizierung. *Bildverarbeitung für die Medizin*, 4:444 – 448, 2006.
- [76] A. Krishnan. GridBLAST: a Globus-based high-throughput implementation of BLAST in a Grid computing framework. *Concurrency and Computation: Practice and Experience*, 17:1607 – 1623, 2005.
- [77] Z. Krivá, K. Mikula, N. Peyriéras, B. Rizzi, A. Sarti, and O. Stasová. 3D Early Embryogenesis Image Filtering by Nonlinear Partial Differential Equations. *Medical Image Analysis*, 14:510 – 526, 2010.
- [78] R. P. Kruger, E. L. Hall, S. J. Dwyer, and G. S. Lodwick. Digital Techniques for Image Enhancement of Radiographs. In *International Journal of Bio-Medical Computing*, pages 215 – 238, 1971.
- [79] J. B. Kruskal. Three-Way Arrays: Rank and Uniqueness of Trilinear Decompositions. In *Linear Algebra and Applications*, pages 95 – 138, 1977.
- [80] L. I. Kuncheva. A Theoretical Study on Six Classifier Fusion Strategies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(2):281 – 286, 2002.
- [81] F. Kurugollu, B. Sankur, and A. E. Harmanci. Color Image Segmentation Using Histogram Multithresholding and Fusion. *Image and Vision computing*, 19:915 – 928, 2001.
- [82] H. Lebesgue. *Intégrale, Longueur, Aire*. PhD thesis, Université de Paris, 1902.

- [83] T.-C. Lee, R. L. Kashyap, and C.-N. Chu. Building Skeleton Models via 3-D Medial Surface/Axis Thinning Algorithms. *Graphical Models and Image Processing*, 56:462 – 478, 1994.
- [84] T. Lehmann, W. Oberschelp, E. Pelikan, and R. Repges. *Bildverarbeitung für die Medizin*. Springer, 1997.
- [85] Y.-M. Li and X.-P. Zeng. A New Strategy For Urinary Sediment Segmentation Based on Wavelet, Morphology and Combination Method. In *Computer Methods and Programs in Biomedicine*, pages 162 – 173, 2006.
- [86] G. Lin, U. Adiga, K. Olson, J. F. Guzowski, C. A. Barnes, and B. Roydam. A Hybrid 3D Watershed Algorithm Incorporating Gradient Cues and Object Models for Automatic Segmentation of Nuclei in Confocal Image Stacks. *Cytometry*, 56:23 – 36, 2003.
- [87] X. Lou, F. O. Kaster, M. S. Lindner, B. X. Kausler, U. Kothe, B. Hockendorf, J. Wittbrodt, H. Janicke, and F. A. Hamprecht. Deltr: Digital Embryo Lineage Tree Reconstructor. In *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1557 – 1560. IEEE, 2011.
- [88] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60:91 – 110, 2004.
- [89] C. M. Maa and M. Sonka. A Fully Parallel 3D Thinning Algorithm and Its Applications. *Journal Computer Vision and Image Understanding*, 64:426 – 433, 1995.
- [90] J. P. MacSleyne, J. P. Simmons, and M. de Graef. On the use of moment invariants for the automated analysis of 3d particle shapes. *Modelling and Simulation in Materials Science and Engineering*, 16:045008, 2008.
- [91] C. D. Martin. The Rank of a Tensor. <http://educ.jmu.edu/~martincd/talks/Rank.pdf> (2012), 2007.
- [92] H. Meine and U. Köthe. Image Segmentation with the Exact Watershed Transform. *Visualization, Imaging and Image Processing (VIIP)*, 5:400 – 405, 2005.
- [93] R. Mikut. *Data Mining in der Medizin und Medizintechnik*. Universitätsverlag Karlsruhe, 2008.

-
- [94] R. Mikut, S. Böhlmann, B. Cuno, J. Jäkel, A. Kroll, T. Rauschenbach, B.-M. Pfeiffer, and T. Slawinski. *Fuzzy-Logik und Fuzzy Control - Begriffe und Definitionen*, 1999.
- [95] R. Mikut, O. Burmeister, M. Reischl, and T. Loose. Die MATLAB-Toolbox Gait-CAD. In *Proc., 16. Workshop Computational Intelligence*, volume 10, pages 114–124. Universitätsverlag Karlsruhe, 2006.
- [96] R. Mikut and K. Hilpert. Interpretable Features for the Activity Prediction of Short Antimicrobial Peptides Using Fuzzy Logic. *International Journal of Peptide Research and Therapeutics*, 15:129 – 137, 2009.
- [97] J. N. Mordeson and S. N. Premchand. Cycles and Cocycles of Fuzzy Graphs. *Information Sciences*, 90:39 – 49, 1996.
- [98] K. Mosaliganti, L. Cooper, R. Sharp, R. Machiraju, G. Leone, K. Huang, and J. Saltz. Reconstruction of Cellular Biological Structures from Optical Microscopy Data. *IEEE Transactions on Visualization and Computer Graphics*, 14:863 – 876, 2008.
- [99] D. P. Mukherjee, N. Ray, and S. T. Acton. Level Set Analysis for Leukocyte Detection and Tracking. In *IEEE Transactions on Image Processing*, volume 10, pages 562–572, 2004.
- [100] H.-H. Nagel and T. Krämer. *Einführung in die Mustererkennung*. Universität Karlsruhe (TH), Fakultät für Informatik, 1985.
- [101] P. M. Narendra. A Separable Median Filter of Image Noise Smoothing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 3:20 – 29, 1981.
- [102] T. Nodes and N. J. Gallagher. Two-Dimensional Root Structures and Convergence Properties of the Separable Median Filter. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 31:1350 – 1365, 1983.
- [103] N. Otsu. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9:62 – 66, 1979.
- [104] J. C. Otte, S. Pfeiffer, T. Schenker, and C. Schmidt. *Poster: Concept of Multi-Dimensional Image Analysis with a Flexible Sequence of Operators and Cluster Computing*. European Molecular Imaging Meeting, Leiden, 2011.

- [105] P. Pamfilos. On the Maximum Rank of a Tensor Product. In *Acta Mathematica Hungarica*, pages 95 – 97. Akadémiai Kiadó, co-published with Springer Science+Business Media B.V., Formerly Kluwer Academic Publishers B.V., 1985.
- [106] F. Pampaloni and E. Stelzer. Three-Dimensional Cell Cultures in Toxicology. *Biotechnology and Genetic Engineering Reviews*, 26(1):117 – 137, 2009.
- [107] M. N. Patrenahalli. A Separable MedianFilter for Image Noise Smoothing. *Pattern Analysis and Machine Intelligence*, 1:20 – 29, 1981.
- [108] J. B. Pawley. *Handbook of Biological Confocal Microscopy*. Springer, 2008.
- [109] S. Pfeiffer. In Vitro Vergleich der 3D-Kinematik verschiedener Knieendoprothesen. Master’s thesis, Karlsruhe Institute of Technology (KIT), 2007.
- [110] S. Pfeiffer, M. Mai, W. Globke, and J. Calliess. On Generalized Separation and the Speedup of Local Operators on Multi-Dimensional Signals. In *Multidimensional (nD) Systems, 2009*, pages 1 – 8, 2009.
- [111] S. Pfeiffer, J. C. Otte, T. Schenker, and F. Zhenhua. *Poster: A New Concept for Analyzing Multi-Dimensional Biomedical Imaging Data*. Proc. BioImaging, Karlsruhe, 2010.
- [112] D. L. Pham and J. L. Prince. An Adaptive Fuzzy C-Means Algorithm for Image Segmentation in the Presence of Intensity Inhomogeneities. *Pattern Recognition*, 20:57 – 68, 1999.
- [113] T. Piekos. Confidence Connected Segmentation using ITK. <http://hdl.handle.net/1926/1306> (2012), 2007.
- [114] L. Pinotti, A. Campagnoli, G. Tognon, F. Cheli, V. Dell’Orto, and G. Savoini. Microscopic Method in Processed Animal Proteins Identification in Feed: Applications of Image Analysis. *Biotechnology, Agronomy, Society and Environment*, 8:249 – 251, 2004.
- [115] J. Rasmussen. Skills, Rules, and Knowledge: Signals, Signs, Symbols and Other Distinctions in Human Performance Models. In *IEEE Transactions on Systems: Man and Cybernetics, SMC-13*, volume 9, pages 257 – 266, 1983.
- [116] I. V. Röder, Y. Petersen, K. R. Choi, V. Witzemann, J. A. Hammer III, and R. Rudolf. Role of Myosin Va in the Plasticity of the Vertebrate Neuromuscular Junction In Vivo. In *PLoS ONE*, 2008.

-
- [117] T. Rohlfing, R. Brandt, R. Menzel, and C. R. Maurer jr. Segmentation of Three-Dimensional Images Using Non-Rigid Registration: Methods and Validation with Application to Confocal Microscopy Images of Bee Brains. In *Bildverarbeitung für die Medizin (2003)*, pages 363 – 374, 2003.
- [118] H. Runz and C. Firscher. Einführung zum Schwerpunktthema Datenbanken. *Medizinische Genetik*, 22(2):209 – 212, 2010.
- [119] M. R. Sabuncu, B. T. T. Yeo, K. van Leemput, B. Fischl, and P. Golland. A generative model for image segmentation based on label fusion. *IEEE Transactions on Medical Imaging*, 29(10):1714 – 1729, 2010.
- [120] J. R. Sanes and J. W. Lichtman. Induction, Assembly, Maturation and Maintenance of a Postsynaptic Apparatus. *Nature Reviews Neuroscience*, 2:791 – 805, 2001.
- [121] T. Schenker. Konzept zur effizienten Analyse von biologischen multidimensionalen Bilddaten. Master’s thesis, Karlsruhe Institute of Technology (KIT), 2011.
- [122] C. Schmidt. Konzept zum Datenmanagement großskaliger biologischer Datensätze mit der Integration eines skalierbarem Rechen- und Dateisystems. Master’s thesis, Karlsruhe Institute of Technology (KIT), 2011.
- [123] C. Schmidt, T. Schenker, J. C. Otte, and S. Pfeiffer. *Poster: Automated Analysis of Huge Data Sets with LSDF and Hadoop*. LSDF - Kolloquium 2011, Karlsruhe, 2011.
- [124] C. Schmidt, T. Schenker, J. C. Otte, and S. Pfeiffer. *Poster: Cluster Computing of Multi-Dimensional Volume Data Sets*. LSDF - Kolloquium 2011, Karlsruhe, 2011.
- [125] W. Schröder. *The VTK User’s Guide*. Kitware Inc., 2001.
- [126] D. Schuster. Separierbarkeit Zweidimensionaler Filter. In *12. DAGM-Symposium*, Informatik-Fachberichte, pages 288 – 291, 1990.
- [127] R. Sedgewick. *Algorithmen*. Addison-Wesley, 1992.
- [128] J. Sedlacek. *Einführung in die Graphentheorie*. B.G. Teubner, 1968.
- [129] F. H. Seguin, P. Burstein, P. J. Bjorkholm, F. Homburger, and R. A. Adams. X-ray Computed Tomography with 50- μ m resolution. *Applied Optics*, 24:4117 – 4123, 1985.

- [130] S. Seifert. *Biomechanische Modellierung der Halswirbelsäule Basierend auf Tomografischen Bilddaten*. PhD thesis, Karlsruhe Institute of Technology (KIT), 2007.
- [131] H. Shao, T. Svoboda, and L. van Gool. ZuBuD – Zurich Buildings Database for Image Based Recognition. Technical report, Swiss Federal Institute of Technology, 2003.
- [132] C. J. R. Sheppard and D. M. Shotton. *Confocal Laser Scanning Microscopy*. Springer, 1997.
- [133] O. Shimomura, F. H. Johnson, and Y. Saiga. Extraction, Purification and Properties of Aequorin, a Bioluminescent Protein from The Luminous Hydromedusan, Aequorea. *Journal of Cellular and Comparative Physiology*, 59:223 – 239, 1962.
- [134] G. Silversmit, B. Vekemans, F. E. Brenker, S. Schmitz, M. Burghammer, C. Riekel, and L. Vincze. X-ray Fluorescence Nanotomography on Cometary Matter from Comet 81P/Wild2 Returned by Stardust. *Analytical Chemistry*, 81:6107 – 6112, 2009.
- [135] C. Sommer, C. Strähle, U. Kothe, and F. Hamprecht. ilastik: Interactive Learning and Segmentation Toolkit. In *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 230 – 233. IEEE, 2011.
- [136] J. Steiner. Einfacher Beweis der isoperimetrischen Hauptsätze. *Journal für die reine und angewandte Mathematik*, 18:281 – 296, 1838.
- [137] F. Suard, A. Rakotomamonjy, and A. Benschrair. Détection de Piétons par Stéréovision et Noyaux de Graphes. In *GRETSI05, Louvain-la-Neuve, Belgique*, pages 686 – 686, 2005.
- [138] M. Sutter, V. Hartmann, M. Götter, J. van Wezel, A. Trunov, T. Jejkal, and R. Stotzka. File Systems and Access Technologies for the Large Scale Data Facility. In *Proceedings of Ingrid 2010*, volume 1, pages 239 – 256, 2012.
- [139] J. W. H. Tangelder and R. C. Veltkamp. A Survey of Content Based 3D Shape Retrieval Methods. *Multimedia Tools and Applications*, 39(3):441 – 471, 2008.
- [140] K. Toennies. *Grundlagen der Bildverarbeitung*. Person Studium, 2008.
- [141] S. Treitel and J. L. Shanks. The Design of Multistage Separable Planar Filters. In *IEEE Transactions on Geoscience Electronics*, pages 10 – 27, 1971.

-
- [142] A. Tsai, A. Yezzi jr., and A. S. Willsky. Curve Evolution Implementation of the Mumford-Shah Functional for Image Segmentation, Denoising, Interpolation, and Magnification. *IEEE Transactions on Image Processing*, 10:1169 – 1186, 2001.
- [143] M. Tscherepanow, F. Zoellner, and F. Kummert. Segmentierung Ungefärbter, Lebender Zellen in Hellfeld-Mikroskopbildern. In *Bildverarbeitung für die Medizin (2006)*, pages 359 – 363, 2006.
- [144] R. Twogood and S. Mitra. Computer-Aided Design of Separable Two-Dimensional Digital Filters. In *Acoustics, Speech and Signal Processing, IEEE Transactions on*, pages 165 – 169, 1977.
- [145] J. K. Udupa and S. Samarasekera. Fuzzy Connectedness and Object Definition: Theory, Algorithms and Applications in Image Segmentation. *CVGIP: Graphical Model and Image Processing*, 15:246 – 261, 1996.
- [146] C. Waehlby, I.-M. Sintor, N. F. Erlandsson, G. Borgefors, and E. Bengtsson. Combining Intensity, Edge and Shape Information for 2D and 3D Segmentation of Cell Nuclei in Tissue Sections. In *Journal of Microscopy*, volume 9, pages 67–76, 2004.
- [147] T. White. *Hadoop: The Definitive Guide*. O’Reilly Media, Inc., 2009.
- [148] S. Wörz. *3D Parametric Intensity Models for the Localization of 3D Anatomical Point Landmarks and 3D Segmentation of Human Vessels*. PhD thesis, Universität Hamburg, Department Informatik, 2006.
- [149] P. Yampolsky, S. Gensler, J. McArdle, and V. Witzemann. AChR Channel Conversion and AChR-adjusted Neuronal Survival During Embryonic Development. *Molecular and Cellular*, 37:634 – 645, 2008.
- [150] S. Yang, D. Köhler, K. Teller, T. Cremer, R. Eils, and K. Rohr. Non-Rigid Registration of 3D Microscopy Images for the Normalization of Different Cell Nuclei. *Bildverarbeitung für die Medizin*, pages 364 – 368, 2006.
- [151] Y. Yoshida, K. Dowaki, Y. Matsumura, R. Matsushashi, D. Li, H. Ishitani, and H. Komiyama. Comprehensive Comparison of Efficiency and CO₂ Emissions Between Biomass Energy Conversion Technologies-Position of Supercritical Water Gasification in Biomass Technologies, Biomass and Bioenergy. In *Biomass and Bioenergy*, volume 25, pages 257 – 272. Elsevier, 2003.

- [152] C. Zanella, M. Campana, B. Rizzi, C. Melani, G. Sanguinetti, P. Bourgine, K. Mikula, N. Peyri ras, and A. Sarti. Cells Segmentation from 3-D Confocal Images of Early Zebrafish Embryogenesis. *IEEE Transactions on Image Processing*, 19(3):770 – 781, 2010.
- [153] Zeiss. *SPIM – ein neues Mikroskopieverfahren*, 2005.
- [154] T. Zhang and G. H. Golub. Rank-One Approximation to High Order Tensors. *SIAM, Journal on Matrix Analysis and Applications*, 23:534 – 550, 2001.
- [155] F. Zhenhua, J. Kittler, W. Christmas, X. Wu, and S. Pfeiffer. Face Annotation by Multilinear AAM with Missing Values (M2AAM). In *The 21st International Conference on Pattern Recognition*, 2012.

A. Gegenüberstellung separierter und nicht-separierter Filterung

Wie in Kap. 2 beschrieben, kann eine Vielzahl von nicht-separierten Filtern genauso wie der jeweilige separierte Filter angewandt werden. Im Folgenden werden Beispielberechnungen mit einem separierten Filter und mit einem nicht-separierten Filter gegenübergestellt.

Ein grau-skaliertes zweidimensionales Bild \mathfrak{X} wird als ein $\mathbb{R}^{p_1 \times p_2}$ -Tensor festgelegt. Wie in Abs. 2.2 beschrieben, gilt: $\mathfrak{X} = (x_{j_1, j_2})$, wobei hier x je einem Muxel bzw. Pixel entspricht (s. Def. 1). Als Filter wird bei dem Beispiel ein separierbarer 3x3-Meanfilter gewählt:

$$\mathfrak{F} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}^T \quad (\text{A.1})$$

A.1. Beispiel einer nicht-separierten Filterung

Es gilt:

$$\mathfrak{F} \star \mathfrak{X} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \star \mathfrak{X} \quad (\text{A.2})$$

Für die Berechnung eines einzelnen Muxels x von Bild \mathfrak{X} wird mit dem nicht-separierten Filter \mathfrak{F} :

- 1 mal geladen,
- 9 mal multipliziert,
- 8 mal addiert und
- 1 mal gespeichert.

A.2. Beispiel einer separierten Filterung

Es gilt:

$$\tilde{\mathfrak{F}} \star \mathfrak{X} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \star \mathfrak{X} = \underbrace{\left(\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}^\top \star \underbrace{\left(\begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \star \mathfrak{X} \right)}_{\text{Schritt 1}} \right)}_{\text{Schritt 2}} \quad (\text{A.3})$$

Für die Berechnung eines einzelnen Muxels x von Bild \mathfrak{X} wird mit dem separierten Filter $\tilde{\mathfrak{F}}$ (s. Gl. A.3), in Schritt 1:

- 1 mal geladen,
- 3 mal multipliziert,
- 2 mal addiert und
- 1 mal zwischengespeichert.

Daraufhin wird in Schritt 2:

- 1 mal geladen,
- 3 mal multipliziert,
- 2 mal addiert und
- 1 mal gespeichert.

Bei der endgültigen Berechnung des Muxels x von Bild \mathfrak{X} (d. h. inklusive Schritte 1 und Schritte 2) wird daraufhin:

- 2 mal geladen,
- 6 mal multipliziert,
- 4 mal addiert,
- 1 mal zwischengespeichert und
- 1 mal gespeichert.

A.3. Vergleich von Filterungsbeispielen

Für die Anzahl einzelner Berechnungsschritte eines 3×3 -Filters gilt (s. Abs. A.2):

	separiert	nicht-separiert
Multiplikationen	6	9
Additionen	4	8
Laden	2	1
Speichern	2	1

Tab. A.1.: Berechnungsbeispiel für die Anzahl der einzelnen Schritte eines lokalen 3×3 -Filters

Wiederum, für einen $5 \times 5 \times 5$ -Filter gilt:

	separiert	nicht-separiert
Multiplikationen	15	125
Additionen	12	124
Laden	3	1
Speichern	3	1

Tab. A.2.: Berechnungsbeispiel für die Anzahl der einzelnen Schritte eines lokalen $5 \times 5 \times 5$ -Filters

Die Anzahl der Multiplikationen, Additionen, Lade- und Speicherschritte bei separierten Filterungen steigt, je nach Dimension, linear.

Bei nicht-separierten Filterungen bleibt die Anzahl der Lade- und Speicherschritte stets bei 1. Hingegen die Anzahl der Additionen und Multiplikationen steigt bei nicht-separierten Filterungen exponentiell. Das spiegelt sich in den hier gegebenen Ergebnissen bei Kap. 2 wieder.

B. Verschiedene Softwarepakete zur Bildanalyse

Eigenschaft / Softwarepaket	<i>ITK</i>	<i>VTK</i>	<i>OpenCV</i>	<i>MATLAB</i>	<i>ImageJ</i>
Betriebssystem	++	++	++	+	++
Lizenz	++	++	+	--	++
Besonderheit	++	++	++	+	+
Sprache	+	+	+	++	-
Programmiersprache	+	+	+	0	0
Dokumentation	++	++	0	++	+
Geschwindigkeit	++	++	++	--	-
Speicherplatz	++	++	++	--	-
Benutzerfreundlichkeit	+	+	+	+	+
Community	++	++	+	++	++
GUI	+	++	+	++	++
Schnittstelle	+	+	+	-	+
Modularität	++	++	+	+	+
Parallelität	0	0	0	0	0
Speichermanagement	++	++	++	--	0
Qualitätssicherung	+	++	+	++	-
Weiterentwicklung	+	++	+	++	0
Support	+	++	+	++	+

Tab. B.1.: Eigenschaften verschiedener Softwarepakete

Die Bewertungen basieren auf Erfahrungen und Sicht des Autors:

++	sehr gut
+	gut
0	befriedigend
-	schlecht
--	sehr schlecht

C. Wichtige Symbole und Bezeichnungen

Beim hier gewählten Bezeichnungsapparat wurde ein Kompromiss zwischen einheitlichen und durchgängigen Bezeichnungen einerseits sowie literaturkonformen und einfachen Bezeichnungen andererseits gewählt.

Anmerkungen:

- Sofern nicht anderweitig vermerkt, kennzeichnen fett gedruckte Großbuchstaben Matrizen, fett gedruckte Kleinbuchstaben Vektoren und normal gesetzte Zeichen in Symbolen skalare Werte.
- *Pixel* steht für *Pictures Element*.
- *Voxel* steht für *Volumes Element*.
- *Muxel* steht für *Multidimensional Element*.

Symbol	Bezeichnung
\times	Kreuzprodukt
\otimes	Tensorprodukt
\star	Verknüpfung
∂	Symbol <i>delta</i> der partiellen Ableitung
2D	zweidimensional
3D	dreidimensional
$\alpha(N_i, N_j)$	Winkel zwischen n_i, n_j bzw. N_i, N_j
$A(G)$	Flächen im Graphen G
$Anz_A(G)$	Anzahl der Flächen im Graphen G
$Anz_E(G)$	Anzahl der Kanten im Graphen G
$Anz_N(G)$	Anzahl der Knoten im Graphen G
CT	Computertomographie
d -Dim	d dimensional; $d \in \mathbb{N}$
\mathfrak{D}	abstraktes und multidimensionales Abbild
$\mathfrak{D}(X)$	Grauwert des Abbildes \mathfrak{D} im Bildpunkt X
D	zweidimensionales Abbild
$D(X)$	Grauwert des zweidimensionalen Abbildes D im Bildpunkt X
d.h.	das heißt
exp	Exponentialfunktion
$E(G)$	Kanten im Graphen G
\mathfrak{F}	multidimensionaler Filter
$\mathfrak{F}(\mathfrak{D}(X))$	multidimensionale Filtermaske; $F(X) = F \star X, \forall X : \mathbb{Z}^d \rightarrow \mathbb{R}$
F	zweidimensionaler Filter
$F(D(X))$	zweidimensionale Filtermaske; $F(X) = F \star X, \forall X : \mathbb{Z}^2 \rightarrow \mathbb{R}$
FCM	<i>Fuzzy C-Means</i>
FFT	Fast Fourier Transformation

Symbol	Bezeichnung
H	Hausdorff-Maß (Sphäre)
K	multidimensionale Kugel
$\hat{l}_{Eukl}(N_i, N_j)$	Euklidische und gewichtete Distanz zwischen N_i und N_j
L	Lebesgue-Maß
l_{Geod}	Geodätische Distanz
l_{Eukl}	Euklidische Distanz
$l(N_i, N_j)$	Länge des bisher offenen Pfades von N_i und N_j
$L(N_i, N_j)$	Länge des Fuzzy-Zyklus durch N_i und N_j
\ln	natürlicher Logarithmus
m_α	Filtermaskenlänge in α -ter Dimension $\alpha \in \{1, \dots, d\}$
MRT	Magnetresonanztomographie
\mathbb{N}	natürliche Zahlen
$N(G)$	Knoten im Graphen G
N_i, N_j	(End)Knoten gegebener Graphen
n_i, n_j	Vektoren $\hat{=}$ einer Kante hin zu N_i bzw. N_j
v	Eine in Gl. 4.4 beschriebene Verhältnisfunktion
O.B.d.A.	Ohne Beschränkung der Allgemeinheit
\mathcal{O}	Landau-Kalkül
p_α	Array-/Bildlänge in α -ter Dimension ($\alpha \in \mathbb{N}$)
\mathbb{R}	reelle Zahlen
r_α	Filtermaskenradius in α -ter Dimension $\alpha \in \{1, \dots, d\}$
r	positiver Radius $r \in \mathbb{R}$
S	Segment
$t(N_i, N_j)$	Translation von $n_i: N_i \rightarrow N_j$
T_{sep}	Rechenaufwand separierter Filter
$T_{non-sep}$	Rechenaufwand nicht-separierter Filter
u.a.	unter anderem
v_i	gegebener Vektor ($i \in \mathbb{N}$)
V	Volumen, L
X	Bildpunkt im zweidimensionalen Raum, ($X \in \mathbb{R}$)
\mathfrak{X}	Bildpunkt im d dimensionalen Raum, ($X \in \mathbb{R}$)
x_{j_1, \dots, j_d}	$\mathfrak{X} = (x_{j_1, \dots, j_d})$
Y	Eine in Gl. 4.6 beschriebene Verhältnisfunktion
y_{i_1, \dots, i_d}	Muxel im d dimensionalen Raum
\mathbb{Z}	ganze Zahlen
z.B.	zum Beispiel

D. Abbildungsverzeichnis

1.1	Balance zwischen Qualität, Quantität und Zeitaufwand	13
1.2	Konzept strukturierter multidimensionaler Bildanalyse	19
2.1	Schematische Darstellung von (nicht-)separiert-basierten Herangehensweisen	31
2.2	Diagramm für multidimensionale lokale Operatoren mit Radius vier	36
2.3	Diagramm für vierdimensionale lokale Operatoren mit variierenden Radien .	36
2.4	Verarbeitungszeit abhängig von Tensorordnung und Filterradius	37
2.5	Dauer per FFT- & separierungs-basierter Methode (variierende Radien) . . .	40
2.6	Dauer per FFT- & separierungs-basierter Methode (variierender Tensorordnung)	41
2.7	Original Volumendatensatz in Form eines Quaders	43
2.8	Ergebnis nach nicht-separierter dreidimensionaler Medianfilterung	44
2.9	Ergebnis nach separierter dreidimensionaler Medianfilterung	44
2.10	Abbild nach exakter Medianfilterung i.) 3D-Filter ii.) Folgen von 2D-Filtern	48
3.1	Stark verrauschtes Ursprungsbild i) 2D-Abbild ii) 3D-Projektion	50
3.2	Verschiedene Segmentierungsverfahren	55
3.3	Verschiedene Verknüpfungsabläufe	57
3.4	Originalabbild	61
3.5	Verschiedene Segmentierungsverfahren	62
3.6	Verschiedene Segmentierungsverknüpfungen	64
3.7	Genauigkeit-Rechenzeit-Diagramm 1	68
3.8	Trefferquote-Rechenzeit-Diagramm 1	69
3.9	Genauigkeit-Rechenzeit-Diagramm 2	71
3.10	Trefferquote-Rechenzeit-Diagramm 2	72
3.11	Genauigkeit-Trefferquote-Diagramm	73
4.1	Dreidimensionale Abbildungen von Synapsen (dilute und nicht-dilute)	76
4.2	Phasen zur Erstellung von Graphen	78
4.3	Planare Graphen i) homogen ii) inhomogen	80
4.4	Diagramm zur Anzahl der Zyklen in gegebenen Synapsen	82
4.5	Euklidische Distanz und Geodäte zwischen X_1 und X_2	84
4.6	Diagramm zum Verhältnis euklidischer und geodätischer Distanzen	86
4.7	Diagramm zu isoperimetrischen Verhältnissen gegebener Synapsen	92
5.1	Abstraktes Konzept des Datenmanagements mit Cluster-Computing	96
5.2	Ablauf des Datenverlaufes	97
5.3	Volumenansicht unter <i>KnowledgeBaseGui</i>	102
5.4	Ansicht eines Oberflächenmodells unter <i>KnowledgeBaseGui</i>	102
5.5	Ansicht eines Graphen bzw. Netzwerkes unter <i>KnowledgeBaseGui</i>	103
5.6	Volumen-, Oberflächenmodelle und Graphen unter <i>KnowledgeBaseGui</i>	103
5.7	Ein Abbild der Oberfläche von <i>Gait-CAD</i>	104

E. Tabellenverzeichnis

1.1	Mustererkennung bei Mensch und Maschine	10
2.1	Berechnungszeiten $T_{nicht-sep}$ bei je 30 Berechnungen	33
2.2	Berechnungszeiten T_{sep} bei je 30 Berechnungen	34
2.3	Filterverfahren und Überlegungen zu Möglichkeiten zur Separierung	45
3.1	Auswertung verschiedener Segmentierungsverfahren	65
3.2	Auswertung verschiedener Verknüpfungen	65
3.3	Anzahl der ermittelten Segmente, Größe der Rechenzeit, Genauigkeit und Trefferquote verschiedener Verfahren und der sich daraus ergebenden Verknüpfungen	67
4.1	Anzahl der Zyklen als Merkmal von gegebenen Synapsen	81
4.2	Verhältnis euklidischer und geodätischer Distanzen gegebener Synapsen	86
4.3	Verschiedene Werte vom Lebesgue-Maß einer Kugel mit Radius 1	88
4.4	Verschiedene Werte vom Hausdorff-Maß einer Kugel mit Radius 1	88
4.5	Isoperimetrie als Merkmal von gegebenen Synapsen	91
A.1	Berechnungsbeispiel für die Anzahl der einzelnen Schritte eines lokalen 3x3-Filters	III
A.2	Berechnungsbeispiel für die Anzahl der einzelnen Schritte eines lokalen 5x5x5-Filters	III
B.1	Eigenschaften verschiedener Softwarepakete	V

