

Inaugural—Dissertation  
zur  
Erlangung der Doktorwürde  
der  
Naturwissenschaftlich—Mathematischen  
Gesamtfakultät  
der  
Ruprecht—Karls—Universität  
Heidelberg

vorgelegt von  
Dipl.-Phys. Kurt Sätzler  
aus Schwaigern/Württ.

Tag der mündlichen Prüfung: 27.07.2000

# 3D Rekonstruktion von elektronenmikroskopischen Serienschnitten

Gutachter: Prof. Dr. B. Sakmann  
Prof. Dr. Dr. C. Cremer

### 3D Rekonstruktion von elektronenmikroskopischen Serienschritten

Im Gegensatz zur Lichtmikroskopie, bei der man mit Hilfe der Methode der sog. *optischen Schnitte* die dreidimensionale Gestalt biologischer Objekte erfassen und auswerten kann, ist in elektronenmikroskopischen, physikalischen Schnittserien keine Orientierung der Schnittbilder zueinander gegeben, was diese Aufnahmemethode für hochauflösende, dreidimensionale Untersuchungen zunächst ungeeignet erscheinen läßt. Die vorliegende Arbeit zeichnet jedoch einen neuen Weg vor, der die dreidimensionale Verarbeitung auch großer elektronenmikroskopischer Schnittserien zuläßt. Hierzu werden der Verlust der Orientierung und die beim Ultradünnschneiden auftretenden Stauchungen mathematisch durch allgemeine, lineare Transformationen modelliert. In einem eigens für diese Problemstellung entwickelten Programm (CAR) dienen die korrigierten, digitalisierten Querschnittsdaten der Objekte als Ausgangspunkt für die Rekonstruktion. Dem Rekonstruktionsalgorithmus ist der sog. Delaunaygraph zugrunde gelegt, der eine Zerlegung des Objektes in Tetraeder ermöglicht. Durch Einführung sog. interpolierender Punkte in die Rekonstruktion ist es möglich, auch Bifurkationen und hierarchisch geschachtelte Konturen topologisch richtig zu repräsentieren. Die in dieser Arbeit entwickelten Techniken wurden am Beispiel der sog. *Calyx von Held*, einer Riesensynapse im Hirnstamm der Ratte (Durchmesser ca.  $20\ \mu\text{m}$ ) angewandt. Die *Calyx* wurde in 270 Ultradünnschnitte der Schnittdicke  $70\ \text{nm}$  geschnitten und mit einer Pixelauflösung von etwa  $3\ \text{nm}$  elektronenmikroskopisch erfaßt und digitalisiert. Neben Volumen und Oberfläche der am Terminal beteiligten prä- und postsynaptischen Zellen wurde die Größe des synaptischen Spaltes, die Anzahl, mittlere Größe und Verteilung der sog. *aktiven Zonen* bestimmt, die für den Vorgang der synaptischen Übertragung sehr wichtig sind. Daher werden diese Daten gemeinsam mit den Eckdaten der physiologischen Messungen als Ausgangspunkt für Reaktions-Diffusions-Simulationen der Transmittermoleküle im synaptischen Spalt dienen, welche zu einem besseren Verständnis der synaptischen Übertragung beitragen werden.

### 3D reconstruction from electron microscopical serial sections

This thesis demonstrates the feasibility of 3D reconstructions in high-resolution electron microscopy. In contrast to light microscopy, where a 3D data set of the specimen is obtained by means of *optical sections*, the orientation of ultrathin sections of biological structures has to be reconstructed in electron microscopy. The sectioning process is mathematically modelled by linear transformations, which also compensates for distortions introduced by the sectioning process. The corrections and the further data processing is done by means of a special software program (CAR), which was developed for this purpose. After applying the linear transformations to the digitized object slices, these serve as a basis for the implemented 3D reconstruction algorithm. In the presented algorithm, object space is divided into so-called Delaunay tetrahedra. By introducing additional interpolating points into the reconstruction, hierarchical contours and bifurcations can be handled and the contour data can be converted to a topologically correct surface and volume representation of the object. This approach was used for the reconstruction of the so-called *Calyx of Held*, a giant synapse in the medial nucleus of the trapezoid body of the rat brain stem (diameter of about  $20\ \mu\text{m}$ ). The *Calyx* was dissected in 270 consecutive serial sections of  $70\ \text{nm}$  thickness, photographed under an electron microscope and later digitized with a lateral resolution of about  $3\ \text{nm}$ . Besides measuring volume and surface of the pre- and postsynaptic cells forming the terminal, cleft volume, number, average size and distribution of the so-called *active zones* were estimated, which are important for synaptic transmission. Therefore, this data will be used together with the results of electrophysiological measurements in simulation calculations of transmitter diffusion and reaction in the synaptic cleft, which will lead to a better understanding of synaptic transmission.

## Inhaltsverzeichnis

Abkürzungen	iii
Ein einführender Ausblick	iv
Einleitung	1
Kapitel 1. Von Präparation zur Rekonstruktion - ein Überblick	4
1.1. Die Elektronenmikroskopie	4
1.2. Präparieren und Schneiden	8
1.3. Der Digitalisierungsprozeß	14
1.4. Lösungsansätze zur Behebung der Fehler	19
Kapitel 2. Computer Aided Reconstruction mit CAR	24
2.1. Das Projekt	26
2.2. Das Alignment	28
2.3. Das Pattering	29
2.4. Das Squeezing	31
2.5. Das Contouring	32
2.6. Die Rekonstruktion	33
2.7. Eine Kurzeinführung in CAR	35
Kapitel 3. 3D-Rekonstruktion von Konturdaten	44
3.1. Das Rekonstruktionsproblem	45
3.2. Grundlagen zum 3D Rekonstruktionsalgorithmus	48
3.3. 3D Beschreibung des Boissonat'schen Ansatzes	61
3.4. Interpolation von Zwischenschichten	70
3.5. Vorverarbeitung der Konturdaten	76
3.6. Möglichkeiten der Delaunayrekonstruktion	79
Kapitel 4. Anwendung am Beispiel der <i>Calyx von Held</i>	81
4.1. Die Präparation der <i>Calyx</i>	81
4.2. Aufnahme und Digitalisierung des Bildmaterials	82
4.3. Die Datenerfassung	89
Kapitel 5. Die Rekonstruktion der <i>Calyx von Held</i>	98
5.1. Die synaptische Übertragung	98
5.2. Volumen und Oberfläche	102
5.3. Kontaktfläche	103
5.4. Synaptischer Spalt	104
5.5. Spezialisierte Kontaktfläche	105

Zusammenfassung	111
Die Datenerfassung	111
Der Rekonstruktionsprozeß	112
Die Auswertung	114
Ausblick	115
Literaturverzeichnis	119
Lebenslauf	125
Publikationen	127
Danksagung	129
Anhang A. Die <code>OpenInventor</code> -Bibliothek	132
Die Inventor Datenbank	132
Nodekits	134
Manipulatoren	134
Komponenten	134
Anhang B. Schnelle Rendertechniken	136
Anhang C. Die Datenstruktur der Voronoibibliothek	140

## Abkürzungen

$\forall$	für alle
$\mathbb{N}$	Menge der natürlichen Zahlen
$\mathbb{R}$	Menge der reellen Zahlen
$\ \cdot\ $	euklidische Norm, Betrag von $\cdot$
$x$	Konstante, Vektorkomponente
$\mathbf{x}, \mathbf{X}$	Vektor, Matrix
bzw.	beziehungsweise
CAR	Contour Alignment Reconstruction
CCD	charge coupled device
D	Dimensionen, dimensional
DBV	digitale Bildverarbeitung
EMBL	European Molecular Biology Laboratory
etc.	et cetera
EM	Elektronenmikroskop, Elektronenmikroskopie
GIF	Graphics Interchange Format
IAB	Institute Albert Bonniot
IWR	Interdisziplinäres Zentrum für wissenschaftliches Rechnen
i.a.	im allgemeinen
JPEG	Joint Photographic Experts Group
LGS	lineares Gleichungssystem
LVD	Linienvoronoidiagramm
m.a.W.	mit anderen Worten
MPI	Max-Planck-Institut
NGG	Numerische Geometrie Gruppe
PPM	portable pixmap format
PVD	Punktvoronoidiagramm
TEM	Transmissionselektronenmikroskop
TIFF	Tagged-Image File Format
sog.	sogenannte
STEM	Scanning Transmissionselektronenmikroskop
u.U.	unter Umständen
vgl.	vergleiche
z.B.	zum Beispiel

## Ein einführender Ausblick

Mit Sicherheit wird es — vor allem in einer wissenschaftlichen Abhandlung — zunächst als ungewöhnlich anmuten, den einleitenden Worten, die in die Thematik der vorgestellten Arbeit einführen sollen, einen Ausblick auf das Kommende, noch in den Sternen stehende voranzustellen.

So ungewöhnlich diese Vorgehensweise auch erscheinen mag, um so verständlicher wird sie, wirft man einen kurzen Blick auf die Seele eines Wissenschaftlers und Forschers: Nur der Blick auf die Zukunft, die Wahrnehmung des noch nicht Wahrnehmbaren, die Einschätzung des Kommenden und die Prognosen, deren Richtigkeit erst noch einer Bestätigung bedürfen, sind die Triebfeder und der Motor seines Denkens und Schaffens.

Meist sind es nicht die großen Visionen, die später von den Historikern als solche geadelt werden, sondern kleinere Schritte, die man zu nehmen, und Hürden, die man zu überwinden gedenkt, die einem das zielstrebige Voranschreiten ermöglichen. Erst später, bei einem Blick in die Vergangenheit, können sich dann die vielen kleinen Schritte zu einem einzigen, großen Schritt zusammenfügen und somit als eine Vision interpretieren lassen.

In der Schaffenszeit selbst ist jedoch das richtige Augenmaß für die nächsten geplanten Schritte ein wichtiger Schlüssel zum Erfolg. Ist das Ziel zu weit gesteckt, verliert man es leicht aus den Augen, sind die Ziele zu klein bemessen, geben sie einem keine Richtung. Um das rechte Maß zu finden, bedarf es einer ständigen Kontrolle und Neubewertung der gesteckten Ziele und somit einer Anpassung des eingeschlagenen Weges.

So war auch zu Beginn dieser Arbeit die Zielsetzung klar und in kurzen Worten formulierbar: Um die Natur der Übertragung von Signalen im Gehirn besser verstehen zu können, schien es sinnvoll, neben den aus elektrophysiologischen Experimenten gewinnbaren Eigenschaften und Parametern der synaptischer Kontakte, nun auch eine Möglichkeit zu schaffen, die morphologisch relevant scheinenden Größen eben dieser Kontakte zu erfassen, damit diese als sinnvolle Randbedingungen für Simulations- und Modellrechnungen einfließen können.

Die Erfassung und Auswertung der morphologischen Daten, welches in kurzen Zügen das Thema dieser Arbeit umreißt und nur ein Mosaiksteinchen in der oben formulierten Zielsetzung markiert, erwies

sich als schwieriger und aufwendiger, als zunächst vermutet. So mußten die Schritte oftmals verkleinert und viel Zeit in kleinste, unsichtbar scheinende Detailarbeiten investiert werden, so daß man fast das Gefühl bekam, auf der Stelle zu treten. Ohne diese Feinarbeiten aber wäre die Vollendung dieser Arbeit nicht möglich gewesen.

So sind durch diese Arbeit noch nicht alle zur Vollendung der angestrebten Zielsetzung zu schürzenden Knoten gefunden, noch gelöst, jedoch bietet sie ein gutes Fundament und einen fruchtbaren Nährboden für weitere Projekte, die die vorgestellte Zielsetzung weiter ins Visier nehmen. Insofern ist ein *einführender Ausblick* nicht nur Vision und Motivation für den experimentierenden Wissenschaftler, sondern auch ein Kompaß und Wegweiser zum Verständnis dieser Arbeit.

## Einleitung

Spätestens seit der Entdeckung von Thomas Schwann 1839, daß auch tierisches Leben auf dem Zusammenspiel kleinster lebensfähiger Einheiten, den sog. Zellen beruht, ist der Forscher an einem tieferen Einblick in diese kleinsten Bausteine des Lebens interessiert. Die Tatsache, daß Zellen funktionelle Einheiten wie Gewebe und Organe formen können, macht sie nur noch faszinierender. Krönung des funktionellen Zusammenschlusses und wichtigstes Organ allen intelligenten Lebens formt das Gehirn. Hauptbestandteile dieses wundersamen Gebildes bilden Neuronen, die untereinander Informationen und Signale über sog. synaptische Kontakte austauschen. Dieser Schaltplan ist Grundlage unseres Wissens und Seins, die Flexibilität dieser Zellen, über einen bestimmten Zeitraum die Effizienz der synaptischen Übertragung zu verändern, könnte dabei Ausdruck unserer Fähigkeit sein, zu lernen.

Will man nun einen Blick auf die dreidimensionale, morphologische Struktur dieser Zellen und ihrer Synapsen werfen, muß man sich bildgebender Methoden wie der Lichtmikroskopie bedienen. Dabei galt lange Zeit die von Ernst Abbe Ende des 18.ten Jahrhunderts formulierte Auflösungsgrenze von  $d = \frac{\lambda}{n \sin \varphi}$  als *Schallmauer* der Mikroskopie. Die Größe  $n \sin \varphi$  bezeichnet hierbei die numerische Apertur des Objektivs, welche sich für die beugungsbegrenzte Auflösung des Mikroskops in der Fokusebene verantwortlich zeichnet<sup>1</sup>. Bewegt man nun die Fokusebene des Mikroskops durch das Objekt, kann man dessen dreidimensionale Struktur erfassen. Diese Methode wird als *Optical Sectioning* bezeichnet (siehe [Wi90]). Größter Flaschenhals dieser Methode stellt dabei das ungleiche Auflösungsverhältnis in der Fokusebene zu der Auflösung in Aufnahme-richtung dar, das bis zu einem Faktor 7 schlechter ist.

Erst durch die Einführung von fluoreszierenden Farbstoffen, mit denen man bestimmte Gewebe- oder Zellregionen anfärben kann, gelang die Entkoppelung des Beleuchtungs- und Detektionsprozesses, die in diesen Experimenten als statistisch unabhängig betrachtet werden können. Durch die Entwicklung des konfokalen Mikroskops, bei dem im Aufnahme-, wie auch im Detektionsstrahlengang eine kleine Öffnung, das sog. *Pinhole* eingeführt wird, gelingt es, ungewünschtes Licht außerhalb des Fokusses herauszufiltern und somit das Mißverhältnis von lateraler zu axialer Auflösung auf einen Faktor 2 zu reduzieren ([CC78]

---

<sup>1</sup> $\varphi$  bezeichnet den halben Öffnungswinkel des Objektivs

und [Bra79]). Erreicht werden mit derartigen Mikroskopen Auflösungen von ca.  $250 \times 500 \text{ nm}$ .

Gerade im Hinblick auf morphologische Untersuchung in biologischen Präparaten stellte dieser Fortschritt einen großen Gewinn dar und erlaubte nun dreidimensionale Untersuchungen mit deutlich höherer Präzision als im konventionellen Fluoreszenzmikroskop ([EBS<sup>+</sup>95], [EDB<sup>+</sup>96] und [3]). Weiterentwicklungen des konfokalen Prinzips wie die Axialtomographie ([BHS<sup>+</sup>96], [11] und [12]) und die Theta-Mikroskopie ([LSH95] und [LCS96]) zeigen dabei Wege auf, die es ermöglichen, die axiale Auflösung derart zu verbessern, daß das bisherige Auflösungsverhältnis von 1 : 2 fast in einen isotropen Wert überführt werden kann. Ist es jedoch erforderlich, höhere Auflösungen zu erzielen, als mit diesen Mikroskopen bisher zu erzielen war, hat man nach der Abbeschen Auflösungsgrenze lediglich die Wahl, die numerische Apertur zu vergrößern (siehe  $4\pi$ -Mikroskopie [HSLC94]), oder die Beobachtungswellenlänge zu verkleinern.<sup>2</sup>

Dies führt uns sogleich zu den Elektronenmikroskopen, die uns durch die extrem kurze Wellenlänge hochenergetischer Elektronen theoretisch in den Ångström-Bereich vorstoßen lassen. In biologischen Präparaten, die speziell für diese Mikroskopietechnik eingebettet und möglichst dünn geschnitten werden, erzielen wir immer noch Auflösungen von wenigen Nanometern, was ein deutliches Plus im Vergleich zur Lichtmikroskopie darstellt. Dieses bildgebende Verfahren bildet gleichzeitig die Grundlage der vorliegenden Arbeit, bei der eine 3D Rekonstruktionsmethode vorgestellt werden soll, die trotz der präparativen Nachteile, die die Elektronenmikroskopie im Bezug auf die Untersuchung biologischer Objekte mit sich bringt, eine hochauflösende Rekonstruktion und detaillierte Auswertung der biologischen Strukturen erlaubt.

Im ersten Kapitel werden die Probleme dieses Aufnahmeverfahrens herausgearbeitet, die ein derartiger 3D Rekonstruktionsalgorithmus berücksichtigen muß, um eine repräsentative Rekonstruktion des untersuchten Objektes zu erlauben. Im Gegensatz zu bisher publizierten Untersuchungen ([LMM<sup>+</sup>99], [SH98], [HS95] and [SS97]), die sich verschiedenster Software bedienen, um die einzelnen Schritte von der Datenerfassung über das Registrieren der Bilder bis hin zu Auswertung zu bewerkstelligen, werden in der vorgestellten Arbeit alle Arbeitsschritte mit Hilfe eines speziell entwickelten Programms sozusagen „unter einem Dach“ erfaßt. Dies bietet den entscheidenden Vorteil, daß die beim Rekonstruktionsprozeß vorgenommenen Korrekturen systematisch erfaßbar und unmittelbar auswertbar sind.

---

<sup>2</sup>Es soll hier nicht verschwiegen werden, daß jüngste Errungenschaften der Lichtmikroskopie, wie die Multiphotondetektion ([DSW90] und [HSH95]) oder die gerade von Hell et al. entwickelte STED-Mikroskopie (*Stimulated Emission Detection*) die Abbesche Auflösungsgrenze schon überwunden haben.

Im zweiten Kapitel wird die im Rahmen dieser Arbeit entwickelte Software und Methodik vorgestellt, wobei im dritten Kapitel ausführlich auf den für quantitative Auswertungen entscheidenden Schritt der 3D-Rekonstruktion von zweidimensionalen Konturdaten eingegangen wird.

Abschließend wird die vorgestellte Methode am Beispiel der *Calyx von Held* demonstriert. Die *Calyx* ist eine Riesensynapse, die im Hirnstamm der Ratte<sup>3</sup> angesiedelt und maßgeblich am Prozeß der Ortung von akustischen Signalen beteiligt ist. Eine Besonderheit der *Calyx von Held* ist, daß der Vorgang der synaptischen Transmission direkt am Terminal der beiden beteiligten Neuronen durch *Patch*-Elektroden gemessen und untersucht werden kann. Diese Synapse bildet außerdem die einzigste Kontaktstelle zwischen beiden Neuronen. Aufgrund dieser Tatsache kann die *Calyx von Held* als „Mustersynapse“ im Hinblick auf Simulationsrechnungen zur synaptischen Transmission bezeichnet werden.

Neben den ausführlichen elektrophysiologischen Untersuchungen der letzten Jahre an der *Calyx von Held* (siehe [BHS95], [BS96], [HBS97], [BS98], [BHBS98], [BS99] und [WWB<sup>+</sup>99]), fehlen noch morphologische Daten, die in die Simulationsrechnungen einfließen können. Da zur Anatomie der *Calyx* nur wenige Studien bekannt sind (siehe z.B. [LR66] und [CF88]), wurden diese Daten mit Hilfe der hier entwickelten Methoden zum ersten Mal auf der elektronenmikroskopischen Auflösungsstufe bestimmt (siehe Kapitel 4 und 5). Somit ist die Grundlage für weiterführende Simulationsrechnungen, die ein tiefergehendes Verständnis der Vorgänge bei synaptischer Transmission erhoffen lassen, geschaffen worden.

---

<sup>3</sup>oder anderen Säugetieren

## KAPITEL 1

# Von Präparation zur Rekonstruktion - ein Überblick

Um dem Ziel einer möglichst detailgenauen Rekonstruktion der Morphologie biologischer Strukturen<sup>1</sup> nahe zu kommen, muß man schon bei der Präparation und Bildaufnahme auf möglichst präzise und hochauflösende Techniken zurückgreifen. Wollen wir in der Lage sein, Strukturen in der Größe von wenigen Nanometern aufzulösen, wie z.B. Zellmembranen, so steht einem nach momentanem Stand der Technik nur die *Elektronenmikroskopie* (kurz EM) zur Verfügung.

In den nächsten Abschnitten sollen die physikalischen Grundlagen zur EM aufgeführt und vor allem im Hinblick auf die erzielbare Auflösung in biologischen Präparaten diskutiert werden. Weiterhin werden die speziellen Präparationsmethoden, die zur Aufbereitung der Zellen für die Aufnahme unter dem EM notwendig sind, skizziert. Abschließend wird diskutiert, in wie weit das gewählte Zellmaterial, die Präparations- und Aufnahmetechnik auf die Weiterverarbeitung des digitalisierten Bildmaterials Einfluß hat, und wie weit die möglicherweise auftretenden Fehler korrigierbar sind.

### 1.1. Die Elektronenmikroskopie

Die von Ernst Abbe Ende des 18.ten Jahrhunderts angegebene Formel, die die Auflösung eines Lichtmikroskops in Bezug zur Wellenlänge des verwendeten Lichtes setzt, ist durch den Welle-Teilchen-Dualismus auch auf Elektronen, und somit auf Elektronenmikroskope anwendbar. Dabei gibt die De-Broglie-Wellenlänge der Elektronen eine Abschätzung für die durch ein Elektronenmikroskop (kurz EM) erzielbare Auflösung. Im folgenden soll nun in kurzen Zügen die Funktionsweise eines EM beschrieben und in Bezug auf die auftretenden Aberrationen diskutiert werden. Für eine vertiefende Studie sei das Buch [Rei97] von L. Reimer empfohlen.

**1.1.1. Grundlagen zur Elektronenmikroskopie.** Elektronenmikroskope sind in ihrem Aufbau den Lichtmikroskopen sehr ähnlich. Als Lichtquelle benutzen sie Feldemissionskathoden, anstelle von Glaslinsen benutzen sie magnetische Linsen und das Okular wird durch einen Leuchtschirm ersetzt.

---

<sup>1</sup>im vorliegenden Fall ist es die sog. *Calyx von Held*

Elementarladung	$e_0 = 1.602192 \cdot 10^{-19} C$
Ruhemasse	$m_0 = 9.10956 \cdot 10^{-31} kg$
Ruheenergie	$E_0 = m_0 \cdot c^2 = 511 keV$
Relativistische Masse	$m = \left( \frac{m_0}{\sqrt{1-v^2/c^2}} \right)$
Kinetische Energie	$W_k = m_0 \cdot c^2 \cdot \left( \frac{1}{\sqrt{1-v^2/c^2}} \right)$
Massenverhältnis Proton/Elektron	$\frac{m_p}{m_e} = 1836.1$
Elektronenvolt	$eV = e_0 \cdot 1V = 1.602 W \cdot s$
Lichtgeschwindigkeit in Vakuum	$c = 2.997925 \cdot 10^8 \frac{m}{s}$
Planck-Konstante	$h = 6.6256 \cdot 10^{-34} N m s$

TABELLE 1. Physikalische Konstanten des Elektrons.

Auf ein Elektron im elektrischen Feld wirkt die Kraft

$$(1.1.1) \quad \mathbf{F} = e_0 \cdot \mathbf{E}$$

wobei  $e_0$  die Elementarladung (siehe Tabelle 1) des Elektrons und  $\mathbf{E}$  die Feldstärke des elektrischen Feldes angibt. Wird das Elektron im elektrischen Feld beschleunigt, erhält es die kinetische Energie

$$W_k = e_0 \cdot \int_A^B \mathbf{E} ds$$

Handelt es sich um ein homogenes elektrisches Feld, z.B. in einem Plattenkondensator, an dem die Spannung  $U = \|\mathbf{E}\| \cdot l$  anliegt, erhalten wir in klassischer Rechnung eine Geschwindigkeit  $v$  des Elektrons, die sich wie folgt bestimmt

$$v = \sqrt{2 \frac{e_0 \cdot U}{m}}$$

Bei den am EM eingesetzten Beschleunigungsspannungen von über  $80 kV$  muß relativistisch gerechnet werden. Dies führt uns zu

$$m \cdot c^2 = m_0 \cdot c^2 + e_0 \cdot U$$

und nach Umformungen zu

$$v = c \cdot \sqrt{1 - \left( \frac{1}{1 + \frac{e_0 \cdot U}{m_0 \cdot c^2}} \right)^2}$$

Bei  $80 kV$  Beschleunigungsspannung erhalten wir Geschwindigkeiten von  $v = 1,21 \cdot 10^8 \frac{m}{s}$ . Dies entspricht einer De-Broglie-Wellenlänge

$$(1.1.2) \quad \lambda = \frac{h}{\|\mathbf{p}\|} = \frac{h}{m_0 \cdot v} \sqrt{1 - \frac{v^2}{c^2}}$$

von  $\lambda = 4.2 \cdot 10^{-3} nm$ . Nach der von Ernst Abbe formulierten Auflösung eines Lichtmikroskops von

$$(1.1.3) \quad d = \frac{\lambda}{n \cdot \sin \vartheta}$$

mit der Numerischen Apertur  $A = n \cdot \sin \vartheta$  kann man demnach bei EM um Größenordnungen höhere Auflösungen erzielen als beim Lichtmikroskop. Aufgrund der noch zu besprechenden Aberrationen sind jedoch bei EM keine Aperturen im Bereich von  $A = 1.3$  möglich. Diese liegen in der Praxis in Bereichen von  $A \approx 10^{-2}$ .

Vergleichen wir die theoretisch erzielbare Auflösungen von einem EM bei  $U = 80 kV$  und einer Apertur von  $A_{EM} = 10^{-2}$  mit einem Lichtmikroskop, daß bei einer sichtbaren Wellenlänge von  $\lambda = 400 nm$  und einem Objektiv der numerischen Apertur von  $A_{Licht} = 1.0$  arbeitet, erhalten wir einen theoretischen Auflösungsunterschied von  $400 nm$  zu  $0.4 nm$ , also einen Unterschied von 3 Größenordnungen!

**1.1.2. Auftretende Aberrationen.** Wie auch bei der Lichtmikroskopie treten bei der EM verschiedene Formen von Aberrationen auf, die mit dem Aufnahmesystem zusammenhängen. Betrachten wir also zunächst die Funktionsweise einer magnetischen Linse.

Ein Elektron erfährt in einem Magnetfeld die Lorentz-Kraft

$$(1.1.4) \quad \mathbf{F}_L = -e_0 \cdot \mathbf{v} \times \mathbf{B}$$

Bei Eintritt des Elektrons in ein homogenes Magnetfeld führt die Geschwindigkeitskomponente  $\mathbf{v}_\perp$  dazu, daß das Elektron senkrecht zur Feldrichtung eine Kreisbewegung durchführt. Lorentzkraft und Zentripetalkraft halten sich das Gleichgewicht, was uns zu der Gleichung

$$e_0 \cdot v \cdot B = \frac{m \cdot v^2}{r}, \quad v = \|\mathbf{v}\| \quad \text{und} \quad B = \|\mathbf{B}\|$$

und anschließend zur der Umlauffrequenz

$$(1.1.5) \quad f_Z = \frac{v}{2\pi \cdot r} = \frac{e_0 \cdot B}{2\pi \cdot m},$$

die auch als Zyklotronfrequenz bekannt ist, führt. In der Zeit  $t_z = 1/f_Z$  legt nun das Elektron die Strecke

$$z = \mathbf{v}_\parallel \cdot t_Z = v \cdot t_Z \cdot \cos \alpha$$

zurück, wobei  $\alpha$  den Winkel zwischen der Magnetfeldrichtung  $\mathbf{B}$  und der Einflugrichtung  $\mathbf{v}$  des Elektrons bildet. Schreiben wir obige Gleichung mit Hilfe der Cosinus-Reihe um, so erhalten wir

$$z = \frac{2\pi \cdot m \cdot v}{e \cdot B} \left( 1 - \frac{1}{2}\alpha^2 + O(\alpha^4) \right)$$

1.1.2.1. *Sphärische Aberration.* Betrachten wir nun das Magnetfeld unserer Linse. Sei  $\alpha$  die Abweichung von unserem idealen Strahlengang. So erhalten wir einen Gangunterschied

$$\Delta z \propto C_S \cdot \alpha^2,$$

der die Abweichung vom idealen Gaußfokus der Magnetlinse darstellt. Mit der Vergrößerung  $M$  der Magnetlinse ergibt sich nun im Gaußfokus eine aufgeweitete *Airy-Scheibe* des Objektpunktes von der Größe

$$(1.1.6) \quad \Delta r_S = M \cdot \tan \alpha \cdot C_S \cdot \alpha^2 \approx M \cdot C_S \cdot \alpha^3$$

für kleine Winkel  $\alpha$ . Dies liefert eine Abschätzung der sphärischen Aberration. Da Magnetlinsen starke sphärische Aberrationen aufweisen, wird den Linsen eine Lochblende vorgeschaltet, die Bereiche, in denen starke sphärische Aberration auftritt, aus dem Elektronenstrahl ausblendet.

1.1.2.2. *Chromatische Aberration.* Durch Schwankungen in der Beschleunigungsspannung und in der Linsenstromstärke entstehen Schwankungen in der Brennweite der magnetischen Linse. Diese Aberration läßt sich durch die Gleichung

$$(1.1.7) \quad \Delta r_C = C_C \cdot \alpha \cdot \sqrt{\left(\frac{\Delta U}{U}\right)^2 + \left(\frac{2\Delta I}{I}\right)^2}$$

beschreiben (siehe [Lic79]). So tritt diese Aberration schon durch die Maxwell'sche Geschwindigkeitsverteilung in der Feldemissionselektrode auf und kann durch Unstabilitäten in der Spannungsversorgung der Elektrode und der Linse verstärkt werden. Auch Wechselwirkungen mit dem mikroskopierten Objekt kann, wie wir später sehen werden, zu erheblichen chromatischen Aberrationen führen. Da energiearme Elektronen im Magnetfeld stärker abgelenkt werden als schnelle Elektronen, verhält sich die chromatische Aberration in der EM gerade umgekehrt wie in der Lichtmikroskopie, wo kurzwelliges Licht stärker gebrochen wird als langwelliges.

1.1.2.3. *Axialer Astigmatismus.* Der Vollständigkeit wegen soll auch der Effekt des axialen Astigmatismus nicht unerwähnt bleiben, der durch die Abweichung von der Rotationssymmetrie des Magnetfeldes in der Linse herrührt. So haben verschiedene Bereiche der Linse verschiedene Brennweiten. Durch den Einsatz von sog. Stigmatoren kann jedoch der axiale Astigmatismus einer Magnetlinse sehr gut ausgeglichen werden.

1.1.2.4. *Wahl der optimalen Apertur.* Interessanterweise kann man in der EM eine sog. optimale Apertur bestimmen, bei der die Aberrationen minimal werden. Dies ist dadurch bedingt, daß die beugungsbegrenzte Auflösung, wie sie nach der Abbeschen Formel beschrieben wird (1.1.3) sich umgekehrt proportional zur Apertur verhält wie die sphärische Aberration (1.1.6). Im Gegensatz zur Lichtmikroskopie ist

also nicht eine möglichst hohe Apertur für eine gute Auflösung verantwortlich. Außerdem ist es gerade die sphärische Apertur, die für den Bildkontrast in EM-Bildern sorgt: In *elektronendichten* Bereichen des Präparats werden die Elektronen stärker von ihrer Idealfugbahn abgelenkt und können daher durch die geringe Apertur nicht mehr detektiert werden. Eine Erhöhung der Beschleunigungsspannung jedoch senkt die Wahrscheinlichkeit der Interaktion und der Bildkontrast sinkt. Auch hier muß eine Wahl zwischen optimalem Kontrast und optimaler Auflösung getroffen werden.

**1.1.3. Auflösung in biologischen Präparaten.** Durchfliegen nun die Elektronen ein biologisches Präparat, finden sog. Streuprozesse statt. Man unterscheidet zwei Prozesse: die inelastische Streuung, wie sie durch Interaktion der Elektronen mit den Hüllenelektronen der Atome im Präparat vorkommen. Diese ist vor allem in kleine Winkelbereichen dominierend ( $< 10^{-3}$ ). Zum anderen finden — vor allem in größeren Winkelbereichen — elastische Streuprozesse statt, die durch die Interaktion der Strahlelektronen mit den Atomkernen hervorgerufen wird.

Bei den inelastischen Streuvorgängen verlieren die Strahlelektronen an kinetischer Energie. Der Energieverlust beträgt bei einem Streuvorgang ungefähr  $20\text{ eV}$ . Je dicker ein Präparat ist, je höher ist die Wahrscheinlichkeit, daß ein derartiger Streuvorgang stattfindet, bzw. auch mehrere Male stattfindet. Nach der Formel 1.1.7 und einem typischen Wert  $C_C \approx 2\text{ mm}$ , einer Apertur von  $\alpha = 10^{-2}$  und einer Beschleunigungsspannung von  $80\text{ keV}$  eines Elektrons erhalten wir einen Wert für die chromatische Aberration von  $\Delta r_C \approx 5\text{ nm}$ , was erheblich von der theoretisch erzielbaren Auflösung bei  $80\text{ kV}$  abweicht. In biologischen Präparaten ist also abhängig von der Schnittdicke eine schlechtere Auflösung zu erwarten. In [Lic79] ist folgende Faustformel angegeben:

$$d_{\text{obj}} = \frac{D}{10}$$

wobei  $D$  die Objektstärke des Präparats beschreibt.

**1.1.4. Mögliche Fehlerquellen.** Wie wir gesehen haben, ist aufgrund der auftretenden Aberrationen vor allem in biologischen Präparaten mit einer eingeschränkten Auflösung, die weit von der optimalen Auflösung entfernt ist, zu rechnen. Hinzu kommt noch, daß Spannungsschwankungen zu Aberrationen und Vergrößerungsfehler führen können (siehe [Rei97]). Auch muß darauf geachtet werden, daß die Strahlendosis für das Präparat einen kritischen Grenzwert nicht überschreitet, damit dieses nicht strukturellen Veränderungen unterliegt.

## 1.2. Präparieren und Schneiden

Da durch das Hochvakuum und die hohe Strahlendosis im EM an eine *In-Vivo*-Mikroskopie nicht im entferntesten zu denken ist, muß es

sich beim untersuchten Gewebe um totes Zellmaterial handeln. Da außerdem die Objektdicke das mögliche Auflösungsvermögen bestimmt, muß das Zellgewebe in ausreichend dünne Scheiben geschnitten werden. Um wiederum durch den Schneidevorgang nicht zu starke mechanische Deformationen einzuführen, muß das Gewebe in ein geeignetes Material eingebettet werden. Außerdem muß es gelingen, elektronendichtes Material an den Zellmembranen anzulagern, damit man unter dem EM ausreichend kontrastreiche Aufnahmen machen kann. Die hier vorgestellten Methoden und Daten sind zum größten Teil aus [Rei67] und [Lic79] entnommen.

**1.2.1. Fixation und Einbettung.** Die Methoden zur Fixation und Einbettung unterscheiden sich leicht für prokaryotische und eukaryotischen Zellen. Wir werden hier nur das Vorgehen für eukaryotische Zellen skizzieren. Grundsätzlich gilt, daß bei der Einlagerung der Fixanz und vor der Entwässerung zügig und gründlich gearbeitet werden muß, damit die Fixanz gut und gleichmäßig in die Zellen eindringen kann; hier ist die Gefahr der Erzeugung von Artefakten am höchsten. Während und nach der Entwässerung hingegen besteht eher die Gefahr, daß die erzeugten Gewebelöcke nicht gut zu schneiden sind; die Gefahr von Artefakten ist bei diesen Arbeitsschritten fast vernachlässigbar.

Da die Fixanz in Zellregionen tiefer als  $500\ \mu\text{m}$  von der Gewebeoberfläche entfernt schlecht eindringen kann, müssen die Gewebestücke ausreichend klein geschnitten sein, bevor diese fixiert werden. Sollten dennoch etwas größere Gewebefragmente verwendet werden, so ist eine erhöhte Konzentration der Fixanz zu verwenden oder auf die Methode der sog. *Perfusionsfixierung* zurückzugreifen. Im weiteren werden wir kurz das Vorgehen bei einer Standardpräparationsmethode in der EM von Zellgewebe, die sog. *Doppelfixierung*, beschreiben.

In einem ersten Schritt wird das Gewebematerial in 2-4%igem Glutaraldehyd eingelegt. Dies führt zu einer verstärkten Vernetzung der Proteinstrukturen, wobei das Glutaraldehyd selbst mit in die Zelle eingebaut wird. Dadurch erhält das Gewebematerial zusätzlich Stabilität. Danach muß das Gewebe gründlich in einer Pufferlösung gewaschen werden.

Nun kommt es zum zweiten Fixierungsschritt: Das Gewebe wird in Osmiumtetroxid eingelegt, was zu einer Fixierung der Phospholipide führt. Dadurch wird die Struktur des Gewebematerials abermals verfestigt und durch die Einlagerung des Osmiums als elektronenoptisch dichtes Material zusätzlich kontrastiert. Nach diesem Schritt wird das Gewebe erneut gründlich ausgewaschen.

In einem dritten Schritt, der auch als dritter Fixierungsschritt angesehen werden kann, wird das Gewebe in eine wäßrige Uranylacetatlösung eingelegt. In diesem wäßrigen Milieu erfährt die Zellstruktur kaum noch Veränderungen. Das Uranylacetat stabilisiert vor allem Membranen und DNS und kontrastiert das Gewebe gleichzeitig durch

die weitere Einlagerung eines Schwermetalls. Uranylacetat kann ebenso wie Bleicitrat zur Nachkontrastierung der Ultradünnschnitte verwendet werden.

Nun wird das Gewebematerial in einer aufsteigenden Alkohol- oder Acetonreihe langsam entwässert, da bei der späteren Aufnahme eingelagertes Wasser zur Zerstörung des Gewebes führen könnte. Außerdem dient das Ethanol oder Aceton zur Lösung des Einbettungsmittels, daß im nächsten Präparationsschritt zur Anwendung kommt.

Bei diesem letzten Schritt, dem Einbettungsschritt, wird das Gewebe in einer stufenweise ansteigenden Epoxidlösung gebadet. Hierdurch wird das Lösungsmittel nach und nach gegen das Einbettungsmittel ausgetauscht. Anschließend wird der Gewebekblock bei Temperaturen um die 60° über 1-2 Tage ausgehärtet. Somit erhalten wir einen polymerisierten Gewebekblock<sup>2</sup>, der als Grundlage für die Ultradünnschnitte dient.

**1.2.2. Ultradünnschnitte.** Ist das Zellmaterial genügend ausgehärtet, kann es an die Vorbereitungen für die Anfertigung der Ultradünnschnitte (auch Dünnschnitte genannt) gehen. Zuerst wird der im vorigen Schritt angefertigte Gewebekblock auf einen Kunstharzzapfen (siehe Abbildung (1.2.1)) aufgebracht und mit diesem verschmolzen (aufpolymerisiert). Dieser Kunstharzzapfen wird benötigt, um den Gewebekblock im Ultramikrotom einspannen und befestigen zu können.

Nun wird die *Region of Interest* mit einer scharfen Rasierklinge unter einem Stereomikroskop herausgetrimmt. Der Anschnitt wird im allgemeinen pyramidenförmig gewählt, wobei die Grundfläche eine trapezoidale Form einnimmt (siehe Abb. (1.2.1)d und (1.2.2)b). Durch diese Trimmtechnik ist gewährleistet, daß die Schnittreihenfolge und die Orientierung der Schnitte im nachhinein noch bestimmbar ist: Die Serienschnitte nehmen eine immer größere Fläche ein, wobei eine bestimmte Seite der trapezoidalen Grundfläche in Schneiderichtung des Ultramikrotoms orientiert wird.

Für die Schneiden in einem Ultramikrotom wählt man speziell vorbereitete, glatte und extrem scharfe Glasbruchkanten oder Diamant. Der Vorschub des Messers ist entweder mechanisch oder thermisch realisiert, wobei die mechanische Realisierung als genauer gilt. Mit den üblichen Ultramikrotomen können Schnitte der Dicken 10 bis 200 nm hergestellt werden, wobei Schnittdicken < 50 nm schon eine hervorragende Einbettung voraussetzen.

Ist der Block im Ultramikrotom eingespannt, kann mit dem Schneiden begonnen werden. Es wird nun die gewünschte Schnittdicke eingestellt, die von der tatsächlich erzielten Schnittdicke abweichen kann (s.u.). Hinter der Schneide befindet sich ein Wasserbad, in dem die

---

<sup>2</sup>oder eine folienartige Gewebescheibe

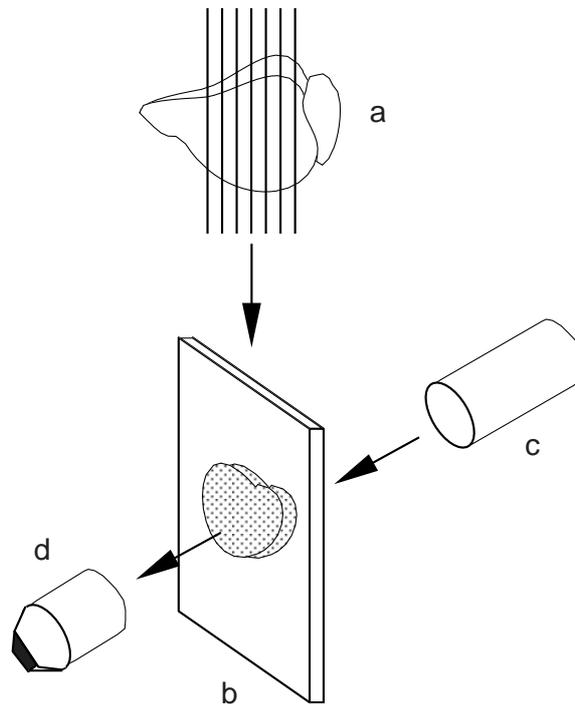


ABBILDUNG 1.2.1. Das Präparat (a) wird zunächst mit einem Vibrator in dickere Scheiben geschnitten. Diese Gewebeproben werden anschließend in einem Epoxidbad in Kunstharz eingebettet. Somit entstehen folienartige Gewebescheiben (b), die auf einen Kunstharzzapfen (c) aufpolymerisiert werden. Dieser wird schließlich zu einer Pyramide angespitzt (d).

Schnitte aufgefangen werden. I.a. bilden die Schnitte sog. Schnittbänder, die dann mit Hilfe eines Haarpinsels oder einem adäquaten Werkzeug auf die kurz vorher mit einer Folie befilmten Netze (engl. *grids*) aufziehen kann (siehe Abbildung (1.2.2)). In bestimmten Fällen können die präparierten Schnitte auf dem Netz nachkontrastiert werden.

Um die tatsächlich erzielte Schnittdicke der Ultradünnschnitte zu bestimmen, gibt es verschiedene Verfahren. Bei diesen Messungen ist jedoch zu beachten, daß es 3 verschiedene Arten von zu messenden Schnittstücken geben kann:

1. die tatsächlich vom Gewebeblock abgeschnittene Schnittdicke
2. die nach dem Schneiden und Aufbringen der Schnitte auf die Trägernetze meßbare Schnittdicke
3. die nach Elektronenbeschuß der Probe verbleibende Schnittdicke

Die unter Punkt 1 angegebene Schnittdicke, ist diejenige, die für eine 3D-Rekonstruktion des beobachteten Objektes von Bedeutung ist. Leider erfahren die Schnitte beim Schneidevorgang größere Stauchungen, die zu einer erhöhten Schnittdicke auf dem Objektträger führen kann.

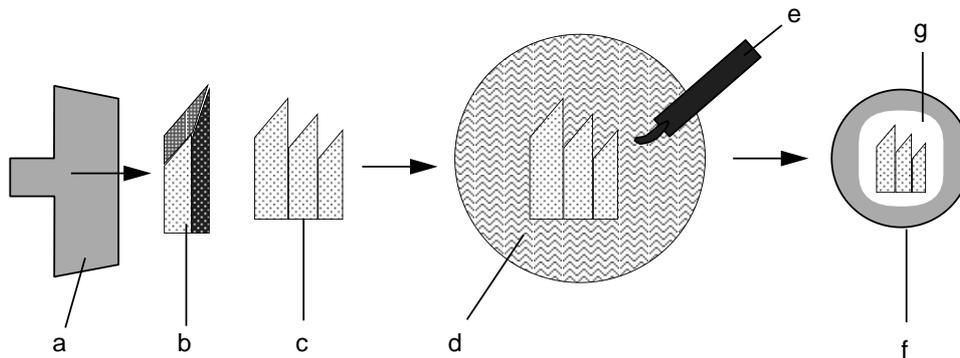


ABBILDUNG 1.2.2. Ablauf des Dünnschneidevorgangs: Mit der Diamant- oder Glasschneide des Ultramikrotoms (a) werden vom zuvor in Kunstharz eingebetteten und getrimmten Gewebsblock (b) Dünnschnitte erzeugt. Diese bilden i.a. Schnittbänder (c), die in einem Wasserbad (d) aufgefangen werden. Diese Schnittbänder werden anschließend mit einem Haarpinsel (e) auf die Folie (g) aufgezogen, mit der das Lochgrid (f) zuvor manuell überzogen wurde.

Eine Nachbehandlung der Schnitte mit Chloroform- oder Xylol-Dampf kann zu einer erneuten Streckung der gestauchten Schnitte führen, wodurch der Stauchungsfehler kompensiert, evtl. aber auch überkompensiert wird. Eine Schnittdickenbestimmung nach Bestrahlung wird auf jeden Fall einen zu kleinen Wert liefern, da ein gewisser Massenverlust des Präparats unter Elektronenbeschuss auftritt (siehe [Rei67]).

Bei der genauesten Methode, die Schnittdicke zu bestimmen, werden die Serienschnitte auf Glasobjektträger aufgebracht und mit einer lichtundurchlässigen Silberschicht bedampft. Mittels der Vielstrahlinterferenz nach Tolansky wird danach die Schnittdicke bestimmt. Man erzielt dabei eine Genauigkeit von  $\pm 1 \text{ nm}$ . Durch Ablösen der Silberschicht mittels Salpetersäure können die Serienschnitte nach der Schnittdickenbestimmung problemlos unter dem EM weiteruntersucht werden.

Eine andere, sehr genaue Methode zur Bestimmung der Schnittdicke stellt die sog. Zweistrahlinterferenzmethode dar, wie sie von Cosselett und Zelander 1960 vorgestellt wurde (siehe [Cos60] und [ZE60]). Mit dieser Methode sind Genauigkeiten von  $2 - 5 \text{ nm}$  erzielbar. Vorteil dieser Methode ist es außerdem, daß sie direkt auf den Trägernetzen ausführbar ist.

Weitere Methoden sind die *Schrägbeschattung*, bei der sich die Schnittdicke aus der Schattenlänge der Schnittkante ergibt. Diese Methode kann jedoch bei keilförmig auslaufenden Schnittkanten sehr ungenau werden ([SP58]). Eine andere Möglichkeit, die Schnittdicke zu

bestimmen, besteht darin, bei Stereobildern, die man durch das Kippen des Präparats im EM erhält ( $\pm 30^\circ$ ), die Parallaxe zu bestimmen. Diese von Williams und Kallmann vorgestellte Methode in [WK55] liefert jedoch gerade für dünne Schnitte große Ungenauigkeiten. Weiterer Nachteil ist, daß hier nur die Schnittdicke nach Bestrahlung des Präparats meßbar wird.

Die beliebteste und im Labor am häufigsten angewandte Methode<sup>3</sup> zur Bestimmung der Schnittdicke ist, diese Anhand der Interferenzfarben, die sich beim Beobachten der Schnitte im Wasserbad ergeben, zu beurteilen. Eine von Bachmann und Sitte (siehe [BS58]) durchgeführte Untersuchung ergab folgende Wertetabelle (im Vergleich zu Werten von Peachey in [Pea58])<sup>4</sup>:

	Bachmann und Sitte	Peachey (Werte in $nm$ )
grau	45	60
silber	45-80	60-90
gold	80-130	90-150
kupfer	130-160	
purpur		150-190
violett	160-200	
blau	200-230	190-240
grün		240-280
gelb		280-320

Diese Werte sind für senkrechten Lichteinfall bestimmt. Bei einer Änderung des Einfallswinkels um den Winkel  $\Theta$  und einem Brechungsindex  $n$  des Objektes, muß der in Tabelle 1.2.2 ermittelte Wert  $D_{\text{measure}}$  nach der Formel:

$$D_{\text{real}} = D_{\text{measure}} \sqrt{1 - \frac{\sin^2 \Theta}{n^2}}$$

korrigiert werden, um die tatsächliche Schnittdicke  $D_{\text{real}}$  zu bestimmen.

**1.2.3. Mögliche Fehlerquellen.** Hauptfehlerquelle im vorgestellten Verfahren ist der Schritt der Fixation. Bei diesem Schritt sind es vor allem die ersten Arbeitsschritte, bei denen das Zellgewebe meist noch relativ instabil ist, wo Fixationsartefakte erzeugt werden können. Die hier auftretenden Fehler und Schrumpfungsartefakte sind allerdings schwer erfaßbar und quantifizierbar. Eine nächste und im Hinblick auf eine 3D-Rekonstruktion wichtige Fehlerquelle entsteht beim Ultradünnschneiden: Hier können die Schnitte stark verformt werden, womit sich auch die eingebettete Objektgeometrie ändert.

<sup>3</sup>nicht die genaueste, aber die praktikabelste Methode

<sup>4</sup>Die vorgestellten Werte wurden [Rei67] entnommen.

### 1.3. Der Digitalisierungsprozeß

Damit die Bilder im Computer weiterverarbeitet werden können, müssen die Aufnahmen der Serienschritte zuerst digitalisiert werden. Es gibt hierbei zwei mögliche Wege, die beschränkt werden können:

- die Digitalisierung per CCD-Kamera, die am EM angebracht ist
- die Digitalisierung der Negative oder Abzüge der EM-Aufnahmen.

Beide Wege bieten verschiedene Vor- und Nachteile, welche im folgenden diskutiert werden sollen. Zunächst muß jedoch eine Abschätzung des auftretenden Datenaufkommens anhand der Auflösung des EM und der Auflösung der diskutierten Aufnahmemethoden gemacht werden.

Wie wir aus Abschnitt 1.1 gelernt haben, ist bei der Aufnahme von biologischen Objekten die Auflösung durch die Objektstärke begrenzt. Bei einer Schnittstärke von ca. 60 nm erwartet man also maximal eine Auflösung von ca. 6 nm in den Aufnahmen selbst (siehe [Lic79]). Bei Berücksichtigung des Abtasttheorems der digitalen Bildverarbeitung, dürfen die Bilder eine maximale Pixelgröße von 3 nm aufweisen. Dies ist ausreichend, um bei nicht spezifisch gefärbten Gewebeproben die Membranen sichtbar zu machen und aufzulösen.

Sind diese Randwerte einmal bestimmt, so hängt das auftretende Datenaufkommen lediglich von der Größe des Beobachtungsgebietes ab. Es läßt sich nach der einfachen Formel

$$(1.3.1) \quad m_{\text{mem}} = \frac{o_x o_y}{r_{\text{obj}}^2}$$

die Anzahl der aufgenommenen Pixel, oder auch das Speicheraufkommen in *bytes* bestimmen, wenn man pro Pixel eine Grauwerttiefe von 8 *bit* annimmt.  $o_x$  und  $o_y$  bezeichnen hierbei die Größe des Beobachtungsfeldes, wobei  $r_{\text{obj}}$  die *isotrope* Pixelauflösung in Objektkoordinaten beschreiben soll.

Ob nun eines der oben angegebenen Aufnahmeverfahren für ein bestimmtes Projekt geeignet erscheint, hängt im wesentlichen von der Anzahl der aufzunehmenden Bilder ab. Diese hängt wiederum von der Rastergröße des elektronenmikroskopisch aufgenommenen Bildes ab.

Bei der CCD-Kamera ist diese Größe durch die Dichte der CCD-Elemente auf dem CCD-Chip bestimmt. So verfügen größere CCD-Kameras über einen CCD-Chip mit bis zu 2048 \* 2048 Bildelementen. Zusammen mit der einstellbaren Vergrößerung des Mikroskops  $s_{\text{mag}}$ , ergibt sich eine Objektauflösung im Präparat nach der Formel:

$$(1.3.2) \quad r_{\text{obj}} = \sqrt{\frac{i_x i_y}{m_{\text{cam}}}} s_{\text{mag}}^{-1} \quad \text{mit} \quad m_{\text{cam}} = p_x p_y$$

wobei  $p_x$  und  $p_y$  die Anzahl der Pixel in  $x$ - bzw.  $y$ -Richtung im Bild (also hier dem CCD-Chip) beschreiben.  $i_x$  und  $i_y$  bezeichnen dabei die Abmessungen des Bildes. Die Größe  $d_{\text{cam}} = \sqrt{\frac{i_x i_y}{p_x p_y}}$  gibt somit

die Aufnahmegröße eines einzelnen CCD-Elementes wieder, isotrope Abmessungen der CCD-Elemente vorausgesetzt.

Bei Aufnahmen auf Negativfilm ist zunächst abzuschätzen, mit welcher Dichte die Oberfläche der Negative abgescannt werden kann. Mit heutigen hochauflösenden Flachbettscannern wie dem SCAI von Zeiss<sup>5</sup>, können Pixelgrößen von  $d_{neg} = 7 \mu m$  erzielt werden<sup>6</sup>. Die Körnung der eingesetzten, harten Negative in der EM ist jedoch so fein, daß sie mit dieser Pixelgröße nicht aufgelöst werden kann.

Um beide Aufnahmemethoden direkt miteinander vergleichen zu können, nehmen wir an, daß die Aufnahmeareale die gleiche Grundfläche einnehmen. Daraus ergibt sich, daß sich das Verhältnis der Pixelgrößen von Kamera und Negativ indirekt proportional zum Verhältnis der Anzahl der Pixel verhalten, also gilt  $d_{cam}/d_{neg} \propto p_{neg}/p_{cam}$ . Bei einer Aufnahmegröße des Negativs von  $7 cm$  und einer Pixelscanfläche  $d_{neg} = 7 \mu m$ , beträgt die Anzahl der Pixel für einen Negativfilm  $p_{neg} = 10000$ , während auch ein sehr großer CCD-Chip nur ungefähr  $p_{cam} = 2048$  Pixel aufzubieten hat. Dies bedeutet, daß die Pixelgrößen eines CCD-Elementes und die eines gescannten Pixels auf einem Negativ mindestens um einen Faktor 5 auseinanderliegen, was zur Folge hat, daß die zu wählende Vergrößerung bei gleichbleibender Auflösung im Präparat ebenfalls um diesen Faktor erhöht werden muß. Dies hat eine erhöhte Strahlendosis, deren das Präparat ausgesetzt wird, zur Folge, was leichter zu einer Beschädigung desselben führen kann.

Betrachten wir die Anzahl der aufzunehmenden Bilder, die bei vorgegebener Auflösung im Präparat mit beiden Techniken aufgenommen werden müßten, wird der Unterschied noch dramatischer. So geht bei der in Formel 1.3.1 angegebenen Anzahl der Pixelelemente  $m_{mem}$  die Pixelauflösung  $r_{obj}$ , und somit indirekt auch die Pixelgröße  $d_{cam}$  oder  $d_{neg}$  (siehe Formel 1.3.2) quadratisch ein. Eine um den Faktor 5 schlechtere Pixelauflösung bedeutet nun gleich, daß ungefähr  $5^2 = 25$  mal so viele Aufnahmen zu machen sind, damit das gleiche Objektareal mit vorgegebener Aufnahmegenauigkeit aufgenommen werden kann.

Ist das zu untersuchende Objekt also hinreichend klein, kommt man selbst bei der Aufnahmemethode mit CCD-Kamera bei der geforderten Abtastgenauigkeit im Präparat mit einer Aufnahme pro Ebene aus. Dann ist das Auflösungsverhältnis vernachlässigbar. Umfaßt das Beobachtungsfeld jedoch — wie bei der Studie der *Calyx von Held* — ca.  $30 \times 30 \mu m^2$ , so haben wir es mit einer Datenmengen von ca.  $100 Mb$  pro Schnitt zu tun. Dies bedeutet, daß wir bei optimal wählbarer Vergrößerung von 2000 für das Scannen der Negative mit einer Aufnahme pro Schnitt auskommen würden, während man bei der Aufnahme

---

<sup>5</sup>Scanner, der in der Abteilung Biophysik des MPI's für medizinische Forschung in Heidelberg zu finden ist, und mit welchem die vorgestellten Daten gescannt wurden.

<sup>6</sup>dies entspricht 3600 dpi

mit CCD-Kamera auf ca. 25 Bilder pro Schnitt kommt. Diese müssen anschließend noch mühsam zu einem großen Gesamtbild der Ebene zusammengefügt (*gepatcht*) werden; ein nicht zu vernachlässigender Mehraufwand in der Weiterverarbeitung (siehe Abschnitt 2).

Weiterhin wird klar, daß man bei der Durchführung eines Projektes dieser Größenordnung nach geeigneten Kompressionsverfahren Ausschau halten muß, um die entstehenden Datenmengen gut verarbeiten zu können. Auf diese Verfahren soll im folgenden Unterabschnitt kurz eingegangen werden.

**1.3.1. Bildkompression und Filterung.** In der digitalen Bildverarbeitung kann man die bekannten Kompressionsverfahren grob in zwei Klassen einteilen:

- die verlustfreien Kompressionsverfahren (z.B. GIF, TIFF)
- die verlustbehafteten Kompressionsverfahren (z.B. JPEG, Wavelet)

Die in Klammern angegeben Formate sind die bekanntesten, die diesen Verfahren zuordenbar sind. Da die Bilddaten, wie sie beim Scanner oder auch von der CCD-Kamera aufgenommen werden, ein recht hohes weißes Rauschen aufweisen, greifen die verlustfreien Kompressionsverfahren wie GIF oder TIFF bei den nicht vorgefilterten Bildern z.T. überhaupt nicht. In Extremfällen ist die „komprimierte“ Bilddatei größer als die unkomprimiert abgespeicherte Originaldatei (z.B. im PPM-Format).

Somit ist bei derlei Bildmaterial ein Filterschritt, der der Kompression vorgeschaltet wird, zwingend erforderlich. Die Anwendung eines  $3 \times 3$  Medianfilters kann bei dieser Form des Rauschens gute Dienste leisten, ohne daß die Bilder zu flau wirken, wie dies beim Gaußfilter bei größeren Masken sehr schnell der Fall ist. Man erzielt mit anschließender GIF-Kompression, welche speziell für Grauwertbilder optimiert wurde, eine maximale *Kompressionsrate* von einem Faktor  $0.35^7$ . Die Kompressionsrate wird hierbei als Quotient des Speicheraufkommens des Bildes mit Kompression und dem Originalbild definiert:

$$(1.3.3) \quad k_{\text{comp}} = \frac{m_{\text{comp}}}{m_{\text{org}}}.$$

Da in diesem Fall den verlustfreien Kompressionsverfahren ein verlustbehafteter Filterschritt vorausgeht, sollte man ebenfalls einen Blick auf die verlustbehafteten Kompressionsverfahren werfen. Das momentan gängigste Verfahren ist hier das JPEG-Verfahren, wobei sich neuere Verfahren, wie die Wavelet-Kompression, die im neuen Standard JPEG2000 Einzug halten sollen, langsam aber sicher etablieren. Diese neuartigen Verfahren liefern vor allem für sehr hohe Kompressionsraten

---

<sup>7</sup>dies sind Erfahrungswerte, die bei der Auswertung des bestehenden Bildmaterials gesammelt wurden

erstaunlich gute Ergebnisse und sind daher vor allem zur Bildarchivierung größerer Datenmengen geeignet (siehe die Artikel [BBJ99] und [BSS99] für weitere Informationen).

Beim JPEG-Verfahren wird das Bild in  $8 \times 8$  große Subbilder untergliedert. In diesen Subbildern werden dann aufgrund der Frequenzinformation und der gewünschten Kompressionsrate feine Strukturen, d.h. hohe Frequenzanteile gemindert oder unterdrückt. Daher ist eine Rausch- und Bandpaßfilterung in das JPEG-Verfahren schon integriert, was dieses Verfahren in Kombination mit der oben erwähnten Tatsache, daß diese Bilder ein hohes Peakrauschen aufweisen, für diesen Anwendungszweck sehr interessant gemacht hat.

Im folgenden soll in Kürze das konventionelle JPEG-Verfahren<sup>8</sup> vorgestellt und die erzielbaren Kompressionsraten im Vergleich zur Bildqualität gezeigt werden. Diese Kompressionsraten stellen wir nur zur Illustration den mediangefilterten Bildern und abschließender Kompression mit einem verlustfreien Verfahren gegenüber.

Um die Qualität des komprimierten Bildes im Vergleich zum Originalbild betrachten und darüber nicht nur subjektiv<sup>9</sup>, sondern auch durch numerische Werte belegbare *objektive* Aussagen machen zu können, muß es gelingen, einen Maßstab für die Bildqualität zu definieren. Die Tauglichkeit eines solchen Maßstabes hängt natürlich hauptsächlich vom Anwendungsschwerpunkt ab, worauf bei der Auswertung der Qualität des Alignments später noch einmal eingegangen werden soll.

In dieser Abhandlung sollen zwei zueinander verwandte, einfache und pixelorientierte Qualitätsmaße eingeführt werden. Bezeichnen wir  $f(\mathbf{x})$  als Ausgangsbild oder Originalbild und  $\tilde{f}(\mathbf{x})$  als komprimiertes oder modifiziertes Bild. So definieren wir die Differenzfunktion  $g(\mathbf{x}) := \left\| f(\mathbf{x}) - \tilde{f}(\mathbf{x}) \right\|$ , die den „Abstand“ zwischen den beiden zu vergleichenden Bildern beschreibt. So formen die folgenden beiden statistischen Größen ein mögliches Qualitätsmaß, für die Bewertung der *Ähnlichkeit* von Bildern:

$$(1.3.4) \quad \text{MDS} := \overline{g(\mathbf{x})}$$

$$(1.3.5) \quad \text{PSNR} := 10 \log \frac{F_{\text{peak}}^2}{g(\mathbf{x})^2} \text{ mit } F_{\text{peak}} := \max(f(\mathbf{x}))$$

MDS gibt hierbei den Mittelwert der Differenzfunktion an (*Mean Difference Signal*), während die PSNR das Verhältnis vom Peaksignal von Originalbild zur Standardabweichung der Differenzfunktion angibt

---

<sup>8</sup>bekannt auch durch das World-Wide-Web (kurz WWW)

<sup>9</sup>eine der immer noch — vor allem für den Bildauswerter — zuverlässigsten Bewertungsmethoden

(*Peak Signal to Noise Ratio*)<sup>10</sup>. Ein niedriger MDS (hoher PSNR)–Wert bedeutet eine große Übereinstimmung zwischen den untersuchten Bildern, wobei  $MDS \rightarrow 0$  und  $PSNR \rightarrow \infty$  im Grenzfall konvergiert. Bei schlechter Übereinstimmung konvergiert  $MDS \rightarrow 2^8$ , während  $PSNR \rightarrow 0$  konvergiert<sup>11</sup>. Auf diese Werte werden wir noch im Kapitel 4 zurückkommen.

Um etwas Gefühl für die absoluten Werte für MDS und PSNR zu bekommen, haben wir eine Studie durchgeführt, bei der Bilder vom gleichen Präparat aber von völlig unterschiedlichen Regionen des Gewebes aufgenommen und miteinander verglichen wurden. Dies soll in unseren Versuchen einen Richtwert für die Ähnlichkeit des Bildmaterials im generellen geben; oder anders ausgedrückt: eine Richtgröße für die schlechteste mögliche Übereinstimmung zweier Bilder. Bei dieser Studie haben wir die folgenden Werte ermittelt:

$$(1.3.6) \quad MDS_{\text{high}} = 45.06 \pm 6.67$$

$$(1.3.7) \quad PSNR_{\text{low}} = 16.13 \pm 1.47$$

Wie wir den Diagrammen in Abbildung (1.3.1) und (1.3.2) entnehmen können, verhalten sich MDS und PSNR monoton mit dem *Qualitätsparameter* (auch *Quality*), dem freien Parameter bei der JPEG–Kompression. Der absolute Wert des Qualitätsparameters ist von Software zu Software verschieden. Der qualitative Verlauf des unten aufgezeigten Verhaltens bleibt jedoch bestehen. Bei der vorliegenden Arbeit wurde auf die Bildkonvertierungssoftware *Image Alchemy* von *Handmade Software Inc.* zurückgegriffen<sup>12</sup>, mit der natürlich auch diese Studie durchgeführt wurde. In Abbildung (1.3.2) ist schön zu erkennen, daß die Bildqualität mit der Kompressionsrate ein lineares Verhalten aufweist.

Erstaunlich ist vor allem die Tatsache, daß schon ein mit einer 3x3 Maske mediangefiltertes Grauwertbild mit anschließender GIF–Kompression nicht nur in der Kompressionsrate deutlich schlechter ist, als die in der vorliegenden Arbeit verwendete JPEG–Kompression bei einem Qualitätsparameter von 40 und einer Kompressionsrate von im Schnitt 0.2, sondern daß dieser für Peakrauschen gemeinhin als gut geeignet geltende Medianfilter schon deutlich schlechtere MDS und PSNR–Werte ( $4.35 \pm 0.87$  bzw.  $34.01 \pm 1.50$ ) liefert als die vergleichbare JPEG–Kompression ( $3.48 \pm 0.54$  bzw.  $38.05 \pm 0.96$ ). Diesen Unterschied

---

<sup>10</sup>Die PSNR wurde bei dem im Artikel [BBSS99] ausgedescriben Wettbewerbfür verlustbehaftete Kompressionsalgorithmen als objektives Bewertungskriterium gewählt.

<sup>11</sup>2<sup>8</sup> bei 8 *bit* Tiefen Grauwertbildern

<sup>12</sup>Mit diesem ressourcenschonenden und sehr effizient programmierten, kommandozeilenorientierten Softwaretool, konnten in der vorliegenden Arbeit etliche hunderte von *Gb* über Nacht konvertiert werden, was eine nicht zu unterschätzende Zeitersparnis war.

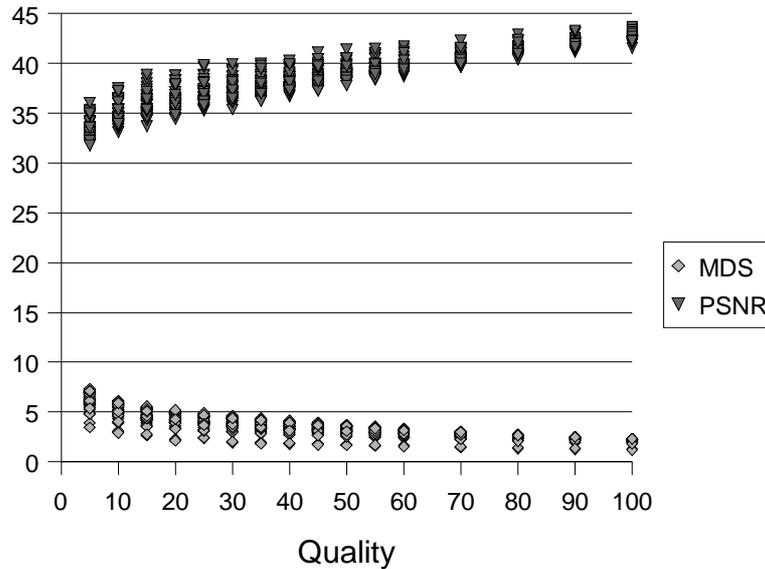


ABBILDUNG 1.3.1. Diese Abbildung zeigt die absoluten Werte für MDS und PSNR für eine Reihe von ausgewerteten Bildern, die zu diskreten Schritten des *Qualitätsparameters* der Bildkonvertierungssoftware *Image Alchemy* von *Handmade Software Inc.* ausgewertet wurden. Das Verhalten sollte bei Verwendung von anderer Software (wie z.B. *xv* oder *gimp*) qualitativ die gleiche Aussage liefern, auch wenn die absoluten Werte für den Qualitätsparameter unterschiedlich sein werden.

kann man selbst anhand der Gegenüberstellung in Abbildung (1.3.3) visuell nachvollziehen.

**1.3.2. Mögliche Fehlerquellen.** Die in diesem Abschnitt besprochenen Digitalisierungsverfahren liefern eine Präzision, die weit höher ist, als die der vorangegangenen Schritte. Das bedeutet auch, daß die Genauigkeit der hier vorgestellten Geräte und Methoden das Endergebnis der Rekonstruktion nahezu unbeeinträchtigt lassen und daher vernachlässigt werden können. Evtl. auftretende Ungenauigkeiten in der Grauwertwiedergabe der Geräte sind dabei für die Rekonstruktion von elektronenmikroskopischen Serienschnitten, wie wir sie untersucht haben, nicht von Bedeutung.

#### 1.4. Lösungsansätze zur Behebung der Fehler

Bevor wir nur zur Diskussion der möglichen Fehler einer Rekonstruktion von biologischen Objekten aus seriellen elektronenmikroskopischen Schnitten kommen, ist es notwendig kurz auf die bisher in der Literatur vorgestellten Verfahren und deren Anwendungsgebiete einzugehen, um zu sehen, welche Methoden am besten bei der in dieser

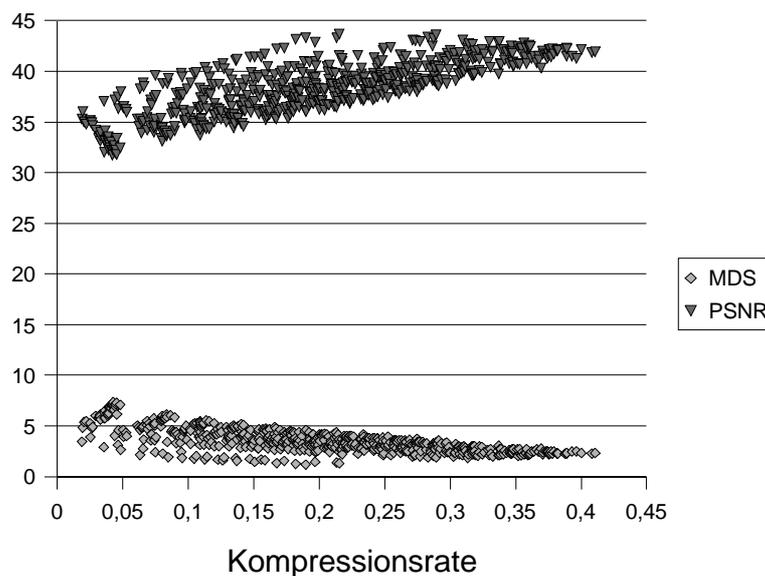


ABBILDUNG 1.3.2. In dieser Abbildung ist die gleiche Meßreihe wie in Abbildung (1.3.1) verwendet worden. Als Abszisse wurde hier jedoch die erzielte Kompressionsrate aufgetragen (siehe 1.3.3).

Arbeit angestrebten Rekonstruktion von Nervenzellen (hier im speziellen die *Calyx von Held*) Anwendung finden könnten.

**1.4.1. Ein historischer Überblick.** Schon bald, nach dem man die EM für biologische Fragestellungen entdeckt hat, und die dazu notwendigen Präparations- und Schneidetechniken verfeinert hatte, hat man sich für die 3D Morphologie von Zellen im allgemeinen interessiert. So entstanden schon 1958 durch Sjöstrand und später durch Berger (siehe [Sjö58] und [Ber73]) graphische Darstellungen oder gar Modellnachbildungen der beobachteten Zellen aus Balsaholz oder ähnlichen Materialien. Hierbei wurden die einzelnen Querschnitte der Zellen maßstabsgetreu aufgezeichnet, ausgesägt und durch Übereinanderstapeln und Kleben der einzelnen Schichten 3D Modelle gefertigt. Das Übereinanderlegen (das sog. *Alignment*) der Schnitte erfolgte demnach noch per Augenmaß anhand der extrahierten Objektkonturen (einen schönen Überblick liefert das Buch [GG78] von Gaunt & Gaunt).

Erst langsam hielten in den 80er Jahren die Computerprogramme im Bereich der 3D-Rekonstruktion von Serienschnitten Einzug. Sie waren z.T. aber noch durch die eingeschränkten Speicher- und Rechenkapazitäten auf kleinere Rekonstruktionen begrenzt ([PG82], [YF87]). Erst Ende der 80er und Anfang der 90er Jahre gab es erste Ansätze, mit computergestützten Programmen nicht nur dreidimensionale Abbildungen und Darstellungen der Zellen, sondern auch morphologische

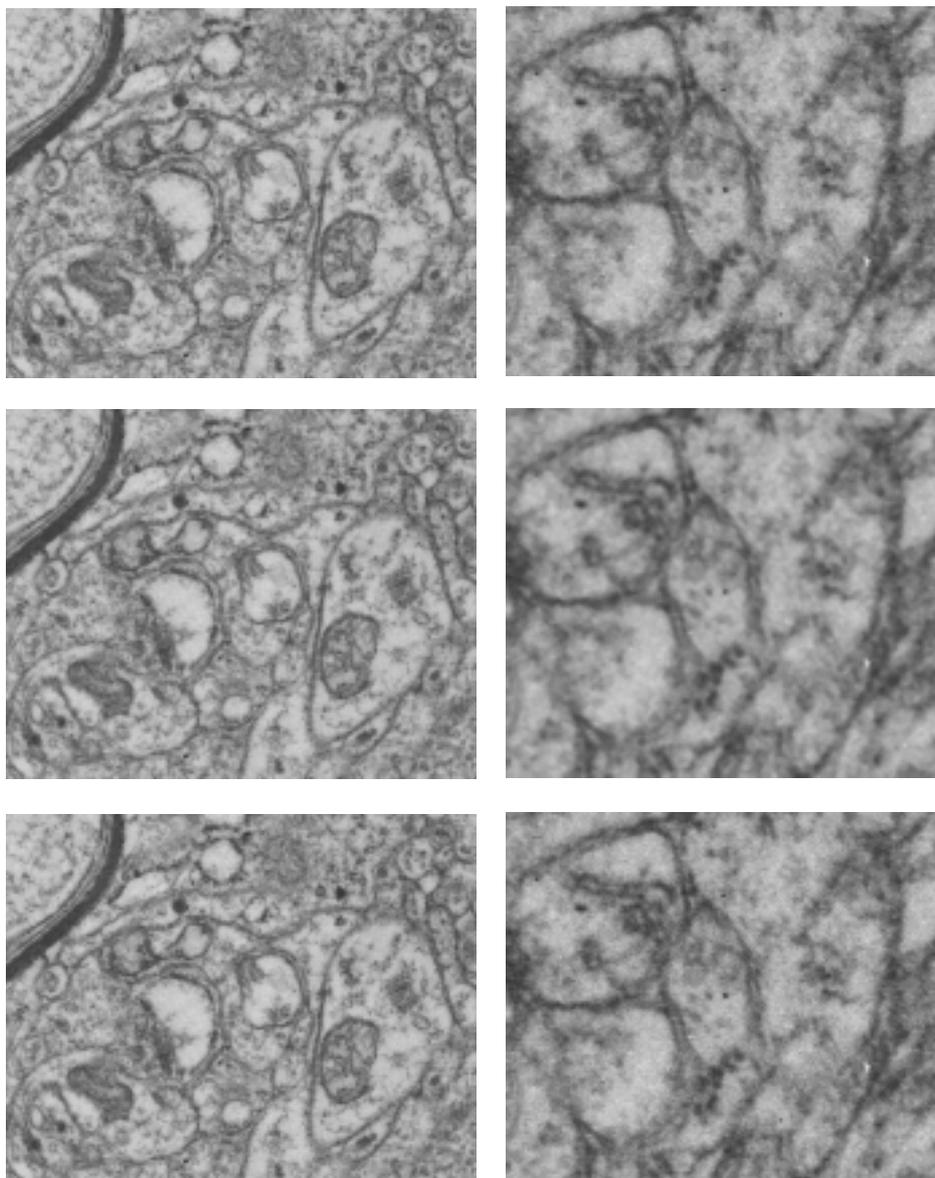


ABBILDUNG 1.3.3. Das jeweils obere Bild stellt das Originalbild dar. In der Mitte ist das mit einem 3x3 Medianfilter gefilterte Bild abgebildet, während in der unteren Bildreihe das jeweils JPEG-komprimierte Bild dargestellt wird. Das JPEG-Bild kommt visuell dem Original mit seinen verrauschten Strukturen näher. In der rechten Spalte ist ein Gewebeausschnitt in höherer Vergrößerung dargestellt.

Daten anhand der digitalisierten Konturen der Zellen zu erheben (Überblick in [Rus90]). So werden von Moss in [Mos92] erste Ansätze von computergestütztem und automatisiertem Alignment vorgestellt, mit denen die Parameter der Transformation und Rotation der einzelnen

Schnittbilder zueinander korrigiert werden. Allerdings wird hier schon auf die Schwierigkeit hingewiesen, diese Schritte des Alignments und der Rekonstruktion auf komplizierte Strukturen anzuwenden. Moss erwähnt auch, daß das Alignment per „Augenmaß“ meist ein zufriedenstellendes Ergebnis liefert. Weiterhin wird darauf hingewiesen, daß die Verformungen der Schnitte, die durch das Dünnschneiden entstehen, zu größeren Störungen der Rekonstruktion führen können.

Um diese Fehler zu kompensieren, wird in [ML92] vorgeschlagen, größere Schnittdicken beim Untersuchen des Zellmaterials zu verwenden. Dieser Verlust von Auflösung soll dann durch die Verwendung von gekippten Aufnahmen durch die Verwendung stereoskopischer Betrachtungsmethoden beim Digitalisierungsprozeß, d.h. beim Erfassen der Konturen, berücksichtigt werden.

In [BG92b] wird ein Verfahren vorgeschlagen, bei dem es gelingt, Landmarken derart im Präparat zu plazieren, daß diese für die Korrektur der Alignmentparameter, die in diesem Fall auch die Verzerrungen und Stauchungen beinhalten, herangezogen werden können. Gearbeitet wird bei dieser Methode in zwei verschiedenen Vergrößerungsstufen: Für die kleinere Vergrößerung, durch die die Visibilität der Landmarken gewährleistet ist, wird ein UV-Laser benutzt, der in den polymerisierten Block vor dem Schneiden Löcher der Größe  $2\ \mu\text{m}$  brennt. In der zweiten, höheren Auflösungsstufe wird das STEM dazu benutzt, kleinere Landmarken in dem nun begrenzteren Gesichtsfeld zu plazieren, deren Lokalisation in Bezug auf die laserinduzierten Löcher bestimmt werden kann.

Diese Art der Methode erlaubt ein sehr genaues und zuverlässiges Alignment. Sie bringt jedoch den ganz großen Nachteil mit sich, daß man schon *vor* der Aufnahme wissen muß, wo sich das Objekt der *Begierde* im Zellblock befindet. Eine Zerstörung desselben durch den Laserstrahl würde zur unweigerlichen Neudurchführung des Experimentes führen. Leider können durch die begrenzten Gesichtsfelder bei den verschiedenen Vergrößerungen die Landmarken nicht beliebig weit außen plaziert werden, da zumindest auf der Vergrößerungsstufe eines Schnittes alle Landmarken zu erkennen sein müssen.

**1.4.2. Fehlerkorrekturen im vorgestellten Rekonstruktionsprozeß.** Da die bis dato vorgestellten Methoden entweder durch eine maximale Bildgröße der erfaßbaren Datenmenge eingeschränkt sind, die Stauchungen von EM-Schnitten unberücksichtigt lassen, oder versuchen, diese durch aufwendige Methoden zu bestimmen, soll mit CAR ein Programm geschaffen werden, welches die Datengewinnung unter einer einheitlichen Benutzeroberfläche ermöglicht und die Korrektur möglichst vieler potentieller Fehlerquellen erlaubt, sei es manuell oder semiautomatisch. Ebenfalls soll in das Programm ein Rekonstruktionsalgorithmus integriert sein, der eine Oberflächendarstellung des untersuchten Objektes und somit eine Quantifizierung desselben erlaubt. Im

folgenden soll lediglich nochmals kurz aufgelistet werden, welche möglichen Fehlerquellen im Rekonstruktionsprozeß Berücksichtigung finden und welche unberücksichtigt bleiben.

Wie wir im Abschnitt 1.1 gesehen haben, weist also das EM leichte Vergrößerungsschwankungen bei der Aufnahme auf. Schwerer wiegt jedoch die Tatsache, daß sich das biologische Präparat unter dem EM erhitzen kann, wodurch ebenfalls ein scheinbarer Vergrößerungseffekt eingeführt wird. Dieser Art von Fehler kann durch Einführung eines Vergrößerungsfaktors  $S_{\text{mag}}$  in das Rekonstruktionsmodell Rechnung getragen werden.

Der bei der Präparation auftretende Fehler, der als *Schrumpfungsartefakt* bezeichnet wird, kann ebenfalls durch die Einführung eines globalen Vergrößerungsfaktors  $S_{\text{shrink}}$  berücksichtigt werden. Allerdings ist die Bestimmung dieser Größe bisher noch nicht gelungen und hängt stark von Experiment zu Experiment, also von Zellmaterial zu Zellmaterial und Einbettung zu Einbettung ab. Dieser Fehler wird daher bei der im Kapitel 4 vorgestellten Auswertung nicht berücksichtigt.

Der beim Dünnschneiden auftretende Fehler der Verformung der Schnitte (siehe Abschnitt 1.4.1) kann in begrenztem Maße im Modell berücksichtigt werden, indem wir beim Übergang von einem Schnitt zum nächsten eine allgemeine lineare Transformation, eine sog. allgemeine Skalierung (*Non-uniform Scale*)  $S_{\text{nuscale}}$  einführen (siehe auch [BG92b]). Dieser Ansatz erlaubt es, zwei Bilder derart aufeinander anzupassen, daß lineare Veränderungen zwischen den beiden Bildern (Vergrößerung, Verzerrung) berücksichtigt werden können.

Sollten starke Nichtlinearitäten auftreten, bei denen die Approximation nullter Ordnung nicht ausreichen sollte, müßten die Ebenen in Unterregionen unterteilt werden, auf denen diese nun stückweise linearen und stetigen Transformationen angewendet werden. Dieser Ansatz bleibt jedoch in dieser Arbeit unberücksichtigt.

Wie sich bei der Auswertung im Kapitel 4 zeigen wird, ist der lineare Ansatz zur Bestimmung der globalen morphologischen Parameter der *Calyx von Held* notwendig, aber auch vollauf ausreichend. Möchte man jedoch in kleinen Regionen des synaptischen Spaltes Simulationsrechnungen durchführen, müßte zuerst geprüft werden, ob man die linearen Transformationen optimieren kann, oder ob man lokal lineare Transformationen einführen muß.

Eine genaue Implementation dieser Korrekturterme und die programmtechnische und mathematische Umsetzung werden im folgenden Kapitel ausführlich erläutert und aufgelistet.

## KAPITEL 2

### Computer Aided Reconstruction mit CAR

Nachdem wir im Kapitel 1 die der Rekonstruktion vorangehenden Schritte kennengelernt und die dabei auftretenden Fehler diskutiert haben, soll in diesem Kapitel anhand des im Rahmen dieser Arbeit entwickelten Softwarepaktes zur Rekonstruktion serieller Schnitte (**CAR** - Contour Alignment Reconstruction) die Vorgehensweise zur Erfassung der Parameter, die für eine erfolgreiche 3D-Rekonstruktion erforderlich sind, illustriert werden (für einen Überblick siehe Abbildung (2.0.1)).

Ziel ist es hierbei ein Programm zu entwickeln, daß keinen Größenrestriktionen<sup>1</sup> unterliegt und die Datenerfassung der zur Rekonstruktion notwendigen Korrekturen und Transformationen unter einer einheitlichen und leicht bedienbaren Oberfläche ermöglicht. Diese Daten, sowie die digitalisierten Konturdaten sollen leicht manipulier- und in einer zentralen Datei abspeicherbar sein. Diese Datei sollte in einem plattformübergreifenden und austauschbaren Format geschrieben werden. Auch die berechneten Geometrien und Objektdarstellungen sollen plattformübergreifend austauschbar und somit auf verschiedenen Systemen darstellbar und weiterverwertbar sein.

Als erstes wollen wir den Rekonstruktionsprozeß, wie er in **CAR** realisiert wurde, schematisch vorstellen, bevor wir in den nächsten Abschnitten auf die Funktionen und Parameter im einzelnen eingehen werden. Der Rekonstruktionsprozeß selbst wird dann im Kapitel 3 vorgestellt werden.

Abbildung (2.0.1) zeigt den prinzipiellen Aufbau von **CAR**. Der Aufbau von **CAR** gibt auch gleichzeitig die einzelnen Arbeitsschritte beim Rekonstruktionsprozeß vor:

1. Im ersten Arbeitsschritt wird das Projekt festgelegt. In diesem werden die wichtigsten Parameter, wie die Vergrößerung oder die mittlere Schnittdicke eingestellt.
2. Im nächsten Arbeitsschritt, werden die einzelnen Schnittebenen erzeugt. Außerdem ist es möglich, in diesem Schritt das Alignment *ohne* Stauchungen durchzuführen.
3. Sollte man durch die gewählte Vergrößerung bedingt mehrere Aufnahmen pro Schnitt durchführen müssen, so können diese im

---

<sup>1</sup>außer die der Festplatten und des Hauptspeichers

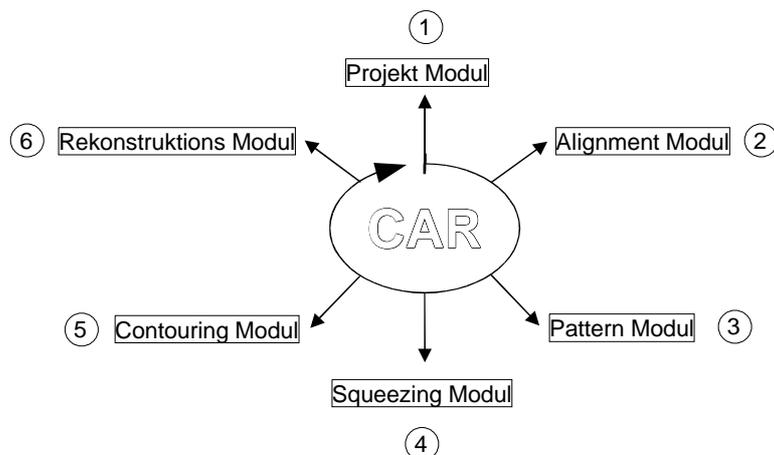


ABBILDUNG 2.0.1. Der modulare Aufbau von CAR im Überblick: Die einzelnen Module arbeiten unabhängig voneinander, werden aber im Laufe des Rekonstruktionsprozesses in Pfeilrichtung abgearbeitet (1-6).

sog. *Pattern*-Modus eingefügt werden. Hierbei wird berücksichtigt, daß die scheinbare Vergrößerung des Mikroskops zwischen den Aufnahmen variieren kann.

4. In diesem Schritt können nun auch die Stauchungen, die vor allem beim Dünnschneiden auftreten, korrigiert werden (*Squeezing*).
5. Im *Contour*-Modus können die Konturdaten manuell erfaßt werden. Bei nicht spezifisch gefärbtem Zellmaterial ist nur begrenzt an eine semiautomatische Datenerfassung denkbar.
6. Im letzten Arbeitsschritt wird aus den gewonnenen, planaren Konturdaten ein virtuelles Modell des Objektes erzeugt. Auf diesen Punkt wird ausführlich im Kapitel 3 eingegangen.

In den folgenden Abschnitten sollen die einzelnen Arbeitsschritte und die jeweils steuerbaren und bestimmbar Parameter dargestellt werden. So wird anhand einer baumartigen Darstellung<sup>2</sup>, der Aufbau des jeweiligen Parameterfeldes schematisch dargestellt. Die einzelnen Parameterknoten stellen i.a. lineare Transformationen dar, die — je nach Arbeitsschritt — zur Korrektur eines bestimmten Fehlers, der im vorigen Kapitel diskutiert wurde (siehe Seite 19ff), dient. Diese linearen Transformationen wollen wir, wie in der Computergraphik üblich, in sog. *homogenisierten Koordinaten* darstellen<sup>3</sup>.

<sup>2</sup>wie im Anhang in Kapitel A näher erläutert wird

<sup>3</sup>für eine kurze Einführung siehe [Wat00]

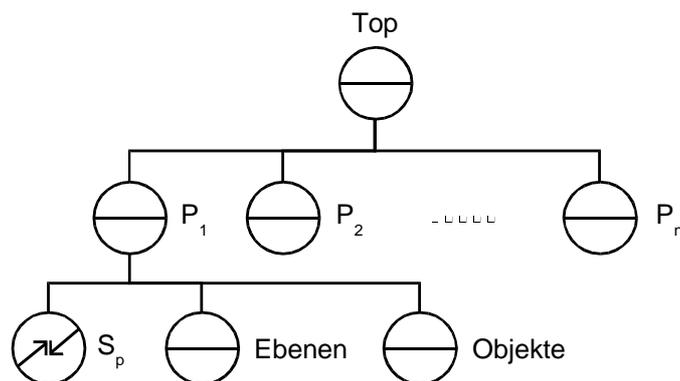


ABBILDUNG 2.1.1. Unterhalb des *Top*-Knotens, können mehrere Projekte  $P_1$  bis  $P_n$  verwaltet werden. Für jedes einzelne Projekt werden in  $\mathbf{S}_p$  die Projektparameter, wie Vergrößerung und  $z$ -Schnittdicke abgelegt.

Wir betrachten hierzu speziell die linearen Transformationen der Rotation in der Ebene, der Skalierung und der Translation. Diese werden im weiteren wie folgt abgekürzt:

$$(2.0.1) \quad \mathbf{R} := \begin{pmatrix} \cos \alpha & \sin \alpha & 0 & 0 \\ -\sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(2.0.2) \quad \mathbf{S} := \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(2.0.3) \quad \mathbf{T} := \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

## 2.1. Das Projekt

Im Hauptmodus des Programms, dem sog. *Projekt*-Modus, werden die für ein Projekt wichtigen Parameter eingestellt und in der Datenstruktur abgelegt, wie es in Abbildung (2.1.1) angegeben ist. Eine Erläuterung der einzelnen Knotensymbole findet der Leser in Anhang A. Die Parameter sind im einzelnen:

1. Der Projektname
2. Der Projektpfad
3. Die Vergrößerung
4. Die  $z$ -Schnittdicke
5. Die Projekteinheit

Die für die Rekonstruktion wichtigsten Parameter sind offensichtlich die Vergrößerung, mit der der Datensatz aufgenommen wird und die durchschnittliche  $z$ -Dicke der einzelnen Schnitte. Wie wir im Abschnitt 1.2.2 gesehen haben, wird die tatsächlich erzielte Schnittdicke durchaus von der am Mikrotom eingestellten abweichen. Eine Schätzung der tatsächlichen Schnittdicke erlaubt dabei ein Blick in die Tabelle 1.2.2. Daher ist in CAR vorgesehen, im Projekt-Modus lediglich eine mittlere Schnittdicke anzugeben. Diese wird in Einheiten der sog. *Projekteinheit* erfaßt, welche außerdem die Grundeinheit für die Ausgabe der berechneten Parameter wie Volumen und Oberfläche bildet (z.B. in Nanometer oder Mikrometer).

Die Transformationsmatrix, die im wesentlichen eine Skalierung darstellt, erlaubt es, daß die Konturdaten in Einheiten des Bildraumes erfaßt werden. Erst nach der Anwendung der Projektskalierung  $\mathbf{S}_p$ , werden die Koordinaten in den Objektraum der Zelle transformiert. Die Skalierungsmatrix sieht im wesentlichen wie folgt aus, wenn  $s_z$  die  $z$ -Schnittdicke und  $m$  die Vergrößerung bezeichnet:

$$(2.1.1) \quad \mathbf{S}_p := \begin{pmatrix} \frac{1}{m} & 0 & 0 & 0 \\ 0 & \frac{1}{m} & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Wie die tatsächliche Schnittdicke für jeden einzelnen Schnitt eingestellt werden kann, sehen wir im nächsten Abschnitt beim sog. Alignment-Modus.

Die Parameter 1 und 2 in der obigen Auflistung haben mehr Bedeutung bei der Anwendung von CAR in der Praxis und sollen hier der Vollständigkeit halber erklärt werden:

Da, wie in Schema (2.1.1) zu sehen ist, das Datenformat zur Ablage mehrerer sog. Projekte konzipiert ist, gibt der Projektnamen lediglich einen Bezeichner wieder, der das Projekt eindeutig identifiziert. Der Projektpfad gibt den Datenpfad an<sup>4</sup>, unter dem die abgelegte Bildinformation zu finden ist. Zusammen mit der Tatsache, daß in der CAR-Datei lediglich Verweise (*Links*) auf die benutzten Bilddaten verwendet werden, ergibt sich ein in der Praxis nicht zu vernachlässigender Vorteil der Verschiebbarkeit und Austauschbarkeit der Bilddaten: So können die Bilddaten ohne weiteres auf CD gebrannt oder auf der Festplatte verschoben werden, ohne daß die einzelnen *Links* auf die Bilddaten aktualisiert werden müßten. Es muß lediglich der *Projektpfad*, wie er im Projekt-Modus spezifiziert wurde, geändert werden. Außerdem ist durch die Tatsache, daß von den Bilddaten, sind sie einmal eingelesen

---

<sup>4</sup>dieser kann bei Erweiterung des OpenInventor-Datenformates auf VRML auch eine beliebige Lokation im Internet sein

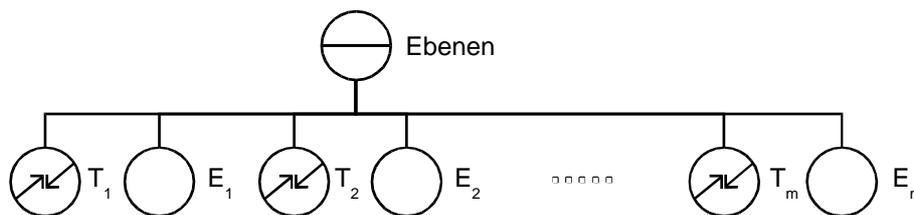


ABBILDUNG 2.2.1. Unter dem Hauptebenen-Knoten sind die einzelnen Schnittebenen angeordnet ( $E_1$  bis  $E_n$ ). Diesen geht jeweils die Transformation der Form  $\mathbf{T}_a$  voraus, die das Alignment der Ebene zur vorhergehenden Ebene beschreibt. Diese Transformationen kumulieren sich entlang des *Renderpfades* von oben nach unten, bzw. von links nach rechts.

(siehe Abschnitte 2.2 und 2.3), die Bildgröße in absoluten Koordinaten abgespeichert wird, das Verwenden von Bildern in unterschiedlichen Auflösungs- oder Kontrastierungsstufen möglich. Dadurch ist das Programm für Arbeiten auf Bildpyramiden vorbereitet, was eine erhebliche Arbeitserleichterung und -beschleunigung bei sehr großen Datenmengen, wie im Beispiel der *Calyx von Held*, bewirken kann.

## 2.2. Das Alignment

Im Alignment-Modus werden die einzelnen Schnittebenen, die jeweils einem Ultradünnschnitt entsprechen, aufeinandergestapelt und nach Schema (2.2.1) in die CAR-Datei einsortiert. Maßgebliche Parameter sind hier:

1. Der Ebenenname
2. Die Schrittweite der Ebene in  $z$ -Richtung
3. Die Translation  $\mathbf{T}$
4. Die Rotation  $\mathbf{R}$

Der Ebenenname ist — wie der Projektname — hauptsächlich in der praktischen Anwendung von Nutzen. Hierdurch ist es möglich, die in der Datei abgelegten Schnittebenen über den Ebenennamen mit einer eindeutigen Bezeichnung und Lokation der Ultradünnschnitte in Verbindung zu bringen, z.B. mit der Gridboxnummer des jeweilig mikroskopierten Grids.

Die Parameter 2-4 spielen für die Rekonstruktion eine wichtige Rolle. Sie dienen im wesentlichen der Registrierung (engl. *Alignment*) der Bilddaten. Das Alignment wird hier ohne Korrekturen, die der Verformung der Schnitte durch das Ultramikrotom Rechnung tragen, durchgeführt. Es werden also lediglich Translation  $\mathbf{T}$  und Rotation  $\mathbf{R}$  für jede Ebene ermittelt.

Wie wir in Abbildung (2.2.1) sehen können, ist die interne Datenstruktur der CAR-Datei so angelegt, daß  $\mathbf{T}$  und  $\mathbf{R}$  die Relativorientierung der Ebene  $E_n$  zu der Ebene  $E_{n-1}$  wiedergibt. Dies wird durch die Verwendung eines sog. *Group-Knotens* im Inventordatenformat sichergestellt (siehe Anhang A für nähere Informationen), der eine Kumulierung der linearen Transformation entlang des Renderpfades, wie er in Abbildung (2.2.1) angegeben ist, erlaubt.

Dieses Prinzip der *relativen Orientierung* der einen Ebene zur vorhergehenden ist ein Grundprinzip in CAR: Dieses Vorgehen spiegelt die Tatsache wider, daß aufeinanderfolgende Schnitte als zueinander *ähnlich*, auch im mathematischen Sinne, betrachtet werden können. Die Ähnlichkeit des Schnittes  $E_n$  zu den Folgeschnitten  $E_{n-k}$  geht mit wachsendem  $k$ , was einer ansteigenden Schnittdicke entspricht, verloren. Daher ist eine starke Gewichtung der Relativorientierung von  $E_n$  zu  $E_{n-1}$  evident.

Weiterhin hat dieses Prinzip einen hohen praktischen Nutzen: Müssen Nachkorrekturen am Datensatz vorgenommen werden, so reicht es aus, die noch schlecht registrierten Schnitte zu verändern, während die anderen Relativorientierungen von diesen Änderungen unberührt bleiben. Dies erspart bei großen Datensätzen viel Zeit und Arbeit.

Hat man die genaue Schnittdicke exakt bestimmt, so kann man die Anpassung der  $z$ -Schrittweite von einem Schnitt zum nächsten in Vielfachen der im Projekt-Modus spezifizierten mittleren Schnittdicke angeben. Sollte also ein Schnitt bei der Präparation verloren gegangen sein, so wird die relative Schnittdicke für den betroffenen Schnitt auf  $z = 2$  gesetzt. Mit dieser Abweichung vom Wert 1 ist es also möglich, Variationen der Schnittdicke, wie sie beim Ultradünnschneiden vorkommen, zu kompensieren.

Das Alignment umfaßt also im wesentlichen die freien Parameter  $\alpha, a_x, a_y, a_z = z$ , wodurch die lineare Transformation für das Alignment die folgende Gestalt annimmt:

$$(2.2.1) \quad \mathbf{T}_a := \begin{pmatrix} \cos \alpha & \sin \alpha & 0 & a_x \\ -\sin \alpha & \cos \alpha & 0 & a_y \\ 0 & 0 & 1 & a_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

### 2.3. Das Patterning

Sollte es nicht möglich sein, die gewünschte Auflösung des Digitalisierungsprozesses mit der Aufnahmevergrößerung derart abzustimmen, daß eine Aufnahme pro Ultradünnschnitt ausreicht, so ermöglicht es der Pattern-Modus, mehr als ein Bild in eine Schnittebene zu übernehmen. Diese Bilder werden dann in eine bestehende Ebene derart integriert, daß das Gesamtbild des Ultradünnschnittes durch die Überlagerung der Einzelaufnahmen entsteht.

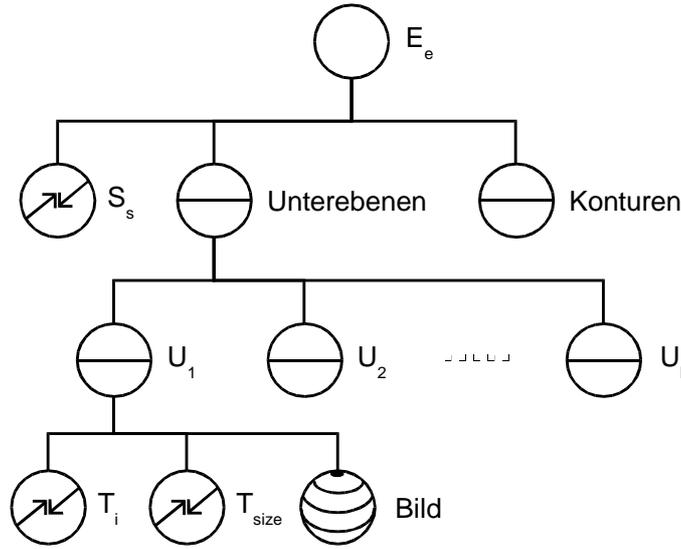


ABBILDUNG 2.3.1. Jede Ebene beherbergt einen Unterebenenknoten, der die Bilddaten aufnimmt, sowie einen Konturknoten, der die Umrandungen der Objekte, d.h. die Konturen verwaltet. Beiden Knoten ist die Transformation  $\mathbf{S}_s$  vorausgeschaltet, die die Verzerrungen in der Ebene korrigiert. Jeder Unterebenenknoten kann mehrere Bildknoten verwalten ( $U_1$  bis  $U_l$ ), in denen jeweils die Bildgröße  $\mathbf{T}_{\text{size}}$  als auch die Bildtransformationen  $\mathbf{T}_i$  abgelegt sind.

Es ist wichtig, bei den Aufnahmen darauf zu achten, daß die Einzelbilder mit Überlappung aufgenommen werden. Nur so ist man in der Lage, die Einzelbilder erfolgreich zu einem Gesamtbild zusammenzusetzen (siehe Abbildung (2.3.2)). Prinzipiell gilt: Je größer die Überlappung, je genauer ist die exakte relative Lage der Einzelbilder zueinander zu bestimmen. Wie wir der schematischen Darstellung in 2.3.1 entnehmen können, werden die Transformationen isoliert und gezielt auf die Einzelbilder angewandt, da diese unterhalb eines speziellen *Group*-Knotens, dem sog. *Separator* (Trenner) organisiert sind. Die Transformation des Einzelbildes mit dem fortlaufenden Index  $i$  hat demnach die Gestalt:

$$(2.3.1) \quad \mathbf{T}_i := \begin{Bmatrix} m_i \cos \alpha_i & \sin \alpha_i & 0 & i_x \\ -\sin \alpha_i & m_i \cos \alpha_i & 0 & i_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{Bmatrix}$$

Wir haben in dieser linearen Transformation die Variable  $m_i$  eingeführt. Sie soll der Tatsache Rechnung tragen, daß das EM bei der Aufnahme leichten Vergrößerungsschwankungen unterliegt. Dies gilt insbesondere bei Aufnahmen von biologischem Material, welches sich



ABBILDUNG 2.3.2. Links sind zwei Bilder dargestellt, bei denen der Überlappungsbereich ungenügend groß erscheint, um ein gutes Patterning vornehmen zu können. Rechts hingegen ist der Überlappungsbereich ausreichend groß.

bei der Aufnahme erhitzt und sich dadurch ausdehnen kann. Besonders auffällig ist es, daß v.a. Sekundäraufnahmen, die einen gewissen Zeitraum nach den Primäraufnahmen gemacht wurden, Vergrößerungsänderungen aufweisen. Nur durch die Einführung des Skalierungs- oder Vergrößerungsparameters  $m_i$  kann es gelingen, diese Aufnahmen in ausreichender Genauigkeit in das bestehende Mosaik einzufügen.

## 2.4. Das Squeezing

Dieser Modus wurde in CAR integriert, um die unterschiedlichen Deformationen, die das Gewebe beim Dünnschneiden durch das Ultramikrotom erfährt, rückgängig zu machen und zu korrigieren, was durch Applikation von Xyloidampf (siehe [Rei67]) nur unzureichend bewerkstelligt werden kann<sup>5</sup>. Es ist auf jeden Fall nicht zu erwarten, daß alle Schnitte gleichstarke Verformungen erfahren. Daher wird versucht, der ungleichmäßigen Verformung des Schnittes durch die Applikation einer Skalierungsmatrix Rechnung zu tragen. Diese Matrix hat die Gestalt:

$$(2.4.1) \quad \mathbf{S}_s := \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Auch hier werden die Skalierungsfaktoren  $s_x$  und  $s_y$  relativ von einer Ebene  $E_n$  zu vorhergehenden Ebene  $E_{n-1}$  justiert, was die schon im Abschnitt 2.2 auf Seite 28 angesprochenen praktischen Vorteile mit sich bringt. Man kann sich also auch hier auf die Relativtransformation von der Ebene  $E_n$  zu vorhergehenden Ebene  $E_{n-1}$  konzentrieren, was eine Nachkorrektur der Transformationsparameter erleichtert.

<sup>5</sup>es scheint sogar so zu sein, daß sich die Schnitte nach dem Bedampfen übernatürlich stark ausdehnen können

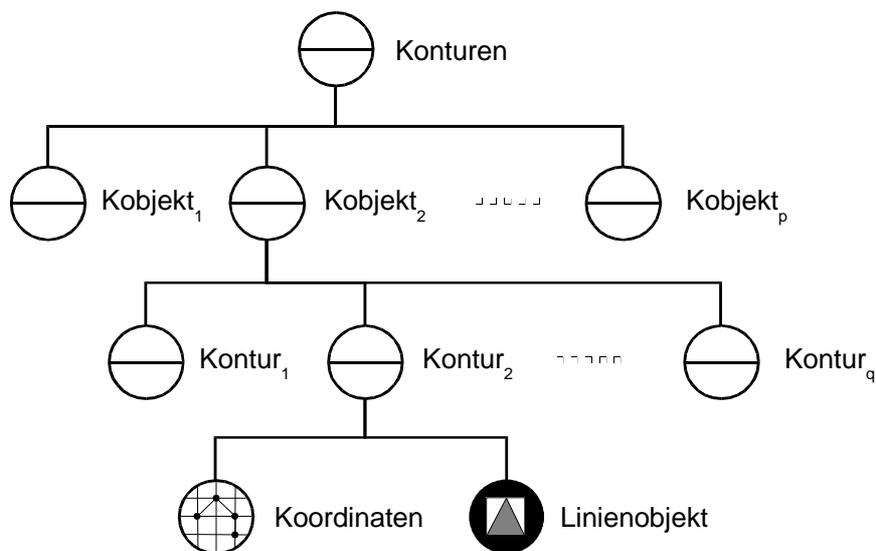


ABBILDUNG 2.5.1. Unterhalb des Konturen-Knotens werden die einzelnen Umfahrungen den jeweiligen Konturobjekten zugeordnet. Die Konturkoordinaten sind in Ebenenkoordinaten abgespeichert.

## 2.5. Das Contouring

Im *Kontur*-Modus werden nun diejenigen Strukturen auf den Bildern umfahren, welche für die 3D-Rekonstruktion verwendet werden. Dabei wird in *CAR* der Begriff des *Konturobjektes* eingeführt: Jedes Konturobjekt entspricht später in der Rekonstruktion einem Objekt. Es spielt hierfür keine Rolle, daß dieses rekonstruierte Objekt durch die Automatismen im Rekonstruktionsalgorithmus (siehe Kapitel 3) in mehrere, räumlich getrennte Subobjekte zerfallen kann.

Ein Konturobjekt ist also eine logische Einheit, unter der die schichtweise erfaßten Objektdaten gesammelt werden. Als freie Parameter für ein Konturobjekt stehen in *CAR* die folgenden zur Verfügung:

1. Der Konturobjektname
2. Der Konturtyp
3. Die Konturfarbe
4. Die Konturliniendicke

Die Punkte 3 und 4 spiegeln lediglich Parameter wieder, die für die Darstellung der Konturen beim Generieren und Editieren von Bedeutung sind. Die Konturfarbe gibt im Rekonstruktionsmodul von *CAR* gleichzeitig die Objektfarbe<sup>6</sup> an.

Der Konturname ist ein eindeutiges Label, unter welchem die erfaßten Konturdaten referenziert werden. Die Konturdaten selbst sind im

<sup>6</sup>speziell: den ambienten Farbanteil

weitesten Sinne Einzelpunkte, die durch Anklicken auf den Ebenenbildern generiert werden. Die Konturdaten werden also pro Ebene erfasst, wobei die Werte im jeweils lokalen Ebenenkoordinatensystem gespeichert werden. Wie wir den Schemata (2.3.1) und (2.5.1) entnehmen können, wirken daher die Alignment- und Squeezingtransformationen ebenfalls auf die Konturdaten. Werden *nach* dem Erfassen der Daten noch die Alignment- oder Squeezingparameter geändert, dann passen sich dementsprechend auch die Konturen den neuen Parametern an: Somit sind die Konturpunkte starr mit der jeweiligen Ebene und den darin angeordneten Bildern gekoppelt, innerhalb derer sie erfasst wurden.

CAR unterscheidet nun zwischen verschiedenen Typen von Konturpunkten:

1. geschlossenen Konturen
2. Linienzügen
3. ungeordnete Punktmenge

Geschlossene Konturen sind hierbei Polygonzüge, bei denen der Anfangs- und Endpunkt übereinstimmt. Sie umranden ein bestimmtes Gebiet in der Ebene, was in unserem Falle einem Schnitt durch das zu rekonstruierende Objekt entspricht. Dementsprechend sollten sich die Konturen nicht selbst durchdringen oder mit anderen Objektkonturen kreuzen. Die Konturen dürfen hingegen durchaus ineinander verschachtelt sein, was im Rekonstruktionsalgorithmus topologisch richtig berücksichtigt wird (siehe auch die Abschnitte 3.3.1 und 3.3.5).

Linienzüge sind dagegen Polygonzüge, bei denen Anfangs- und Endpunkt nicht übereinstimmen müssen. Sie werden in CAR hauptsächlich dazu benutzt, um Distanzen von zwischen 2 Punkten, Längen oder Durchmesser von quergeschnittenen Strukturen zu bestimmen.

Bei Punkt 3 in obiger Liste handelt es sich um eine ungeordnete Menge von 3D Meßpunkten, die zur Berechnung von Punktverteilungen oder als Referenzpunkte für Abstandsmessungen zu (später) rekonstruierten Objektoberflächen dienen können. Weiterhin kann dieser Konturtyp als Marker für ein automatisches Alignment fungieren (für eine Diskussion des automatischen Alignments siehe Seite 115ff).

## 2.6. Die Rekonstruktion

Im *Reconstruction*-Modus können dann aus den erfaßten Punktdaten Objekte generiert werden, die anschließend für weitere morphologische Auswertungen herangezogen werden können. Selbstredend unterscheiden sich die Rekonstruktionsformen auch vom angegebenen Konturtyp: Bei unsortierten Punkten können Darstellungen mit Kugeln eines wählbaren Radius, bei Linien Flächen und bei geschlossenen Konturen geschlossene Oberflächen generiert werden.

Wie wir im Kapitel 3 sehen werden, gibt es eine fülle von verschiedenen Ansätzen, die das Problem der Oberflächengenerierung ausgehend von Konturdaten lösen wollen. Je nach Eingabedaten sind aber nur sehr bestimmte Rekonstruktionsalgorithmen für die Auswertung geeignet. CAR bedient sich im wesentlichen des Rekonstruktionsansatzes von Boissonat ([Boi88]) und dessen Erweiterung auf topologisch komplizierte Strukturen von Boissonat und Geiger ([BG92a]). Dieser Ansatz ist nach Meinung des Autors der vielversprechendste Ansatz, der auch in komplizierten Fällen zu zumindest guten Startrekonstruktionen führt, die dann in einem nächsten Schritt verfeinert und verbessert werden können. Größter Vorteil dieser Vorgehensweise ist jedoch die Tatsache, daß das Objekt im  $\mathbb{R}^3$  in Tetraeder unterteilt und somit eine Tessellierung desselben vorliegt, und nicht nur die Oberfläche des Objektes selbst darstellbar und auswertbar ist (Näheres hierzu im Abschnitt 3.3).

Abgesehen vom Boissonat'schen Ansatz implementiert CAR außerdem noch 3 weitere, schnelle Übersichtsrekonstruktionen, die für bestimmte, eingeschränkte Fälle auch eine gültige Objektrepräsentation inkl. relevanter Auswertedaten liefern.

- Da gibt es zum einen die *Outline*-Rekonstruktion, welche die in Schritt 2.5 erfaßten Konturdaten *as is* darstellt. Die hier gewonnenen Daten geben den Umfang der Umfahrungen wieder und sind für Einfache Oberflächenabschätzungen verwendbar.
- Zum anderen gibt es die sog. *Faces*-Rekonstruktion, bei der die von den Konturen umfahrenden Regionen in triangulierter Form dargestellt werden. Dabei dürfen die Konturen — wie beim Boissonat'schen Rekonstruktionsansatz — beliebig verschachtelt sein. Als morphologischen Parameter kann man aus dieser Rekonstruktion die von den Konturen eingeschlossene Fläche bestimmen.
- Im *Brick*-Rekonstruktionsmodus werden die in der *Faces*-Rekonstruktion generierten Flächen mit einer Höhe, die sich durch den Abstand der Ebenen untereinander ergibt, versehen. Dadurch entsteht ein Volumen, daß bei Objekten, die groß gegenüber der Schnittdicke sind, schon eine erstklassige Abschätzung für das tatsächliche Volumen liefert.

Diese Liste der möglichen Rekonstruktionen soll in einer Weiterführung dieser Arbeit ausgeweitet werden, damit ein Vergleich zwischen den einzelnen Methoden, wie sie im Abschnitt 3.1 kurz vorgestellt werden, möglich wird, und damit eine flexible Auswahl für die jeweilig zu erledigende Rekonstruktionsaufgabe zur Verfügung steht. Es ist bisher nicht zu erwarten, daß ein und derselbe Rekonstruktionsansatz in allen denkbaren Fällen zufriedenstellende Ergebnisse liefern wird.

## 2.7. Eine Kurzeinführung in CAR

Im bisherigen Teil dieses Kapitels wurden grundsätzlich die Parameter diskutiert, die in CAR justierbar sind und die zur Korrektur auftretender Artefakte beim Aufnahme-prozeß (siehe Kapitel 1) dienen sollen. Über das *Wie* wurden hierbei noch keine Aussagen gemacht. Diese Lücke sollen die Ausführungen in diesem Abschnitt füllen. Es soll nicht nur die Bedienung von CAR in kurzen Zügen erläutert werden, sondern auch ein Einblick in die technische Umsetzung, die für die Anwendbarkeit von CAR von großer Bedeutung sind, gegeben werden.

**2.7.1. Hardwarevoraussetzungen.** Entwickelt wurde CAR auf einer *Silicon Graphics Reality Engine II*, einer Graphikworkstation der High-End Klasse, die zur Entstehungszeit der Entwicklung von CAR absolut für die Umsetzung der Grundkonzepte erforderlich war. In der heutzutage überaus schnellebigen Computerbranche, haben wir es dem wachsenden Bedarf im Bereich Multimedia und Spiele zu verdanken, daß mittlerweile Standard-PC's mit z.T. serienmäßig ausgelieferten 3D-Graphikkarten für die Aufgabenstellungen von CAR ausreichend scheinen (einen Überblick über aktuelle Hardware und Techniken findet der Leser in Anhang B).

Zentrale Voraussetzung für die Anwendbarkeit von CAR für das Alignment, Pattering und Squeezing ist es, daß die Bilddaten möglichst größenunabhängig in Echtzeit<sup>7</sup> transparent dargestellt, translatiert, vergrößert und rotiert werden können.

Sind Translationen und Skalierungen mit einigem Aufwand und einigen Tricks per Software auf Echtzeit trimmbar, so ist dies spätestens bei Rotationen nicht mehr möglich. Die Bilder müssen bei jedem Drehschritt aufwendig neu gesampelt oder gerastert werden, bevor sie auf den Bildschirm gezaubert werden können.

Die speziellen Graphikkarten von Silicon Graphics bieten aber dafür eine Lösungsmöglichkeit: Sie verwenden spezielle Prozessoren für die Ausführung der Transformationsberechnungen und die Rasterung des Bildes. Außerdem, und das ist der wichtigste Punkt für CAR, stellten nur die High-End Graphikkarten zu dieser Zeit speziellen Texturspeicher zur Verfügung, der über einen schnellen Bus an das Rasterboard angebunden war.

Durch den massiven Einsatz des Texturemappings und ausgefeilter Pyramidentechniken ist es in CAR gelungen, auch große Bildmengen in Echtzeit transparent übereinanderzulegen. So können alle Transformationen zum Alignment und das Pattering der Ebenen in Echtzeit vom Benutzer interaktiv durchgeführt werden.

Interaktives Eingreifen und Bedienen ist in CAR ganz groß geschrieben, da bei dem vorliegenden Bildmaterial aus der EM nicht daran zu denken ist, daß alle Arbeitsschritte von Beginn an vollautomatisch

---

<sup>7</sup>Bildraten von > 3 Bilder/Sekunde

durchzuführen sind. Dazu müßte man das Problem der Suche eines globalen Minimums in einem möglichst geeigneten Abstandsfunktional zweier Ebenen lösen, wobei darauf zu achten wäre, daß man nicht ungewollt eines der vielen lokalen Minima auswählt, daß vom Optimum weit entfernt sein kann. Einfacher gesagt: Man müßte das Matchingproblem für jedwedes Bildmaterial und für vollständige (Alignment) oder unvollständige (Patterning) Überlappung global lösen. Dies ist Gegenstand der aktuellen Forschung (siehe [MV98]).

Es erscheint hier sinnvoller, per Interaktion die Bilder vorzupositionieren und zu transformieren, um dann mit dieser ersten Schätzung an das Lösen des Matchingproblems heranzugehen (siehe Diskussion im Ausblick, Seite 115ff).

Da also die Benutzung von CAR viel Interaktion enthält, soll diesem Bereich im nächsten Abschnitt etwas Platz gewidmet werden, um dem Leser einen Kurzüberblick über den Aufbau und das Userinterface von CAR geben zu können. Eine ausführliche Beschreibung von CAR würde jedoch den Rahmen dieser Arbeit sprengen, so daß hierfür eine eigene Veröffentlichung in der Preprint-Reihe des IWR unter dem Titel *CAR - a tool for 3D reconstruction* geplant ist.

**2.7.2. Das Userinterface von CAR.** Abbildung (2.7.1) zeigt einen Blick auf das Hauptfenster von CAR. Dieses gliedert sich vornehmlich in 3 Teile:

- Im sog. Viewerfenster werden die Bilddaten dargestellt und die Parameter anhand sog. Manipulatoren eingestellt. Im Rekonstruktionsmodus kann man im Viewer dreidimensional durch den Objektraum navigieren (siehe Abbildung 2.7.1).
- In der *Display*-Liste werden die darzustellenden Bildebenen selektiert. Die Auswahl mehrerer Ebenen bewirkt die transparente Überlagerung der betreffenden Bildebenen (siehe Abbildung 2.7.3).
- In der *Pick/Edit*-Liste werden:
  - entweder diejenigen Ebenen ausgewählt, die überlagert werden sollen (Alignment-Modus — siehe Abbildung 2.7.3)
  - oder dasjenige Bild selektiert, daß in die betreffende Ebene eingefügt werden soll (Pattern-Modus — siehe Abbildung 2.7.4)
  - oder dasjenige Konturobjekt ausgewählt, zu der die anzufertigende Kontur gehören soll (siehe Abbildung 2.7.5)

Das zentrale Herzstück von CAR, welches den modularen Aufbau wie in Abbildung (2.0.1) widerspiegelt, ist das sog. Mode-Menü (siehe Abbildung (2.7.2)). Es dient zum Umschalten zwischen den verschiedenen Modulen oder Modi. Das Userinterface und somit das Erscheinungsbild von CAR bleibt jedoch für die verschiedenen Modi unverändert,

während sich lediglich die Bedeutung der Listeneinträge und auch die Einträge selbst nach dem ausgewählten Modus ändern.

In den Abbildungen (2.7.1) bis (2.7.6), die diesem Abschnitt als Bildergalerie angefügt sind, werden kurze Erläuterungen zu den einzelnen Funktionen in CAR gegeben. Die Bilder geben dabei einen Eindruck über die Funktionalität von CAR.

**2.7.3. Das Benutzerhandbuch.** Das Benutzerhandbuch zu CAR wird in der Preprint-Reihe des IWR unter dem Titel *CAR - a tool for 3D reconstruction* veröffentlicht werden. Weitere Informationen zu CAR erhält man bei Anfrage unter *kurt.saezler@iwr.uni-heidelberg.de*.

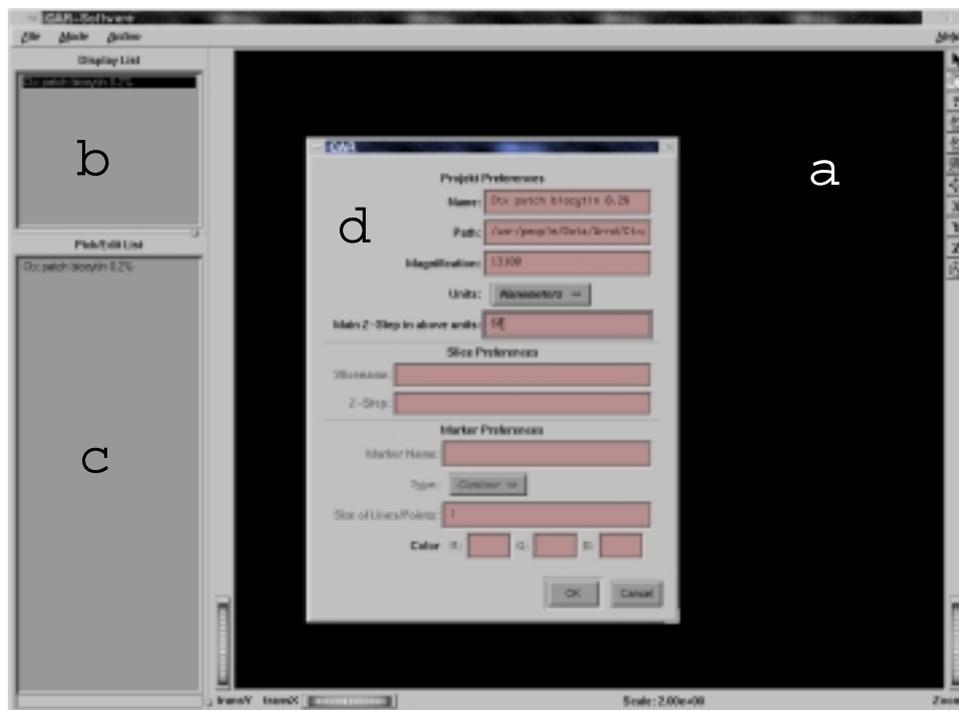


ABBILDUNG 2.7.1. Das Hauptfenster von CAR. Es unterteilt sich in drei Hauptbereiche: (a) das Viewerfenster, in welchem die Bilder, die Manipulatoren und auch die Rekonstruktionen visualisiert werden; (b) die *Display-List*, in welcher die darzustellenden Bilder ausgewählt werden; (c) die *Pick/Edit-List*, in der Ebenen für die Manipulation selektiert werden können. In (d) ist eine Dialogbox gezeigt, in der für die verschiedenen Modi zusätzlich Parameter (wie hier abgebildet u.a. Vergrößerung und Schnittdicke) spezifiziert werden können.

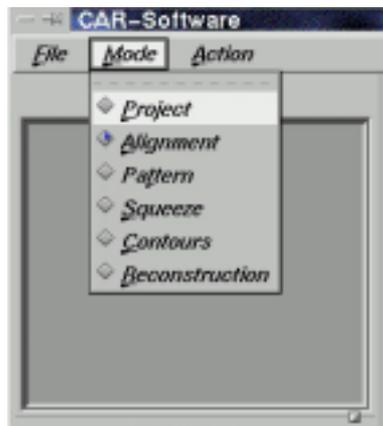


ABBILDUNG 2.7.2. Die zentrale Umschaltstelle von CAR: das Mode-Menü. Hier kann zwischen den einzelnen Modulen in CAR umgeschaltet werden.

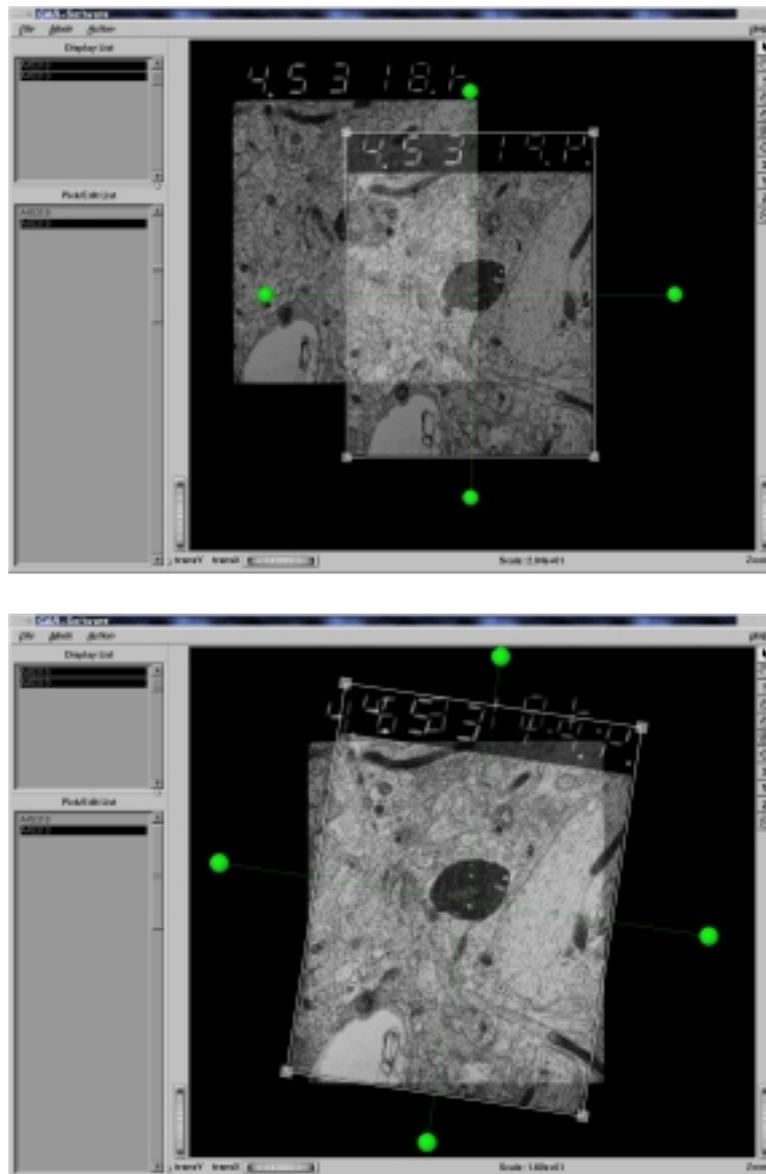


ABBILDUNG 2.7.3. Hier ist der Prozeß des Alignments kurz dargestellt: In der oberen Abbildung sind die beiden Bilder aus den aufeinanderfolgenden Ebenen A45318 und A45319 transparent und überlappend dargestellt. Die Übereinstimmung zwischen den Strukturen in den verschiedenen Ebenen ist noch sehr schlecht. Durch Bedienung des grün dargestellten Manipulators, kann man nun die Translation und Rotation so justieren, daß die Übereinstimmung sehr gut wird: Die transparent überlagerten Bildern sehen nun so aus, als wenn diese von einem Gewebe mit doppelt so dicken Ultradünnschnitten aufgenommen worden wären (siehe unteres Bild).

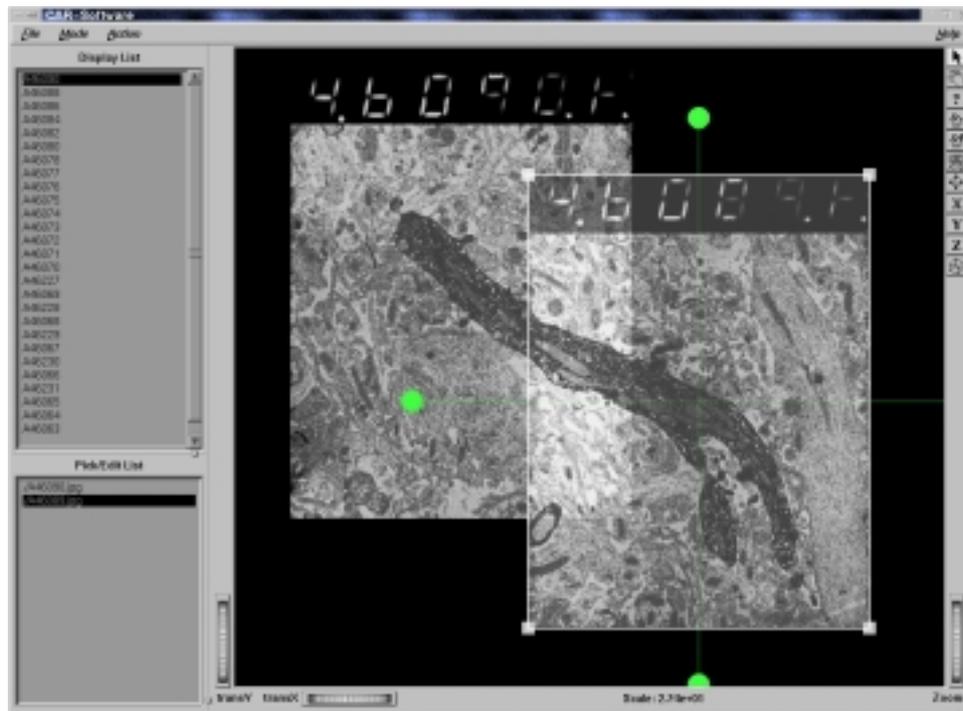


ABBILDUNG 2.7.4. Hier ist dargestellt, wie die beiden Bilder A46089 und A46090 transparent überlagert und durch Bedienung des grün angezeigten Manipulators zu einem großen Gesamtbild zusammengesetzt werden (Pattern-Modus).

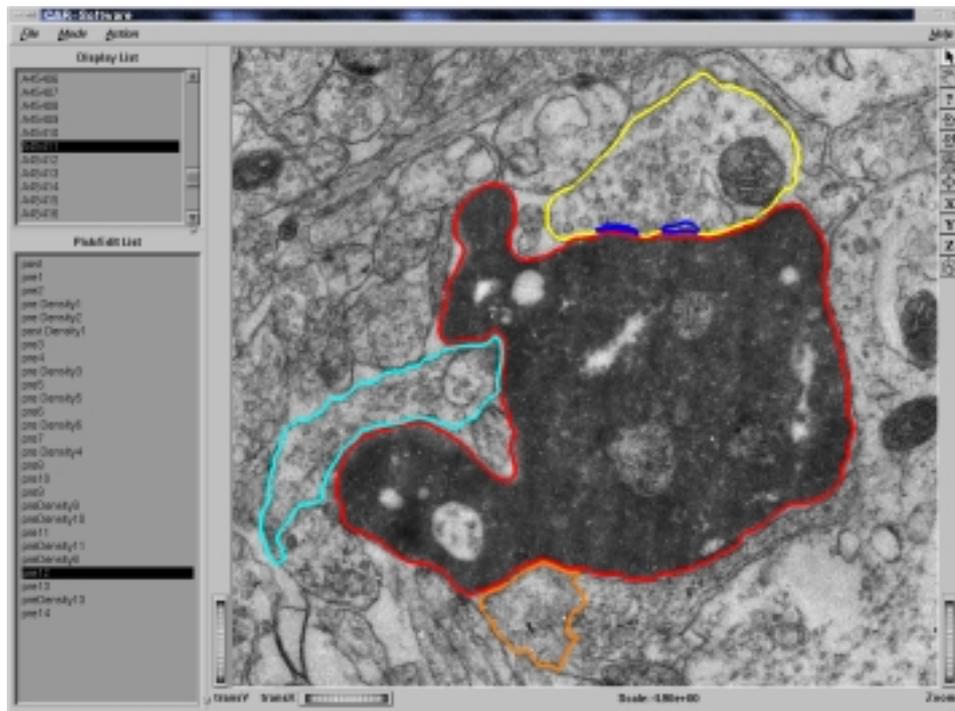


ABBILDUNG 2.7.5. Es ist zu sehen, wie verschiedene Bereiche in den Bildern umrandet sind. Diese verschiedenen Bereiche markieren die äußere Berandung von verschiedenen, biologisch relevanten Strukturen. In diesem Fall sind dies verschiedene präsynaptische Kontakte, die mit dem rot umrandeten Axon Kontakt machen. Die Daten wurden von Arnd Roth mit Hilfe von CAR erfasst. Diese Arbeit wurde ebenfalls am Max-Planck-Institut für medizinische Forschung in Zusammenarbeit mit dem Anatomischen Institut in Freiburg durchgeführt.

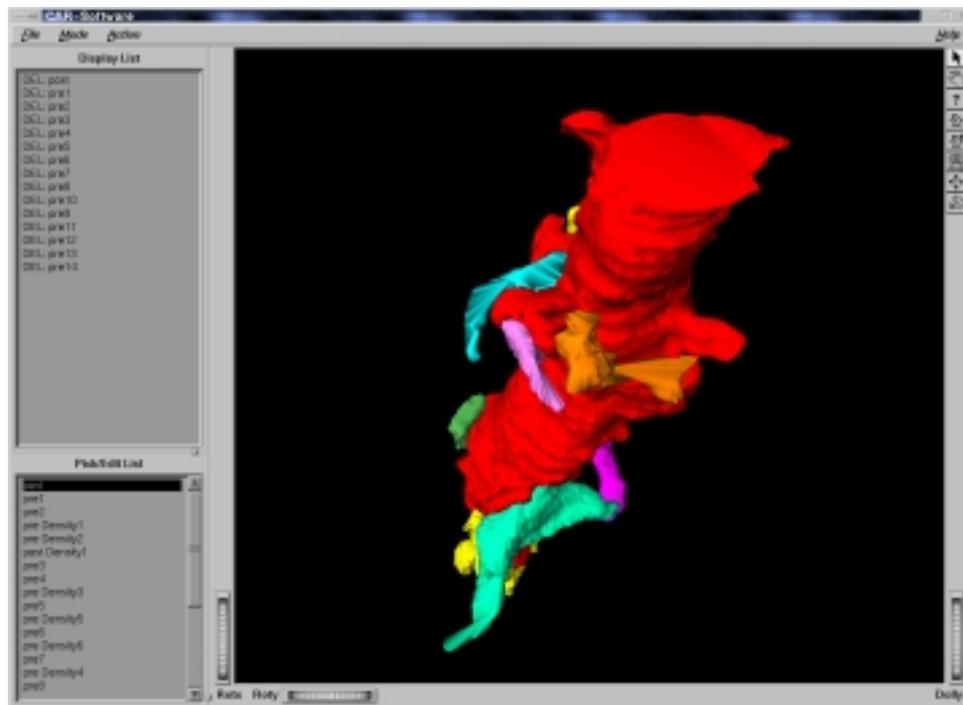
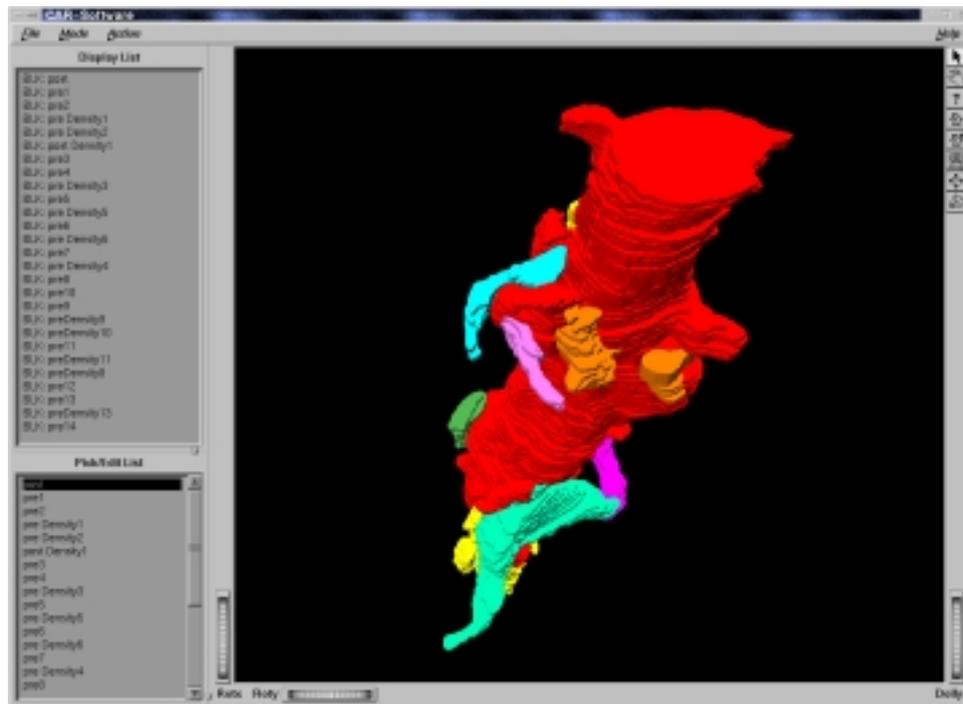


ABBILDUNG 2.7.6. Hier sind die in Abbildung (2.7.5) umfahrenen Strukturen rekonstruiert worden. Im oberen Bild wurde die Rekonstruktion im *Brick-Modus* durchgeführt, im unteren nach der *Delaunay-Rekonstruktionsmethode*, wie sie im Kapitel 3 näher vorgestellt werden wird.

### 3D–Rekonstruktion von Konturdaten

Die dreidimensionale Datenverarbeitung spielt in der medizinischen Bildverarbeitung eine immer größere Rolle. Dies ist auch im wesentlichen dadurch bedingt, daß immer mehr bildgebende Verfahren entwickelt werden, die es erlauben, biologische Präparate dreidimensional aufzunehmen.

Eines der wichtigsten Prinzipien hierbei ist es, optische Schnitte durch das Objekt herzustellen (für eine Übersicht siehe [Wi190]). Zu diesen Methoden gehören sowohl die Computertomographie, als auch die konfokale Lichtmikroskopie, um nur ein paar Beispiele zu nennen. Um aus den entstehenden, volumetrischen Datensätzen die dreidimensionale Struktur des untersuchten Objektes gewinnen zu können, muß aus diesem Bildmaterial erst das Objekt extrahiert oder segmentiert werden. Danach kann das segmentierte Bild in eine dreidimensionale Darstellung oder Repräsentation des Objektes überführt werden.

Als Beispiele für volumenorientierte Rekonstruktionsalgorithmen seien der *Marching-Cube*-Algorithmus (siehe [Wat00] und [FvDFH96]), und der sog. *Split-And-Merge*-Algorithmus genannt, bei dem der Rekonstruktions- und Segmentationsschritt zusammengefaßt werden (siehe [BPC93], [4], [EDB<sup>+</sup>96] und [6]). Beim *Marching-Cube*-Algorithmus erhält man eine Objektzerlegung in achsenorientierte Würfel, während der *Split-And-Merge*-Algorithmus das Objekt in Voronoi-polyeder zerlegt.

Neben dem *Optical-Sectioning* gibt es noch die Möglichkeit, das Objekt physikalisch in dünne Scheiben zu schneiden. Oft interessiert es bei dieser Vorgehensweise primär, qualitative Einblicke in das Objektinnere zu erhalten. Geht es um Auswertung quantitativer Daten, muß — wie in der bisherigen Arbeit vorgestellt — eine registrierte, digitalisierte Bildsequenz in drei Dimensionen erzeugt werden, bevor die volumenorientierten Methoden angewendet werden können.

Neben den volumetrischen Rekonstruktionsalgorithmen hat sich jedoch speziell bei den physikalischen Dünnschnitten eine zweite Art der Rekonstruktionsmethoden etabliert, die sich mit der *3D-Rekonstruktion von Konturdaten* beschäftigt (*3D reconstruction from serial sections*). Als Konturen versteht man hierbei die Ränder der Objektquerschnitte, die in den physikalischen (aber auch lichtoptischen)

Schnitten erfaßt wurden. Die konturbasierten Algorithmen beschäftigen sich vor allem mit der Fragestellung, wie man eine gültige Objektrepräsentation erzeugen kann, auch wenn die Objektquerschnitte von einer Schicht zur nächsten sehr stark variieren, d.h. wenn der Auflösungsunterschied in der Ebene im Vergleich zur Schnittdicke sehr groß wird. Hier versagen meist die volumenorientierten Methoden, da diese im wesentlichen von einem gleichmäßigen Sampling des Objektes in möglichst quaderförmige *Voxel* ausgehen<sup>1</sup>. Schlechte Auflösungsverhältnisse von ebener bzw. lateraler zur  $z$ - bzw. axialer Auflösung, führt zu stark elongierten Voxeln, die dann stufenartige Rekonstruktionen bedingen können.

So wollen wir uns im folgenden Kapitel ausführlich mit den Methoden zur Rekonstruktion von Konturdaten beschäftigen. Dabei wird zuerst ein Überblick über die zentralen Fragen des Rekonstruktionsproblems gegeben und verschiedene publizierte Algorithmen vorgestellt, die eine Lösung der vorliegenden Probleme versprechen. Nach der Diskussion ihrer Vor- und Nachteile, wird im weiteren speziell auf den in der vorliegenden Arbeit gewählten Rekonstruktionsansatz, der auf einer Delaunaytessellierung des Objektes basiert, eingegangen werden.

### 3.1. Das Rekonstruktionsproblem

Das Rekonstruktionsproblem für Konturdaten konzentriert sich auf drei wesentliche Teilaspekte (siehe Abbildung (3.1.1)):

1. Das Korrespondenzproblem (*Linking*-Problem)
2. Das Verbindungsproblem (*Tiling*-Problem)
3. Das Verzweigungsproblem (*Branching*-Problem)

Auf jeden dieser Aspekte wollen wir im folgenden kurz eingehen. Dabei geben wir einen Überblick über die gängigsten Methoden und Verfahren, die die vorgestellten Probleme lösen.

**3.1.1. Das Korrespondenzproblem.** Aus Abbildung (3.1.1)a ist ersichtlich, daß grundsätzlich das sog. Korrespondenz- oder auch *Linking*-Problem bei der Rekonstruktion von Konturdaten besteht. Daher müssen Annahmen darüber gemacht werden, ob zwei verschiedene Konturen auf unterschiedlichen Ebenen zueinander gehören oder nicht. Hier muß man sich für eine von vielen theoretisch möglichen Lösungen entscheiden, die einen plausiblen Übergang zwischen den verschiedenen Konturen beschreiben (siehe Abbildung (3.1.1)a: Wählt man die Repräsentation, wie sie durch die durchgezogene oder die gestrichelte Linie angegeben wurde?).

Dafür bedient man sich i.a. verschiedener Heuristiken, die eine eindeutige Lösung des Problems ermöglichen. Eine dieser Kriterien ist das sog. *Überlappungs-Kriterium*, welches nur dann einen Link zwischen

---

<sup>1</sup>Ein Voxel ist hierbei das dreidimensionale Äquivalent eines Pixels in zwei Dimensionen.

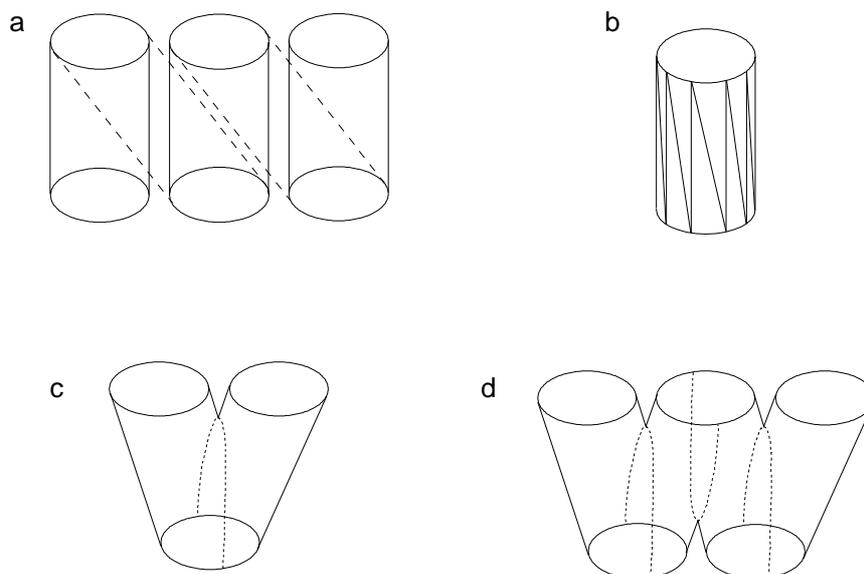


ABBILDUNG 3.1.1. Die drei wesentlichen Aspekte des Rekonstruktionsproblems für Konturdaten: (a) das Korrespondenzproblem (*linking*), (b) das Verbindungsproblem (*tiling*) und (c) + (d) das Verzweigungsproblem (*branching* und *multibranching*).

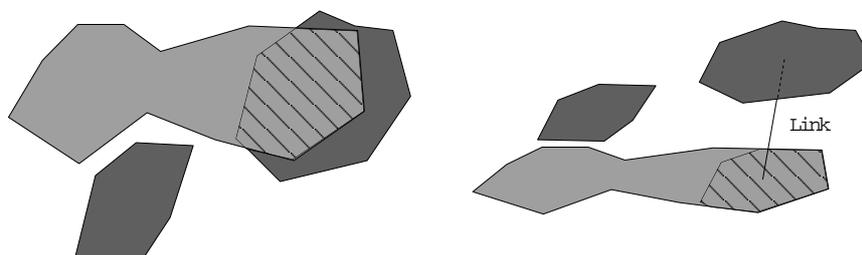


ABBILDUNG 3.1.2. Schematische Darstellung des Überlappungs-Kriteriums: Haben zwei Konturen in der Projektion eine Überlappung, so werden diese *gelinkt* (siehe rechte Seite). Links ist eine Draufsicht, rechts eine Seitenansicht zu sehen.

Konturen ermöglicht, wenn die jeweiligen Projektionen der Konturen in die Ebene eine nichtleere Schnittmenge aufweisen (siehe Abbildung (3.1.2)). Dieses Kriterium auf die abgebildete Konstellation in (3.1.1)a angewendet, würde eindeutig die Variante, die mit durchgezogener Linie markiert wurde, als einzig gültige sehen.

Da das *Überlappungs-Kriterium* in den meisten Fällen eine gesuchte Repräsentation liefert, findet es sich häufig in Algorithmen als Grundvoraussetzung wieder. Somit sind die meisten Algorithmen von diesem Kriterium in ihrer Funktionsweise abhängig ([OPC96]). Manche definieren gar ein stärkeres Kriterium ([BCL96]), andere wiederum

können diese Linkinformation *a priori* oder auch *a posteriori* in die Rekonstruktion einfließen lassen (z.B. [EPO91] und [Boi88]).

**3.1.2. Das Verbindungsproblem.** Das Verbindungs- oder *Tiling*-Problem stellt die Frage, wie man eine gültige Oberflächenrepräsentation zu gegebenen Konturen erzeugen kann. Unter einer gültigen Oberflächenrepräsentation versteht man hierbei eine geschlossene Oberfläche, die keine Löcher aufweist, sich nicht selbst durchdringt und die vorgegebenen Konturen bei einem Schnitt der Objektoberflächen auf Höhe einer der Schnittebenen exakt reproduziert. Bei den ersten Untersuchungen zu dieser Problemstellung konnte man feststellen, daß man das neue Korrespondenzproblem<sup>2</sup> in eine optimierte Pfadsuche in einem gerichteten Graphen überführen kann (siehe [FKU77] und [Kep75]). Die Konturpunkte bilden dabei die Knoten des Graphen, während die Verbindungslinien zwischen den einzelnen Konturen die Kanten des Graphen markieren.

Um nun die mögliche Oberflächenrepräsentation zu optimieren, wurde vorgeschlagen, auf die Knoten des Graphen verschiedene Gewichte zu legen. Als Möglichkeiten wurden dabei das Volumen ([Kep75]), die Oberfläche ([FKU77]) oder eine Form von gewichteter Konturlänge ([GD82]) vorgeschlagen. Wie man jedoch schnell erkennen mußte, können die einfachen *Tiling*-Algorithmen lediglich bei einfachen, einander sehr ähnlichen und möglichst konvexen Polygonen gültige Oberflächenrepräsentationen erzeugen (was passiert, wenn eine kreisförmige Kontur in eine spiralförmige übergeht?). Die größte Unzulänglichkeit der meisten *Tiling*-Algorithmen der Anfangszeit spiegelt sich in der Tatsache wider, daß sie das Verzweigungsproblem, oder gar ineinander verschachtelte Konturen (siehe Abbildung (3.3.6) auf Seite 70) nicht in eine gültige Oberflächerepräsentation überführen können. Dies führt uns sogleich zur nächsten und letzten Problemstellung.

**3.1.3. Das Verzweigungsproblem.** Beim Verzweigungs- oder *Branching*-Problem handelt es sich ganz allgemein um die Frage, wie man große topologische Änderungen der Konturen zweier Ebenen, wie z.B. Aufspaltungen (siehe Abbildung (3.1.1)c) am besten in eine gültige Oberflächenrepräsentation überführen kann. Auf diesem Gebiet wurde intensiv geforscht, wobei die meisten Algorithmen versuchen, die einfachen *Tiling*-Algorithmen dahingehend zu erweitern, daß diese nun auch das Verzweigungsproblem beschreiben können (siehe [GD82], [EPO91] und [MSS92]). Darauf aufbauend schlägt Choi vor, das Mehrfachverzweigungsproblem, bei dem viele Konturen sich in noch mehr Konturen verzweigen können (siehe Abbildung (3.1.1)d), in ein einfaches Verzweigungsproblem zu überführen (siehe [CP94]), welches dann nach einem der Standardalgorithmen gelöst werden kann.

---

<sup>2</sup>wie kann man Konturpunkten der einen Kontur am optimalsten Punkte der anderen Kontur zuordnen — siehe Abbildung (3.1.1)b

Von Bajaj wird in [BCL96] vorgeschlagen, das *Tiling* iterativ durchzuführen, um die Triangulierung nach und nach zu optimieren. Dabei werden die Regionen der Kontur von der Triangulierung zunächst ausgenommen, die einer Verzweigung oder einem Kanal entsprechen (siehe Abbildung (3.4.1) auf Seite 71). Diese Regionen werden nach abgeschlossener Optimierung des *Tilings* durch Standardrepräsentationen mit Hilfe des Inneren Skelettes oder der Inneren Linie trianguliert.

Eine weitere, von Oliva publizierte Methode (siehe [OPC96]), unterteilt die Konturen in Untergebiete, die den Schnitt- und Differenzmengen der Konturen entsprechen (siehe auch Abbildung (3.1.2)), was ein striktes Einhalten des Überlappungs-Kriteriums voraussetzt. Nichtleere Schnittmengen signalisieren dabei, daß die entsprechenden Strukturen gelinkt werden. Sind also die Schnittmengen für das *Linking* maßgeblich, so sind die Differenzmengen für das *Tiling* verantwortlich. Bei der Methode von Oliva werden die Differenzmengen als Polygone beschrieben, die von angrenzenden Konturabschnitten gebildet werden. Dieses Übergangsgebiet kann nun mit Hilfe des LVD (Linienvoronoidiagramm oder *Angular-Bisector-Network*) sukzessive verfeinert werden, bis die dabei entstehenden, vereinfachten Subregionen trianguliert werden können. Durch diesen Prozeß erhält man also interpolierende Zwischenschichten, die den Übergang (oder das Morphing) von einer Kontur zu nächsten beschreiben. Allerdings verfügt der iterative Prozeß über kein natürliches Abbruchkriterium.

Eine weitere Methode, die auch dieser Arbeit zugrunde gelegt wurde, schlägt vor, das Korrespondenz-, Verbindungs- und Verzweigungsproblem mit Hilfe der Delaunaytessellierung der Konturdaten zu lösen (siehe [Boi88]). Dabei wird der Raum zwischen den Konturen unabhängig von der Schnittdicke in Tetraeder zerlegt. Diese Tetraeder werden dann binarisiert, so daß in jedem Falle eine konsistente Oberflächenrepräsentation von *Inneren* zu *Äußeren* Tetraedern entsteht. Topologische Unzulänglichkeiten werden dabei durch Einführen von *Inneren Punkten* innerhalb der Konturflächen behoben (siehe [BG92a]). Mehr hierzu findet der Leser in den Abschnitten 3.3 bis 3.5.1 in dieser Arbeit.

### 3.2. Grundlagen zum 3D Rekonstruktionsalgorithmus

In diesem Kapitel werden die grundlegenden Begriffe und Definitionen erarbeitet, die für ein Verständnis des 3D Rekonstruktionsalgorithmus im allgemeinen und des Rekonstruktionsansatzes von Boissonat im speziellen notwendig sind.

**3.2.1. Grundlegende Definitionen und Begriffe zur Geometrie.** In diesem Abschnitt werden die grundlegenden Begriffe und Definitionen aus der Geometrie, sowie verschiedene Mengenbegriffe eingeführt und vorgestellt. Diese Informationen sind z.T. aus den Werken

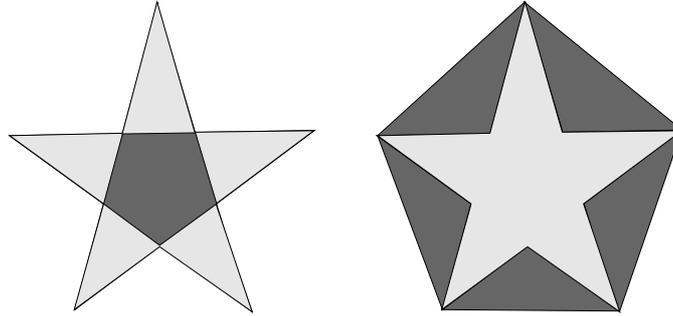


ABBILDUNG 3.2.1. In dieser Abbildung ist ein — im wahrsten Sinne des Wortes — typisches Sterngebiet abgebildet. In der Mitte wird der sog. Kern des Sterngebietes hervorgehoben. Rechts ist die konvexe Hülle des Sterngebietes gezeigt. Ein Sterngebiet ist also nicht notwendigerweise konvex.

[HL92] und [OBS92] entnommen und hier in kompakter Form zusammengestellt worden. Für eine vertiefende Studie seien diese Werke empfohlen.

DEFINITION 3.2.1. Offene und abgeschlossene Mengen

Es sei  $A$  ein Unterraum des  $\mathbb{R}^m$ .  $A$  heißt nun *offen*, wenn für jeden Punkt  $\mathbf{a} \in A$  ein  $\varepsilon > 0$  existiert, so daß:

$$U_\varepsilon(\mathbf{a}) := \{\mathbf{x} \in \mathbb{R}^m \mid \|\mathbf{x} - \mathbf{a}\| < \varepsilon\} \subset A$$

So existiert zu jedem Punkt  $\mathbf{a}$  von  $A$  eine nicht leere Umgebung  $U_\varepsilon(\mathbf{a})$ , die noch in  $A$  enthalten ist.

$A$  heißt hingegen *abgeschlossen*, wenn das Komplement  $\mathbb{R}^m \setminus A := \{\mathbf{x} \mid \mathbf{x} \notin A\}$  offen ist.

DEFINITION 3.2.2. Sterngebiet, konvexer Unterraum, Konvexe Hülle (siehe Abbildung (3.2.1))

1. Existiert in  $A$  mindestens ein ausgezeichnete Punkt  $\mathbf{a}_0$  derart, daß gilt:

$$\mathbf{a}_0 + \lambda(\mathbf{a} - \mathbf{a}_0) \in A \forall 0 \leq \lambda \leq 1 \wedge \mathbf{a} \in A,$$

so bildet  $A$  ein sog. Sterngebiet.  $\mathbf{a}_0$  wird hierbei Sternpunkt genannt. Die Menge aller Sternpunkte  $A_K$  eines Gebietes  $A$ , definiert als:

$$A_K := \{\mathbf{a}_0 \mid \mathbf{a}_0 + \lambda(\mathbf{a} - \mathbf{a}_0) \in A \forall 0 \leq \lambda \leq 1 \wedge \mathbf{a} \in A\}$$

nennen wir *Kern* des Sterngebietes  $A$ . Dies führt uns gleich zum nächsten Begriff der *Konvexität*.

2. Es sei  $A$  ein Unterraum des  $\mathbb{R}^m$ . Wenn für zwei beliebige Elemente  $\mathbf{a}_1, \mathbf{a}_2 \in A$  gilt:

$$\mathbf{a}_1 + \lambda(\mathbf{a}_2 - \mathbf{a}_1) \in A \forall 0 \leq \lambda \leq 1 \wedge \mathbf{a}_1, \mathbf{a}_2 \in A$$

so nennen wir  $A$  einen *konvexen Unterraum* des  $\mathbb{R}^m$ .

3. Es sei  $A$  ein Unterraum des  $\mathbb{R}^m$ . Dann nennen wir:

$$C(A) := \bigcap C_A^j \text{ mit } C_A^j \supseteq A \text{ und } C_A^j \text{ konvex } \forall j \in I_n$$

*konvexe Hülle* von  $A$ .

DEFINITION 3.2.3. Hyperebene und Halbräume im  $\mathbb{R}^m$

Gegeben seien die Vektoren  $\{\mathbf{x}_1, \dots, \mathbf{x}_m\} \in \mathbb{R}^m$  und die Indexmenge  $I_{m-1} = \{1, \dots, m-1\}$ . Sind die Vektoren  $\{\mathbf{x}_1 - \mathbf{x}_m, \dots, \mathbf{x}_{m-1} - \mathbf{x}_m\}$  linear unabhängig, so bezeichnet der Unterraum:

$$H = \left\{ \mathbf{x} \mid \mathbf{x} = \mathbf{x}_m + \sum_{i=1}^{m-1} \lambda_i (\mathbf{x}_i - \mathbf{x}_m) \lambda_i \in \mathbb{R}, i \in I_{m-1} \right\}$$

eine *Hyberebene* im  $\mathbb{R}^m$ .

Gilt weiterhin für zwei Punkte  $\mathbf{p}_i$  und  $\mathbf{p}_j$  die Eigenschaft, daß  $\|\mathbf{x} - \mathbf{p}_i\| = \|\mathbf{x} - \mathbf{p}_j\| \forall \mathbf{x} \in H$ , so ist  $H$  die Begrenzungsfläche des durch:

$$H(\mathbf{p}_i, \mathbf{p}_j) := \{ \mathbf{x} \mid \|\mathbf{x} - \mathbf{p}_i\| \leq \|\mathbf{x} - \mathbf{p}_j\| \}$$

definierten *Halbraumes*  $H$ . Der Vektor  $\mathbf{n} := \mathbf{p}_i - \mathbf{p}_j$  wird dabei als *Normale* der Hyperebene  $H$  bezeichnet und steht senkrecht auf den Vektoren  $\{\mathbf{x}_1 - \mathbf{x}_m, \dots, \mathbf{x}_{m-1} - \mathbf{x}_m\}$ . Die Vektoren  $\{\mathbf{n}, \mathbf{x}_1 - \mathbf{x}_m, \dots, \mathbf{x}_{m-1} - \mathbf{x}_m\}$  bilden demnach eine Basis des  $\mathbb{R}^m$ .

BEMERKUNG 3.2.4. In der Definition der Halbräume kann  $\|\cdot\|$  auch eine beliebige Norm repräsentieren. Dann ist aber die Aussage, daß  $H$  die Trennfläche der beiden Halbräume  $H(\mathbf{p}_i, \mathbf{p}_j)$  und  $H(\mathbf{p}_j, \mathbf{p}_i)$  markiert, nicht mehr gültig.

DEFINITION 3.2.5. Simplex, Polyeder

Ein Simplex  $A \subset \mathbb{R}^m$  der Ordnung  $m$  ist definiert durch einen beliebigen Satz von  $m+1$  Punkten  $\{\mathbf{p}_1, \dots, \mathbf{p}_{m+1}\}$ , welche nicht auf einer Hyperebene im  $\mathbb{R}^m$  liegen. Allgemein kann ein Simplex der Ordnung  $m$  wie folgt geschrieben werden:

$$A = \left\{ \mathbf{x} = \sum_{i=1}^{m+1} \lambda_i \mathbf{p}_i \mid \sum_{i=1}^{m+1} \lambda_i = 1, \lambda_i \geq 0, i \in I_{m+1} = \{1, \dots, m+1\} \right\}$$

Ein Unterraum  $A \subset \mathbb{R}^m$ , den man durch eine endliche Anzahl von Vereinigungen und Schnitten von Halbräumen  $H$  des  $\mathbb{R}^m$  enthält, wobei  $A$  abgeschlossen und zusammenhängend ist, nennt man einen *Polyeder*.

BEMERKUNG. Die Begrenzungsfläche eines Polyeders im  $\mathbb{R}^m$  besteht aus Polyedern des  $\mathbb{R}^{m-1}$ , welche man als  $(m-1)$ -Flächen oder Facetten bezeichnet. Ein Simplex ist ein *konvexer* Polyeder. So nennt man z.B. einen Simplex der Ordnung 3 einen Tetraeder, dessen 2-Flächen Dreiecke bilden. Diese sind nun wiederum Simplexe der Ordnung 2 (siehe Abbildung (3.2.2)). 0-Flächen sind Punkte, 1-Flächen sind Kanten.

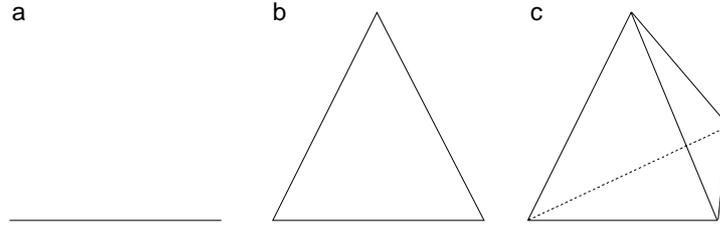


ABBILDUNG 3.2.2. (a) Simplex der Ordnung 1; (b) Simplex der Ordnung 2; (c) Simplex der Ordnung 3

DEFINITION 3.2.6. Tessellierung (Zerlegung)

Sei  $A$  eine abgeschlossene Menge des  $\mathbb{R}^m$ . Sei weiterhin die Menge  $\mathcal{A} = \{A_1, \dots, A_n\}$  gegeben, bei der die Untermengen  $A_i \subset A$  ebenfalls abgeschlossen sind und  $\partial A_i$  deren Rand bezeichnet. Genügen die Untermengen  $A_i \in \mathcal{A}$  den Bedingungen

$$[A_i \setminus \partial A_i] \cap [A_j \setminus \partial A_j] = \emptyset \forall i, j \in \{1, \dots, n\} \wedge i \neq j$$

und

$$\bigcup_{i=1}^n A_i = A,$$

dann nennen wir  $\mathcal{A}$  eine Tessellierung von  $A$ .

**3.2.2. Grundlegende Definitionen und Begriffe zum Punktvoronoidiagramm.** Im folgenden sollen ein paar grundlegende Definitionen und Eigenschaften von Voronoidiagrammen und Delaunaygraphen erläutert werden. Wir betrachten hierbei ausschließlich das sog. Punktvoronoidiagramm (kurz PVD), welches stets zu einer gegebenen Punktmenge  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$  des  $\mathbb{R}^m$  definiert und erzeugt wird. Vor allem der Blick auf die Eigenschaften dieser dreidimensionalen Tessellierungen gibt eine Vorstellung von dem Potential und den Möglichkeiten, die diese Representationen des  $\mathbb{R}^m$  bietet, nicht nur für die in dieser Arbeit vorgestellte Problemstellung der Rekonstruktion von planaren Konturen.

3.2.2.1. Definitionen zum gewöhnlichen Voronoidiagramm.

DEFINITION 3.2.7. Voronoipolyeder, das gewöhnliche Voronoidiagramm

Gegeben sei  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$  mit  $n \in [2, \infty)$  und  $\mathbf{p}_i \neq \mathbf{p}_j, i \neq j \forall i, j \in I_n := \{1, \dots, n\}$  im  $\mathbb{R}^m$ . Dann bezeichnet

$$V(\mathbf{p}_i) = \bigcap_{j \in I_n, j \neq i} H(\mathbf{p}_i, \mathbf{p}_j)$$

das sog. *gewöhnliche Voronoipolyeder*, das mit  $\mathbf{p}_i$  assoziiert ist. Der Halbraum  $H(\mathbf{p}_i, \mathbf{p}_j)$  bezeichnet hierbei die sog. *Dominanzregion* von  $\mathbf{p}_i$ . Das Voronoipolyeder ist somit aufgrund seiner Definition konvex.

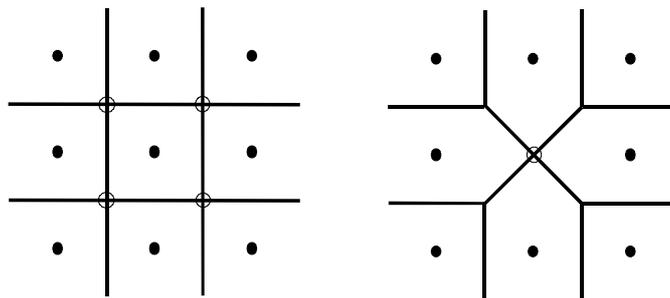


ABBILDUNG 3.2.3. Die linke und rechte Abbildung zeigen degenerierte Voronoidiagramme. Dabei sind diejenigen Voronoikantenpunkte mit einem Kreis markiert, die im  $\mathbb{R}^2$  mehr als 3 Kantennachbarn aufweisen.

Weiterhin bezeichnet

$$\mathcal{V}(P) = \{V(\mathbf{p}_i); i \in I_n\}$$

das von  $P$  erzeugte, gewöhnliche Voronoidiagramm.  $p_i$  nennt man dabei den *erzeugenden Punkt* oder *seed* des *iten* Voronoipolyeders und  $P$  die *Menge aller erzeugenden Punkte* des Voronoidiagramms  $\mathcal{V}(P)$ .

BEMERKUNG 3.2.8. Bei der erweiterten Definition zum *generalisierten Voronoidiagramm* wird anstelle der euklidischen Norm, wie sie in der Definition der Halbräume  $H(\mathbf{p}_i, \mathbf{p}_j)$  auftaucht, eine beliebige Norm eingesetzt. Weiterhin wird der Begriff der Punktmenge auf Linien, Flächen, Polygone, o.ä. verallgemeinert. Eine ausführliche Darstellung von generalisierten Voronoidiagrammen findet man in [OBS92].

DEFINITION 3.2.9. Voronoipunkt, degenerierte Voronoidiagramme

Sei  $\mathcal{V}(P)$  ein Voronoidiagramm im  $\mathbb{R}^m$ . Als *Voronoipunkt* im  $\mathbb{R}^m$  wird hierbei der eindeutige Schnittpunkt zwischen  $m$  Hyperebenen bzw. Seitenflächen des zugehörigen Polyeders bezeichnet, deren Normalenvektoren eine Basis des  $\mathbb{R}^m$  bilden. Gibt es mindestens einen Voronoipunkt, der zu mehr als  $(m + 1)$  Kanten des Voronoidiagramms  $\mathcal{V}(P)$  gehört, so nennen wir das Voronoidiagramm *degeneriert* (vgl. Abbildung (3.2.3)).

3.2.2.2. *Definitionen zum Delaunaygraphen.* Im weiteren wollen wir annehmen, daß wir es nicht mit degenerierten Voronoidiagrammen zu tun haben. Außerdem soll die im folgenden beschriebene *Nicht-Kolinearitätsbedingung* gelten:

BEDINGUNG 3.2.10. Die Nicht-Kolinearitätsbedingung

Gegeben seien die Punkte  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$  des  $\mathbb{R}^m$ , mit  $n \geq m + 1$ . Die Punkte  $P$  heißen nicht kolinear, wenn die Vektoren aus  $P$  eine Basis des  $\mathbb{R}^m$  bilden, also nicht auf einer Hyperebene  $H$  des  $\mathbb{R}^m$  liegen.

BEDINGUNG 3.2.11. Die Eindeutigkeitsbedingung

Es gelte die Nicht-Kolinearitätsbedingung. Außerdem sollen die Punkte in  $P$  derart sein, daß  $\mathcal{V}(P)$  ein nicht degeneriertes Voronoidiagramm bildet.

Diese Annahmen sind für den Ausschluß von topologischen Mehrdeutigkeiten notwendig und garantieren sowohl die Existenz als auch die Eindeutigkeit des Delaunaygraphen, der wie folgt definiert ist:

DEFINITION 3.2.12. Der  $m$ -dimensionale Delaunaygraph

Der duale Graph zum Voronoidiagramm  $\mathcal{V}(P)$  einer Punktmenge  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$ , die der Eindeutigkeitsbedingung 3.2.11 genügen, ist der sog. Delaunaygraph. Zwei Punkte  $\mathbf{p}_i$  und  $\mathbf{p}_j$  der Punktmenge  $P$  bilden dann und nur dann eine Kante  $\kappa(\mathbf{p}_i, \mathbf{p}_j) \in K$ ,  $K \subset P \times P$  des Delaunaygraphen  $\mathcal{D}(P) = \langle P, K \rangle$ , falls folgende Bedingung erfüllt ist:

$$V(\mathbf{p}_i) \cap V(\mathbf{p}_j) \neq \emptyset,$$

d.h. daß die Voronoi-Polyeder  $V(\mathbf{p}_i)$  und  $V(\mathbf{p}_j)$  aneinander angrenzen<sup>3</sup>.

BEMERKUNG 3.2.13. Dualität besteht im verallgemeinerten Falle von  $m$  Dimensionen zwischen den Kanten des Delaunaygraphen  $\mathcal{D}(P)$  und den  $(m - 1)$ -Flächen des Voronoidiagramms  $\mathcal{V}(P)$ .

An diese Definition läßt sich sogleich die Definition der Delaunaynachbarn eines Punktes  $\mathbf{p}_i$  einführen:

DEFINITION 3.2.14. Delaunaynachbarschaft

Die Nachbarn  $N_P$  eines Punktes  $\mathbf{p}_i \in P$  im Delaunaygraphen sind durch die Menge der Punkte

$$N_P(\mathbf{p}_i) = \{\mathbf{p}_j \in P \mid j \neq i \wedge \kappa(\mathbf{p}_i, \mathbf{p}_j) \in K\}$$

gegeben, wenn  $K$  die Kantenmenge von  $\mathcal{D}(P)$  beschreibt.

SATZ 3.2.15. Die  $m$ -dimensionale Delaunaytessellierung der konvexen Hülle

Unter der Annahme 3.2.11 kann der Delaunaygraph auch als eindeutige Simplexzerlegung der konvexen Hülle  $C(P)$  der Punktmenge  $P \subset \mathbb{R}^m$  aufgefaßt werden, wobei die umschreibende Kugel  $S(T_i)$  der Simplexe  $T_i = (\mathbf{p}_{i_1}, \dots, \mathbf{p}_{i_{m+1}}) \in P^{m+1}$  keinen weiteren Punkt von  $P$  enthält. Somit kann das Delaunaydiagramm  $\mathcal{D}(P)$  als folgende Menge definiert werden:

$$\mathcal{D}(P) := \{T_i \in P^{m+1} \mid S(T_i) \cap P \setminus P_{T_i} = \emptyset\}$$

Eine nähere Erläuterung und den Beweis dieser Behauptung findet man in [PS88]. Aus dieser Behauptung leitet sich folgende Definition des Delaunaysimplex ab:

---

<sup>3</sup>Bei nicht degenerierten Voronoidiagrammen, wie hier angenommen, berühren sich die Polyeder in einer  $(m - 1)$ -Fläche.

DEFINITION 3.2.16. Der Delaunaysimplex

Für eine Punktmenge  $P \subset \mathbb{R}^m$  nennt man den Simplex  $T_i = (\mathbf{p}_{i_1}, \dots, \mathbf{p}_{i_{m+1}}) \in P^{m+1}$  dann und nur dann *Delaunaysimplex*, wenn die umschreibende Kugel  $S(T_i)$  keinen weiteren Punkt von  $P$  enthält, also gilt:

$$S(T_i) \cap P \setminus P_{T_i} = \emptyset$$

In diesem Fall nennt man 2 Punkte  $\mathbf{p}, \mathbf{q} \in P_{T_i} = \{\mathbf{p}_{i_1}, \dots, \mathbf{p}_{i_{m+1}}\}, \mathbf{p} \neq \mathbf{q}$  Nachbarn im Delaunaygraphen  $\mathcal{D}(P)$  und  $\kappa(\mathbf{p}, \mathbf{q})$  eine Kante darin.

SATZ 3.2.17. Delaunaysterngebiet

Die Menge aller Simplexe  $T_{\tilde{P}}$ , die durch die Punktmenge  $\tilde{P} := N_P(\mathbf{p}_i) \cup \{\mathbf{p}_i\}$  definiert ist, bildet ein Sterngebiet  $\cup T_{\tilde{P}}$ .

BEWEIS. Da die Delaunaysimplexe per Definition konvex sind, formen diese — jeder für sich gesehen — automatisch ein Sterngebiet. Da  $\mathbf{p}_i$  Teil eines jeden Simplexes ist, ist zumindest  $\mathbf{p}_i$  ein sog. Sternpunkt.  $\square$

Die Eindeutigkeitsbedingung 3.2.11 kann nun auch wie folgt umformuliert werden:

BEDINGUNG 3.2.18. Die Nicht-Kozyklizitätsbedingung

Eine nicht kollineare Punktmenge  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\} \subset \mathbb{R}^m$  mit  $n \geq m + 1$  nennt man zusätzlich *nicht kozyklisch*, wenn jede offene Kugel  $S_o := S \setminus \partial S$ , die keine Punkte von  $P$  im Innern enthält ( $S_o \cap P = \emptyset$ ), maximal  $m + 1$  Punkte auf dem Rand  $\partial S$  besitzt.

### 3.2.3. Eigenschaften des PVD und des Delaunaygraphen.

In diesem Abschnitt werden wesentliche Eigenschaften des PVD und der Delaunaytessellierung herausgearbeitet, die sich aus den Definitionen, die in den vorherigen Abschnitten erarbeitet und vorgestellt wurden, ableiten lassen. Diese Folgerungen sind z.T. zum Verständnis der Vorgehensweise bei der Konstruktion des PVD und dem 3D Rekonstruktionsalgorithmus notwendig und geben gleichzeitig einen Überblick über die Möglichkeiten, die beide Partitionierungen des Raumes bieten. Diese werden anhand der Lösung von häufig auftauchenden Problemen in der Computergeometrie diskutiert.

FOLGERUNG 3.2.19. Voronoipunkt, erzeugende Punkte

Unter Annahme der Bedingung 3.2.11 entsteht ein *Voronoipunkt*  $\mathbf{v}$  im  $\mathbb{R}^m$  als Schnittpunkt zwischen exakt  $m + 1$  Polyedern  $V(\mathbf{p}_{i_1}), \dots, V(\mathbf{p}_{i_{m+1}})$ , d.h. der Voronoipunkt gehört exakt  $m + 1$  Polyedern an. Man nennt die Punkte  $\mathbf{p}_{i_1}, \dots, \mathbf{p}_{i_{m+1}}$  *erzeugende* oder *generierende Punkte* des Voronoipunktes  $\mathbf{v}$ .

FOLGERUNG 3.2.20. Voronoinachbarn

Es sei  $\mathbf{v}$  ein Voronoipunkt des Voronoidiagramms  $\mathcal{V}(P)$ . Unter Annahme der Bedingung 3.2.11 existieren genau  $m + 1$  Nachbarvoronoipunkte.

FOLGERUNG 3.2.21. Der nächste Punkt  
Es seien 2 Punkte  $\mathbf{p}, \mathbf{q} \in P$  gegeben. Gilt:

$$\|\mathbf{p} - \mathbf{q}\| < \|\mathbf{p} - \mathbf{p}_i\| \quad \forall \mathbf{p}_i \in P \setminus \{\mathbf{p}, \mathbf{q}\},$$

so bildet  $\kappa(\mathbf{p}, \mathbf{q})$  eine Kante des Delaunaygraphen  $\mathcal{D}(P)$ .

Diese Aussage bietet sogleich eine Lösung für die folgenden beiden gleichwertigen Problemstellungen, wie sie in [SH75] und [BWY80] vorgestellt wurden:

PROBLEM 3.2.22. *Das Kleinste-Punkt-Abstand-Suchproblem* (The closest pair problem)

Gegeben sei eine endliche Menge von Punkten  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$ . Finde dasjenige Punktepaar  $\{\mathbf{p}_i, \mathbf{p}_j\} \in P \times P$ , für welches der Abstand  $d(\mathbf{p}, \mathbf{q}) := \|\mathbf{p} - \mathbf{q}\|$  minimal wird.

PROBLEM 3.2.23. *Das Nächste-Nachbar-Problem* (The all nearest neighbour problem)

Gegeben sei eine endliche Menge von Punkten  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$ . Finde zu einem Punkt  $\mathbf{p} \in P$  denjenigen Punkt  $\mathbf{q} \in P$ , der minimalen Abstand von  $\mathbf{p}$  hat.

Ein weiteres, zu den vorher erwähnten Problemstellungen verwandtes Problem ist das folgende:

PROBLEM 3.2.24. *Das Nächste-Nachbar-Suchproblem* (The nearest neighbour search problem)

Gegeben seien zwei endliche Mengen von Punkten  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$  und  $Q = \{\mathbf{q}_1, \dots, \mathbf{q}_m\}$ . Suche zu jedem Punkt  $\mathbf{q} \in Q$  den nächsten Punkt  $\mathbf{p}_q \in P$ .

Ist  $m$  sehr klein, kann es durchaus noch praktikabel sein, nach der *Brute-Force*-Methode alle  $n \cdot m$  möglichen Abstände zwischen den Punkten in  $P$  und  $Q$  zu berechnen. Bei größeren Punktemengen ist es jedoch effizienter, das Voronoidiagramm von  $P$  (oder  $Q$ ) zu bestimmen, um sich dann folgende Tatsache zu Nutze zu machen:

FOLGERUNG 3.2.25. Der erzeugende Punkt  $\mathbf{p} \in P$  ist dann und nur dann der nächste Punkt zum gesuchten Punkt  $\mathbf{q}$ , wenn  $\mathbf{q}$  im Voronoi-polyeder  $V(\mathbf{p})$  enthalten ist, also gilt  $\mathbf{q} \in V(\mathbf{p})$ .

BEMERKUNG 3.2.26. Aufgrund der Dualität zwischen Voronoidiagramm und Delaunaygraph, kann man die Lösung des Problems auch durch eine Suche im Delaunaygraph effizient lösen. Genau in diesen unterschiedlichen und doch äquivalenten Methoden zur Suche des nächsten Punktes unterscheiden sich die beiden inkrementellen Ansätze zur Berechnung des PVD von Bowyer und Watson (siehe [Bow81] und [Wat81]), wie wir im Abschnitt 3.2.4.2 sehen werden.

**3.2.4. Algorithmen zur Berechnung des PVD.** Im folgenden Abschnitt werden mögliche Algorithmen vorgestellt, die zur Berechnung des  $m$ -dimensionalen Voronoidiagramms dienen können. In dieser Arbeit soll dieser allgemeine Fall diskutiert werden, obwohl die ursprünglich von Boissonat und Geiger vorgestellten Algorithmen (siehe [Boi88] und [BG92a]), welche die Grundlage für die in dieser Arbeit vorgestellte Rekonstruktionsmethode bilden, nur zweidimensionale Voronoidiagramme voraussetzen.

Dies hat zum einen den Grund, daß die 3D Sichtweise des Boissonat'schen Ansatzes, wie er im Abschnitt 3.3 vorgestellt wird, eine Verallgemeinerung dieser Methode auf den  $m$ -dimensionalen Fall zu versprechen scheint<sup>4</sup>. Zum anderen scheint die Einführung von interpolierenden Punkten in Zwischenschichten, wie sie auch in [OPC96] und [EPO91] vorgestellt werden, der vielversprechendere Ansatz zu sein, um topologische Schwierigkeiten in der Rekonstruktion auszugleichen, als zusätzliche Punkte in die Konturebene selbst zu integrieren (siehe [BG92a]). Dies wird später in dieser Arbeit noch ausführlicher diskutiert werden (siehe Seite 70ff).

Da diese Zwischenpunkte in den dreidimensionalen Delaunaygraphen eingefügt werden sollen, ist es ebenfalls notwendig, eine auf (mindestens) 3 Dimensionen erweiterbare Berechnungsmethode des PVD's zu finden, welche das nachträgliche Einfügen von Punkten in die schon aufgebaute Datenstruktur erlaubt. Dies ist nur durch die sog. *inkrementellen Methoden*, wie wir sie in diesem Abschnitt diskutieren werden, effizient möglich.

Der Vollständigkeit wegen sollen im folgenden die Methoden zur Berechnung des PVD kurz vorgestellt werden, die ebenfalls auf  $m$  Dimensionen erweiterbar sind.

3.2.4.1. *Methoden zur Berechnung des  $m$ -dimensionalen Voronoidiagramms.* Viele Algorithmen, die in 2 Dimensionen vorliegen, sind nicht auf den verallgemeinerten Fall, der hier betrachtet werden soll, übertragbar. So nutzt der 2D-Algorithmus von Green und Sibson ([GS78]) z.B. die Tatsache aus, daß ein Polygon im  $\mathbb{R}^2$  anordenbar ist, d.h. eine Konturnachbarschaft vorliegt (wie wir den Abbildungen (3.3.1) und (3.3.6) auf den Seiten 62 und 70 entnehmen können), was nicht auf den mehrdimensionalen Fall übertragbar ist.

Zu den Generalisten unter den Algorithmen im multidimensionalen Sinne gehören:

1. die inkrementellen Methoden (siehe Abschnitt 3.2.4.2)
2. die Methoden zur Berechnung der konvexen Hülle (*Convex-Hull-Methode*)
3. die *Divide-And-Conquer*-Methode

---

<sup>4</sup>die Verallgemeinerung des Boissonatschen Ansatzes auf  $m$  Dimensionen soll in einer weiterführenden Arbeit untersucht werden

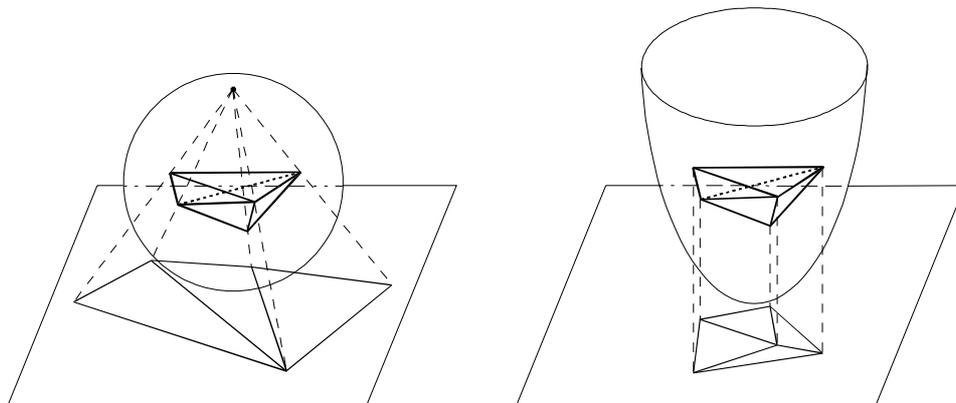


ABBILDUNG 3.2.4. Bei der *Inversion*-Transformation, werden alle Dreiecke der konvexen Hülle, bei denen der Projektionsursprung und die Punkte der konvexen Hülle in ein und demselben der beiden, durch das Dreieck definierten Halbräume liegen, zurücktransformiert und formen die Delaunaytessellierung (links). Bei der *Lift-Up*-Transformation werden diejenigen Dreiecke der konvexen Hülle ausgewählt, bei denen alle Punkte der konvexen Hülle über der vom Dreieck definierten Ebene liegen (rechts).

Die letzten beiden Algorithmen gehören zu den sog. *globalen Methoden*, bei denen die Punktmenge  $P$ , von der das Voronoidiagramm bestimmt werden soll, zu Beginn des Algorithmus vollständig bekannt sein muß. Ein nachträgliches Einfügen von Punkten führt dabei unweigerlich zu einer Neuberechnung des kompletten Voronoidiagramms. Eine nachträgliche Verfeinerung der Rekonstruktion, wie sie im Abschnitt 3.4 vorgestellt wird, ist also nur mit viel Rechenaufwand möglich.

Bei der sog. *Convex-Hull-Methode* nutzt man die Tatsache aus, daß man das  $m$ -dimensionale Voronoidiagramm von Punkten  $P$  des  $\mathbb{R}^m$  als die Projektion eines Voronoidiagramms des  $\mathbb{R}^{m+1}$  bestimmen kann, wenn man zuvor die Punkte durch eine Abbildung  $\mathbb{R}^m \rightarrow \mathbb{R}^{m+1} : \varphi(P) \rightarrow \tilde{P}$  auf den Rand eines konvexen Gebietes abbildet. Für die Berechnung des Voronoidiagramms im  $\mathbb{R}^{m+1}$  kann man sich dann schneller und effizienter Algorithmen bedienen, die sich die konvexe Anordnung der Punkte  $\tilde{P}$  zunutze machen (siehe [Sei90] und [CS89]). Mögliche Abbildungen  $\varphi(P)$  sind die sog. *Lift-Up*-Transformation, die die Punkte  $P$  des  $\mathbb{R}^m$  auf ein Paraboloid des  $\mathbb{R}^{m+1}$  abbildet und die sog. *Inversion*-Transformation, welche die Punkte einer Hyperebene des  $\mathbb{R}^m$  auf eine Hyperkugel des  $\mathbb{R}^{m+1}$  abbildet (siehe [Bro79], [Ede87] und [Buc88]).

Der Algorithmus läßt sich dann wie folgt kurz beschreiben:

ALGORITHMUS 3.2.27. *Convex-Hull-Methode*

1. *Bilde die Punktmenge  $P$  mit Hilfe der Abbildung  $\mathbb{R}^m \rightarrow \mathbb{R}^{m+1} : \varphi(P) \rightarrow \tilde{P}$  auf ein konvexes Gebiet des  $\mathbb{R}^{m+1}$  ab.*
2. *Bestimme  $\mathcal{V}(\tilde{P})$*
3. *Bilde ausgezeichnete  $(m - 1)$ -Flächen der konvexen Hülle von  $\mathcal{V}(\tilde{P})$  mit Hilfe der inversen Transformation  $\varphi^{-1}(\tilde{P})$  in den  $\mathbb{R}^m$  ab. Wir erhalten somit  $\mathcal{V}(P)$  (siehe Abbildung (3.2.4)).*

Diese Form der Algorithmen ist besonders beliebt, um die Komplexität der Berechnung des Voronoidiagramms abzuschätzen. Diese liefert im  $m$ -dimensionalen Fall eine Zeitkomplexität von  $O(n^{m/2+1})$ , was auch als Anhaltspunkt für andere Algorithmen genommen werden kann (siehe [OBS92]).

Ein weiterer Ansatz zur Bestimmung des Voronoidiagramms in  $m$  Dimensionen ist die sog. *Divide-And-Conquer*-Methode, die erstmals von Preparata in [PS88] vorgestellt wurde. Diese Methode stellt gleichzeitig ein grundlegendes Paradigma für effiziente Algorithmen dar: Zerlege rekursiv das Ausgangsproblem in möglichst gleichgroße Unterprobleme. Die Lösung des Gesamtproblems ergibt sich dann durch Zusammenführen (*mergen*) der Lösungen für die Unterprobleme. Somit läßt sich der Algorithmus von Preparata und Shamos kurz wie folgt beschreiben:

#### ALGORITHMUS 3.2.28. Divide-And-Conquer-Methode

1. *Unterteile die Punktmenge  $P$  rekursiv in Untermengen von jeweils gleicher bzw. ähnlicher Größe*
2. *Berechne die Voronoidiagramme der Untermengen*
3. *Füge die einzelnen Voronoidiagramme zu einem gemeinsamen Diagramm zusammen*

Im letzten Schritt dieses Algorithmus liegt die Schwierigkeit: Wie kann man effizient  $m$ -dimensionale Voronoidiagramme zusammenfügen? Eine Möglichkeit wird in [OBS92] vorgestellt. Dieser Algorithmus ist zwar in der Theorie zeitoptimal, was in der praktischen Umsetzung jedoch nicht leicht zu erreichen ist, da der Overhead zur Verwaltung der benötigten Datenstruktur sehr groß ist.

3.2.4.2. *Der inkrementelle Ansatz.* Vorgestellt wurde die inkrementelle Methode erstmals von Watson ([Wat81]) und Bowyer ([Bow81]). Die Grundidee des inkrementellen Ansatzes ist einfach und in beiden Ansätzen gleich:

Unter der Annahme, daß schon eine gültige Voronoizerlegung (Delaunaytessellierung) der ersten  $k$  Punkte der Punktmenge  $P_k \subset P$  existiert, suchen wir einen ersten Polyeder (Bowyer) oder Tetraeder (Watson), den es zu modifizieren gilt. Danach wird der neue Punkt  $\mathbf{p}_{k+1}$  in das bestehende Voronoidiagramm (den bestehenden Delaunaygraphen) eingefügt. Es ist hier schon ersichtlich, daß sich die Vorgehensweise nur durch die Anwendung in den zueinander dualen Strukturen des Voronoidiagramms und des Delaunaygraphen unterscheidet.

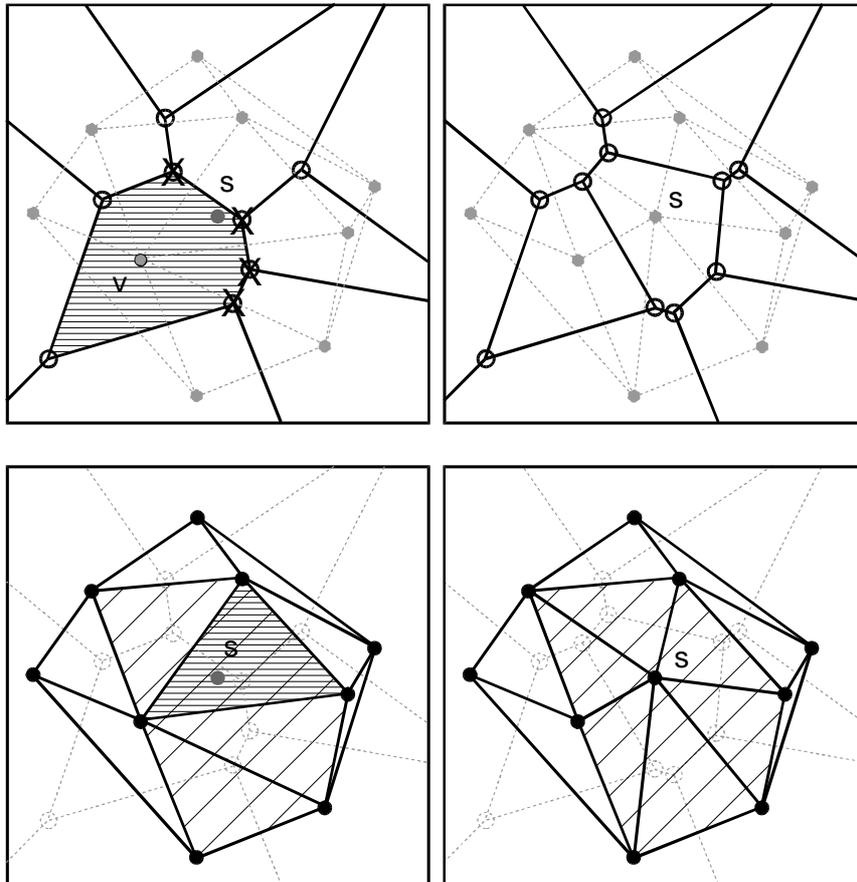


ABBILDUNG 3.2.5. Oben ist das inkrementelle Vorgehen nach Bowyer, unten das nach Watson beschrieben. Bei Bowyer sucht man ein initiales Polygon (schraffiertes Polygon), welches den Punkt  $s$  enthält. Bei diesem Polygon werden nun die Voronoiunkte unterdrückt (dargestellt durch ein  $x$ ), welche näher zu  $s$  sind als zum korrespondierenden Voronoiseed  $v$ . Es entsteht ein neues Polygon (siehe oben rechts). Bei Watson (untere Reihe) suchen wir im Delaunaygraphen dasjenige Dreieck, in dem  $s$  enthalten ist (dicht schraffiertes Dreieck). Danach wird das Gebiet um die Dreiecke erweitert, in deren Umkreis  $S(T)$  der Punkt  $s$  enthalten ist (weit schraffiertes Gebiet). Dieses Sterngebiet wird neu trianguliert (siehe unten rechts im schraffierten Bereich). Die jeweils duale Struktur ist hellgrau der Zeichnung unterlegt.

Beide Methoden sind daher als gleichwertig zu betrachten. Im weiteren wollen wir kurz die Vorgehensweise nach Watson beschreiben. Eine detaillierte Beschreibung der Vorgehensweise nach Bowyer findet man in [6].

ALGORITHMUS 3.2.29. *Berechnung des PVD nach Watson*

1. Es existiere  $\mathcal{D}(\tilde{P})$ , wobei die Punktmenge  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$  des  $\mathbb{R}^m$  im Raum der bestehenden Delaunaytessellierung von  $\mathcal{D}(\tilde{P})$  enthalten sind. Setze  $\mathbf{p}_s = \mathbf{p}_1$ .
2. Suche denjenigen Simplex  $T_s$ , der  $\mathbf{p}_s$  enthält.
3. Suche alle Simplexe  $T_{u_j}$ , die  $\mathbf{p}_s$  in der umschreibenden Hyperkugel  $S(T_{u_j})$  enthalten, also gilt  $S(T_{u_j}) \cap \mathbf{p}_s \neq \emptyset$ . Die Menge aller Simplexe  $S_s := \cup_j T_{u_j}$  formt nun ein Sterngebiet.
4. „Tetraedisiere“ das Sterngebiet  $S_s$  neu.
5. Ist  $s < n$ , setze  $s = s + 1$ . Gehe zu 2.

Bei der Suche nach dem Tetraeder  $T_s$  im obigen Algorithmus macht man sich die Tatsache zunutze, daß ein Tetraeder ein konvexes Polygon formt (siehe Definition 3.2.5). Somit kann man den Tetraeder  $T_s$  als Schnittmenge der  $m + 1$  Halbräume, die sich durch die  $(m - 1)$ -Flächen des Simplexes und den jeweils gegenüberliegenden Punkt ergeben, auffassen. Nutzt man die Dualität zwischen  $(m - 1)$ -Fläche und Voronoikante aus, so kann man die Suche im Voronoidiagramm wie folgt beschreiben:

ALGORITHMUS 3.2.30. *Suche nach dem inneren Simplex*

Es sei gegeben  $\mathcal{D}(P)$  (respektive  $\mathcal{V}(P)$ ) und ein Punkt  $\mathbf{s}$ , der in der Tessellierung  $\mathcal{D}(P)$  enthalten ist. Weiterhin bezeichne  $\mathbf{v}_T \in \mathcal{V}(P)$  den zum Delaunaysimplex  $T$  dualen Voronoipunkt und  $N_T = \{\mathbf{v}_T^1, \dots, \mathbf{v}_T^{m+1}\}$  die Menge der  $m + 1$  Nachbarn von  $\mathbf{v}_T$ . Wir definieren nun den Halbraum  $H(\mathbf{v}_T, \mathbf{v}_T^i)$  durch die Normalenform wie folgt:

$$H(\mathbf{v}_T, \mathbf{v}_T^i) := \left\{ \mathbf{x} \mid \left( \mathbf{x} - \frac{\mathbf{v}_T + \mathbf{v}_T^i}{2} \right) \cdot \mathbf{n} \geq 0 \text{ mit } \mathbf{n} = \mathbf{v}_T - \mathbf{v}_T^i \right\}$$

Mit diesen Definitionen und Annahmen läßt sich nun die Suche wie folgt beschreiben:

1. Starte mit einem Simplex  $T$  von  $\mathcal{D}(P)$
2. Existiert nun ein  $j$ , für das  $H(\mathbf{v}_T, \mathbf{v}_T^j) \cap \mathbf{s} = \emptyset$  gilt, dann gehe zu Schritt 3  
ansonsten  $\Rightarrow T$  ist der gesuchte Simplex
3. Setze  $T = T(\mathbf{v}_T^j)$  und wiederhole Schritt 2

Der letzte Tetraedisierungsschritt des Sterngebietes im Algorithmus von Watson gestaltet sich einfach: Das Sterngebiet formt ein Polyeder, dessen  $(m - 1)$ -Flächen (in diesem Falle  $(m - 1)$ -Simplexe) an  $\mathcal{D}(P) \setminus S_s$  angrenzen. Jeder dieser  $(m - 1)$ -Simplexe stellt nun eine Seite eines neuen Simplex dar, der aus den Eckpunkten des  $(m - 1)$ -Simplex und dem Sternpunkt selbst geformt wird (siehe Abbildung (3.2.5)).

### 3.3. 3D Beschreibung des Boissonat'schen Ansatzes

Der Rekonstruktionsalgorithmus, der in der vorliegenden Arbeit der Rekonstruktion der dreidimensionalen Struktur ausgehend von Konturlinien (Polygonzügen) zugrunde gelegt wurde, bedient sich der Methoden und Theorien aus der Graphentheorie. Speziell wird hier das Voronoidiagramm und dessen dualer Graph — der Delaunaygraph — benutzt, um das rekonstruierte Objekt durch Tetraeder und somit die Oberfläche desselben durch Dreiecke darzustellen. Eine derartige Zerlegung des Objektes bringt mehrere Vorteile mit sich:

- Das Dreieck ist in der Computergraphik die kleinste Flächeneinheit, die zur Visualisierung von geometrischen Objekten benutzt wird. So werden kompliziertere Oberflächen von geometrischen Objekten stets in diese kleinste Einheit zerlegt.
- Von Dreiecken kann man sehr einfach den Flächeninhalt bestimmen. Somit ist eine einfache und analytische Berechnung der Oberfläche des generierten Objektes möglich.
- Weiterhin ist das Volumen von Tetraedern analytisch bestimmbar<sup>5</sup>. Daher ist auch das Volumen des Objektes einfach und analytisch bestimmbar.

Die Delaunay-Tetraedisierung zerlegt das Objekt in sich nicht durchdringende Tetraeder, die dicht liegen. Sie stellt somit — wie wir schon in den vorigen Abschnitten sehen konnten — eine Tessellierung des Objektes dar. Durch die einfache Gewinnung von morphologischen und topologischen Größen, wie Volumen und Oberfläche der Objekte, kann auf diese Weise eine große Anzahl von biologisch und physikalisch relevanten Daten gewonnen werden (siehe Kapitel 5).

**3.3.1. Grundvoraussetzungen zur Bestimmung der Nachbarschaftsgraphen.** Um die Tetraedisierung des Objektes zu ermöglichen, bestimmen wir zu den Punkten auf den vorgegebenen Konturlinien das Voronoidiagramm und den dazu dualen Graphen, den Delaunaygraph. Eine Kontur ist hierbei definiert als ein geschlossener und ebener Polygonzug, der eine vorgegebene Struktur<sup>6</sup> umrandet (siehe Abbildung (3.3.1)). Dabei kann diese Struktur durchaus auch Löcher haben oder sich verzweigen (siehe Abbildung (3.3.6) auf Seite 70). Auf einer derartigen Konturlinie kennt man die sog. *Konturnachbarschaft*, welche besagt, daß innerhalb einer Kontur die Reihenfolge eine bidirektionale Nachbarschaft vorgibt<sup>7</sup>.

Möchte man nun so definierte Konturlinien zu einem dreidimensionalen Objekt zusammenfügen und durch Delaunaytetraeder approximieren, so muß man berücksichtigen, daß Konturnachbarn auch Kanten

---

<sup>5</sup>siehe Heron'sche Formel

<sup>6</sup>in unserem Falle: eine biologische Struktur

<sup>7</sup>dies ist, wie vorher schon diskutiert, nur in 2 Dimensionen möglich

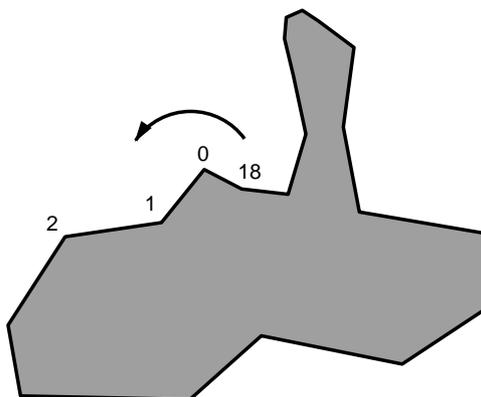


ABBILDUNG 3.3.1. Die Abbildung zeigt, wie eine einfache, geschlossene Kontur definiert ist. Es besteht innerhalb einer geschlossenen Kontur eine bidirektionale Nachbarschaft, die durch die Numerierung der Konturpunkte entsteht.

im Delaunaygraphen formen müssen. Ansonsten ist eine Approximation der Objektquerschnitte durch Delaunaytetraeder nicht möglich. Dies führt uns zu folgendem Kriterium:

**KRITERIUM 3.3.1.** *Das Nachbarschafts- oder Contour-Containment-Kriterium*

*Alle Segmente einer Kontur  $C \subset P$  müssen als Kanten im Delaunaygraphen  $\mathcal{D}(P)$  enthalten sein. Somit müssen Konturnachbarn auch Nachbarn im entsprechenden Delaunaygraphen sein.*

Im Abschnitt 3.5.1 beschäftigen wir uns ausführlich damit, wie man prüfen kann, ob diese Voraussetzungen gegeben sind und wie man diese im Notfall erzwingen kann.

**3.3.2. Klassifizierung der Tetraeder.** Unter der Annahme, daß das Nachbarschaftskriterium, wie im vorigen Abschnitt erläutert, erfüllt sei, teilt man die Menge der Tetraeder in folgende vier Klassen ein (siehe auch Abbildung (3.3.2)):

1. Tetraeder, bei denen 3 Punkte auf einer Kontur liegen.
2. Tetraeder, bei denen jeweils 2 Punkte auf einer Kontur liegen.
3. Tetraeder, bei denen zwei Punkte auf einer Kontur liegen und die restlichen beiden Punkte auf verschiedenen Konturen.
4. Tetraeder, bei denen alle 4 Punkte auf verschiedenen Konturen liegen.

Nun muß man — hat man die Delaunaytetraedisierung schon ausgerechnet — die vorhandenen Tetraeder zwei verschiedenen Gruppen zuzuordnen: den *inneren* oder *äußeren* Tetraedern. Diesen Vorgang nennen wir Binarisierung der Delaunaytessellierung. Hierzu werden die oben klassifizierten Tetraeder wie folgt untersucht und gelabelt:

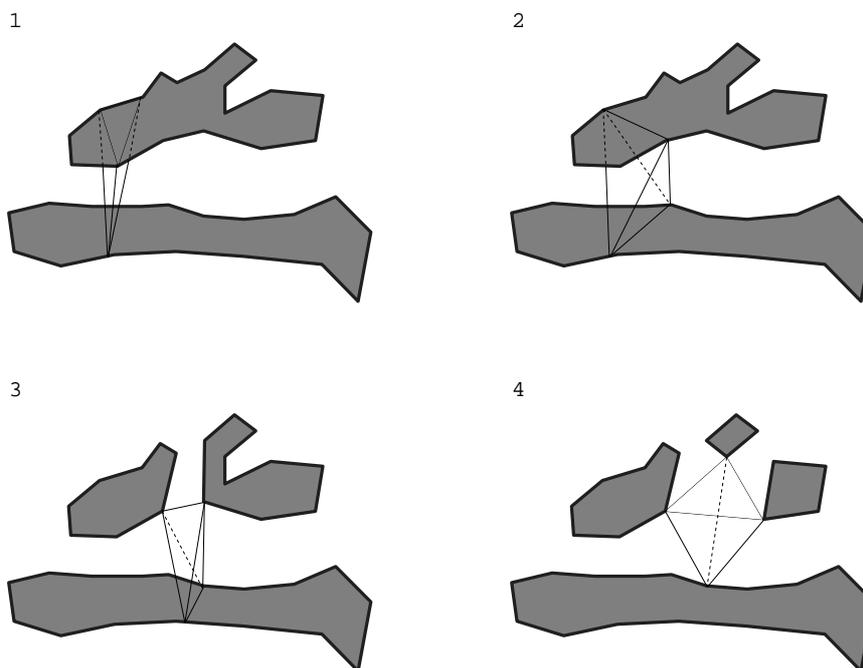


ABBILDUNG 3.3.2. In dieser Abbildung ist die Einteilung der Delaunaytetraeder in 4 Klassen schematisch aufgezeichnet.

1. Liegt diejenige Dreiecksfläche, die in einer Ebene liegt, innerhalb der umrandeten Kontur, so wird der Tetraeder als *Innen* gelabelt.
2. Die jeweils paarweise in einer Kontur liegenden Punkte bilden eine gemeinsame Kante. Man muß nun zwei Fälle unterscheiden:
  - (a) Die beiden Punkte sind paarweise benachbart, d.h. die eingeschlossene Kante ist in der Kontur enthalten. Diese Tetraeder werden als unbestimmt gelabelt und später iterativ zugeordnet (siehe unten).
  - (b) Liegt einer der zwei entstehenden Kanten eindeutig innerhalb oder außerhalb der Konturflächen, so wird der Tetraeder dementsprechend eindeutig als *Innen* bzw. *Außen* gelabelt, wobei das *Außen*-Kriterium über das *Innen*-Kriterium Vorrang hat.
3. Tetraeder vom Typ 3 werden eindeutig als *Außen* gelabelt.
4. Tetraeder vom Typ 4 werden ebenfalls eindeutig als *Außen* gelabelt.

Die Uneindeutigkeit von Fall 2 wird durch einen iterativen Prozeß kompensiert: Es wird die Menge der *uneindeutigen* Tetraeder solange iterativ durchlaufen, bis man keinen Tetraeder mehr findet, der zu zwei *inneren* Tetraedern benachbart ist, d.h. mit diesen eine gemeinsame

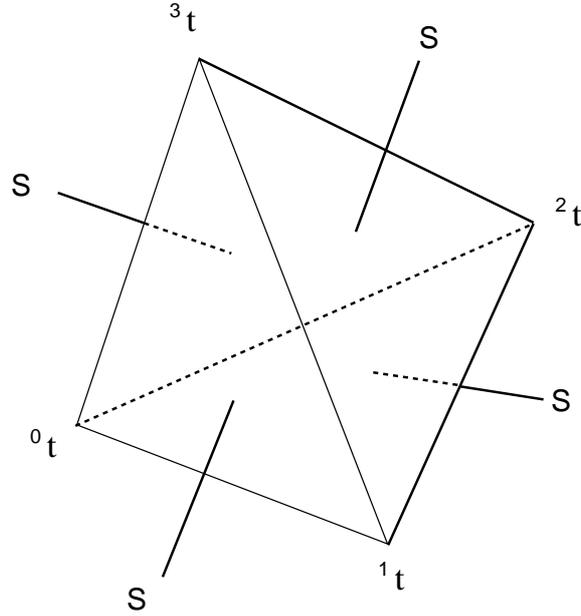


ABBILDUNG 3.3.3. Schematische Zeichnung eines Tetraeders: Benennung der Eckpunkte und Seiten.

Berührungsfläche bildet. Nach Abschluß der Iteration werden alle verbleibenden, unbestimmbaren Tetraeder als *Außen* gelabelt.

**3.3.3. Mathematische Beschreibung.** Betrachten wir die Menge  $\mathcal{T}$  aller Delaunaytetraeder der Delaunaytessellierung  $\mathcal{D}(P)$  zur gegebenen Punktmenge  $P$ . Sei weiterhin  $t$  ein Element in dieser Menge, so bezeichnen wir die vier Eckpunkte des Tetraeders mit  ${}^0t, {}^1t, {}^2t, {}^3t$  (siehe Abbildung (3.3.3)).  $\mathcal{C}$  beschreibt die Menge aller Konturen, während  $C_i \in \mathcal{C}$  das von der Kontur  $\partial C_i$  umrandete Gebiet bezeichnet. Die Konturpunkte, die zum Rand einer Kontur gehören, werden weiterhin als  $\mathbf{p}_m^i$  bezeichnet, wobei  $m$  die Punktnummer angibt. Gehört der Punkt  $\mathbf{p}_m^i$  zu Tetraeder  $t$ , so bezeichnen wir diesen Punkt als  ${}^t\mathbf{p}_m^i$ , wobei dieser Punkt einem der vier Eckpunkte der Menge  $\{{}^0t, {}^1t, {}^2t, {}^3t\}$  entspricht.

Wir unterscheiden vier Klassen von Tetraeder:

1.  $\mathcal{T}_1 := \{t \in \mathcal{T} \mid {}^t\mathbf{p}_a^i, {}^t\mathbf{p}_b^i, {}^t\mathbf{p}_c^i, {}^t\mathbf{p}_d^j, i \neq j\}$
2.  $\mathcal{T}_2 := \{t \in \mathcal{T} \mid {}^t\mathbf{p}_a^i, {}^t\mathbf{p}_b^i, {}^t\mathbf{p}_c^j, {}^t\mathbf{p}_d^j, i \neq j\}$
3.  $\mathcal{T}_3 := \{t \in \mathcal{T} \mid {}^t\mathbf{p}_a^i, {}^t\mathbf{p}_b^i, {}^t\mathbf{p}_c^j, {}^t\mathbf{p}_d^k, i \neq j, j \neq k, i \neq k\}$
4.  $\mathcal{T}_4 := \{t \in \mathcal{T} \mid {}^t\mathbf{p}_a^i, {}^t\mathbf{p}_b^j, {}^t\mathbf{p}_c^k, {}^t\mathbf{p}_d^l, \{i, j, k, l\} \text{ paarweise verschieden}\}$

Ausgehend von diesen vier Klassen müssen wir nun ein Entscheidungskriterium finden, das eine Aussage darüber erlaubt, ob ein Tetraeder zum Objekt gehört oder nicht, d.h. ob er als *innerer* oder *äußerer* Tetraeder gelabelt werden kann. Dazu werden die Punkte betrachtet, welche zu einer Kontur gehören.

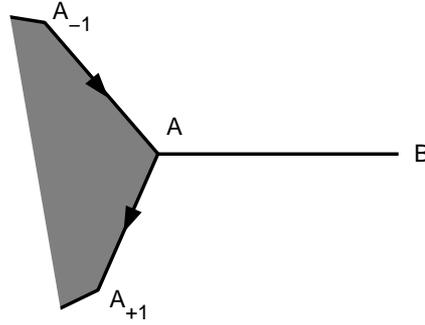


ABBILDUNG 3.3.4. Skizze zum Kantenkriterium: Man entscheidet lokal anhand der Konturorientierung, die durch die Nachbarn  $\mathbf{A}_{-1}$  und  $\mathbf{A}_{+1}$  gegeben ist, ob die Kante  $\kappa(\mathbf{A}, \mathbf{B})$  innerhalb oder außerhalb der Kontur verläuft.

Bezeichnet  $\Delta(\mathbf{A}, \mathbf{B}, \mathbf{C})$  das Dreieck, welches von den Punkten  $\mathbf{A}$ ,  $\mathbf{B}$  und  $\mathbf{C}$  gebildet wird, und  $\kappa(\mathbf{A}, \mathbf{B})$  die Verbindung oder Kante zwischen  $\mathbf{A}$  und  $\mathbf{B}$ , so kommen wir zu folgender Fallunterscheidung:

1. Fall: Ist  $t \in \mathcal{T}_1 \wedge \Delta({}^t\mathbf{p}_a, {}^t\mathbf{p}_b, {}^t\mathbf{p}_c) \subset C_i$ , so gilt der Tetraeder als *innerer* Tetraeder.
2. Fall: Ist  $t \in \mathcal{T}_2$  und gilt weiterhin:
  - (a) Ist  $\kappa({}^t\mathbf{p}_a, {}^t\mathbf{p}_b) \cap C_i \setminus \partial C_i \neq \emptyset$  und  $\kappa({}^t\mathbf{p}_a, {}^t\mathbf{p}_b) \cap C_j \setminus \partial C_j \neq \emptyset$ , so gilt der Tetraeder als *innerer* Tetraeder.
  - (b) Ist  $\kappa({}^t\mathbf{p}_a, {}^t\mathbf{p}_b) \cap C_i \setminus \{{}^t\mathbf{p}_a, {}^t\mathbf{p}_b\} = \emptyset$  oder  $\kappa({}^t\mathbf{p}_a, {}^t\mathbf{p}_b) \cap C_j \setminus \{{}^t\mathbf{p}_a, {}^t\mathbf{p}_b\} = \emptyset$ , so gilt der Tetraeder als *äußerer* Tetraeder.
  - (c) In allen anderen Fällen ( $\kappa({}^t\mathbf{p}_a, {}^t\mathbf{p}_b) \subset \partial C_i \vee \kappa({}^t\mathbf{p}_a, {}^t\mathbf{p}_b) \subset \partial C_j$ ) gilt der Tetraeder als unbestimmt.
3. Fall: Ist  $t \in \mathcal{T}_3$  so gilt der Tetraeder als *äußerer* Tetraeder.
4. Fall: Ist  $t \in \mathcal{T}_4$  so gilt der Tetraeder als *äußerer* Tetraeder.

Somit läuft die Binarisierung des Tetraederraumes auf eine Innen–Außen–Entscheidung von Kanten auf einer Konturlinie heraus, denn auch Fall 1 läßt sich anhand einer exemplarisch ausgewählten Seite lokal entscheiden. Dies liegt vornehmlich an der Eigenschaft des Delaunaygraphen, daß er eine Tessellierung des Objektraumes bildet.

3.3.3.1. *Das Kantenkriterium.* Betrachten wir die Kante  $\kappa(\mathbf{A}, \mathbf{B})$  in Skizze (3.3.4), wobei sich durch die Anordnung der Punkte, d.h. durch den Umlaufsinn der Punkte auf der Kontur, *Innen-* und *Außenseite* der Kontur festlegen lassen. Hierzu betrachten wir das Dreieck, welches von  $\kappa(\mathbf{A}, \mathbf{B})$ ,  $\overline{\mathbf{A}_{-1}\mathbf{A}}$  (respektive  $\overline{\mathbf{A}\mathbf{A}_{+1}}$ ) und der positiven z–Achse bestimmt wird.

Wir unterscheiden die Fälle:

1. Ist  $\mathbf{A}_{-1} = \mathbf{B} \vee \mathbf{A}_{+1} = \mathbf{B}$ , d.h.  $\kappa(\mathbf{A}_{-1}, \mathbf{A}) = \kappa(\mathbf{B}, \mathbf{A}) \vee \kappa(\mathbf{A}, \mathbf{A}_{+1}) = \kappa(\mathbf{A}, \mathbf{B})$ , so ist keine Entscheidung möglich.

2. Ist  $\kappa(\mathbf{A}_{-1}, \mathbf{A}) \neq \kappa(\mathbf{B}, \mathbf{A}) \wedge \kappa(\mathbf{A}, \mathbf{A}_{+1}) \neq \kappa(\mathbf{A}, \mathbf{B})$ , so betrachten wir die Kreuzprodukte  $\kappa(\mathbf{A}_{-1}, \mathbf{A}) \times \kappa(\mathbf{A}, \mathbf{A}_{+1})$ ,  $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}_{-1}, \mathbf{A})$  und  $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}, \mathbf{A}_{+1})$ . Wir unterscheiden:
- (a) Ist  $\kappa(\mathbf{A}_{-1}, \mathbf{A}) \times \kappa(\mathbf{A}, \mathbf{A}_{+1}) \leq 0$  und weiterhin:
    - (i)  $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}_{-1}, \mathbf{A}) > 0$  oder  
 $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}, \mathbf{A}_{+1}) > 0 \Rightarrow \text{Innen}$
    - (ii)  $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}_{-1}, \mathbf{A}) \leq 0$  und  
 $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}, \mathbf{A}_{+1}) \leq 0 \Rightarrow \text{Außen}$
  - (b) Oder  $\kappa(\mathbf{A}_{-1}, \mathbf{A}) \times \kappa(\mathbf{A}, \mathbf{A}_{+1}) > 0$  und weiterhin:
    - (i)  $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}_{-1}, \mathbf{A}) > 0$  oder  
 $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}, \mathbf{A}_{+1}) > 0 \Rightarrow \text{Außen}$
    - (ii)  $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}_{-1}, \mathbf{A}) \leq 0$  und  
 $\kappa(\mathbf{A}, \mathbf{B}) \times \kappa(\mathbf{A}, \mathbf{A}_{+1}) \leq 0 \Rightarrow \text{Innen}$

Im Falle 1 des Kantenkriteriums, bei dem keine Entscheidung möglich ist, muß bei der Dreiecksentscheidung eine Seite gesucht werden, die *nicht* von Konturnachbarn geformt wird. Kann eine solche Seite nicht gefunden werden, besteht die Kontur lediglich aus den vorgegebenen drei Punkten, bildet also ein einziges Dreieck. Dieses ist somit als *Innen* zu bewerten. Bei der Zweierentscheidung 2 hingegen muß die Uneindeutigkeit durch einen iterativen Algorithmus beseitigt werden, der oben kurz skizziert wurde und im nächsten Abschnitt näher erläutert werden soll.

3.3.3.2. *Der Füllalgorithmus.* Nachdem wir die Tetraeder nach dem bisher beschriebenen Vorgehen in die drei Kategorien *Innen*, *Außen* und *Unbestimmt* überführt haben, müssen die *unbestimmten* Tetraeder noch binarisiert werden. Hierzu betrachten wir den zu einem Tetraeder gehörenden, dualen Voronoipunkt  $\mathbf{v}$ . Sind mindestens zwei seiner Voronoinachbarn als *Innen* gelabelt worden, so markieren wir den zu  $\mathbf{v}$  gehörenden Tetraeder mit *Innen*. Wir durchlaufen nun die ganze Voronoistruktur solange iterativ, bis kein weiterer Voronoipunkt mehr gelabelt werden kann. Die verbleibenden, *unbestimmten* Tetraeder werden nun als *Außen* markiert.

Formalisiert kann man den Algorithmus wie folgt beschreiben:

ALGORITHMUS 3.3.2. *Der Füllalgorithmus*

Bezeichne  $V = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$  die Menge aller Voronoipunkte des Voronoidiagramms  $\mathcal{V}(P)$  und  $T(\mathbf{v}_i)$  den zugehörigen Delaunaytetraeder. Die 4 Nachbarn des Voronoipunktes  $\mathbf{v}_i$  bezeichnen wir als  $\mathbf{n}_j^i \in V$ , wobei  $i \in I_n$  und  $j \in I_4$  ist. Weiterhin sei die Funktion  $\theta(\mathbf{v}_i, j)$  gegeben durch:

$$\theta(\mathbf{v}_i, j) := \begin{cases} 0 & \text{falls } \mathbf{n}_j^i \text{ Außen} \\ 1 & \text{falls } \mathbf{n}_j^i \text{ Innen} \end{cases}$$

Dann wird der Algorithmus wie folgt durchlaufen:

1. Überprüfe  $\forall i \in I_n$ , ob  $\mathbf{v}_i$  unbestimmt ist.  
Gilt außerdem  $\sum_j \theta(\mathbf{v}_i, j) \geq 2$ , dann setze den IO-Label von  $\mathbf{v}_i$  auf Innen und erhöhe den Zähler  $n_{\text{changed}}$  um 1.
2. Ist  $n_{\text{changed}} > 0$ , dann setze  $n_{\text{changed}} = 0$  und gehe zu Schritt 1 ansonsten  $\Rightarrow$  **Ende**

3.3.3.3. *Die Analogie zum Boissonat'schen Ansatz.* Die hier vorgestellte dreidimensionale Beschreibung bildet auch die Basis des in der Arbeit angewandten Rekonstruktionsalgorithmus. Dieser Ansatz ist analog zu dem von Boissonat in [Boi88] 1988 erstmals publizierten Algorithmus und dessen Erweiterung von Boissonat und Geiger (siehe [BG92a]) von 1992. Die dreidimensionale Beschreibung und der Boissonat'sche Ansatz weisen jedoch im Detail Unterschiede auf, auf welche im folgenden kurz eingegangen werden soll.

Wird in der vorliegenden Arbeit davon ausgegangen, daß die dreidimensionale Delaunaytessellierung der Punkte der Konturen  $C_1$  und  $C_2$ , die in den verschiedenen Ebenen  $E_1$  und  $E_2$  liegen, vorliegt, beschreibt Boissonat eindrucksvoll, wie die dreidimensionale Delaunaytessellierung der beiden Ebenen durch „Verschneiden“ der beiden zweidimensionalen Voronoidiagramme der einzelnen Konturen konstruiert werden kann. Dies ist besonders im Hinblick auf die Rechenzeit und die numerische Stabilität geschickt und sehr elegant.

In der dreidimensionalen Tessellierung der beiden ebenen Konturen  $C_1$  und  $C_2$  existieren drei verschiedene Formen von Tetraedern:

- $T_1$  sind dabei Tetraeder, mit einer Grundfläche in  $E_1$
- $T_2$  sind Tetraeder mit einer Grundfläche in  $E_2$
- $T_{12}$  sind Tetraeder, bei den jeweils eine Kante in  $E_1$  und  $E_2$  liegt

Liegen uns die beiden zweidimensionalen Voronoidiagramme  $\mathcal{V}(C_1)$  und  $\mathcal{V}(C_2)$  vor, so erzeugen wir durch  $\mathcal{B}(C_1, C_2) := \mathcal{V}(C_1) \cap \mathcal{V}(C_2)$  einen neuen Graphen. Knotenpunkte im neuen, planaren Graphen, die aus Knotenpunkte von  $\mathcal{V}(C_i)$  hervorgehen, repräsentieren hierbei Tetraeder vom Typ  $T_i$  und haben die Ordnung 3. Schnittpunkte zwischen den Kanten von  $\mathcal{V}(C_1)$  und  $\mathcal{V}(C_2)$  entsprechen den Tetraedern vom Typ  $T_{12}$  und haben die Ordnung 4.

Durch die Dualität der Seite eines Voronoipolygons zu einer Delaunaykante sind die Eckpunkte der Tetraeder  $T_{12}$  festgelegt. Um die Tetraeder vom Typ  $T_i$  zu erzeugen, muß lediglich noch der zum entsprechenden Voronoikantenpunkt  $\mathbf{v}_i \in \mathcal{V}(C_i)$  nächste Punkt der Kontur  $C_j$  mit  $j \neq i$  gefunden werden. In einem abschließenden Schritt kann nun der Delaunaynachbarschaftsgraph mit Hilfe von  $\mathcal{B}(C_1, C_2)$  aufgebaut werden.

Ein weiterer Unterschied zwischen dem hier vorgestellten und dem Boissonat'schen Ansatz besteht in der Art des *Füllalgorithmus*. Bei Boissonat wird nach einem *soliden* Pfad im Graphen  $\mathcal{B}(C_1, C_2)$  gesucht, der zwei als *Innen* gelabelte Tetraeder vom Typ  $T_i$  verbindet. Alle

IO-Label	$t_3$	$t_2$	$t_1$	$t_0$	Sum-Label	$d_3$	$d_2$	$d_1$	$d_0$
----------	-------	-------	-------	-------	-----------	-------	-------	-------	-------

ABBILDUNG 3.3.5. Aufteilung von einem Bitfeld der Länge 32 in 4 Felder der Länge 8 zur Eindeutigen Labelung der Voronoikanten. Der Dreieckslabel ( $d_i$ ) und der Zeichenlabel ( $t_i$ ) bestehen aus jeweils 2 bit.

Tetraeder vom Typ  $T_{12}$ , die auf diesem Pfad liegen, werden ebenfalls als *Innen* gelabelt.

Für eine Vertiefung des hier vorgestellten und einen Beweis des hier gezeigten sei die Lektüre von [Boi88] empfohlen.

**3.3.4. Programmiertechnische Umsetzung.** Zur Umsetzung des oben dargestellten Konzeptes ist es im Programm erforderlich, verschiedene *Labels* zur Verfügung zu stellen. In diesem Fall müssen wir die Voronoikantenstruktur labeln. Da ein Tetraeder aus vier Eckpunkten besteht, kann in diesem Falle sehr schön mit Bitfeldern gearbeitet werden.

Im implementierten Algorithmus wurden die Labels analog zur Skizze (3.3.5) eingeführt. Die Labels haben folgende Bedeutung:

- Im *Dreieckslabel* wird gezählt, zu wievielen Konturen ein Dreieck aus einem Tetraeder gehört.
- Das *Dreieckssummenlabel* ist definiert als  $S := \sum_i d_i$ . Aus  $S$  lassen sich die Tetraeder eindeutig in die oben vorgestellten Klassen einteilen:

1.  $S = 7 \Rightarrow$  Fall 1
2.  $S = 8 \Rightarrow$  Fall 2
3.  $S = 10 \Rightarrow$  Fall 3
4.  $S = 12 \Rightarrow$  Fall 4

- Das *In-Out-Label* gibt an, ob ein Tetraeder zum Objekt gehört oder nicht (0 oder 1). Uneindeutigkeitsfälle werden mit einem weiteren Label versehen (2).
- Als letztes Label wird das sog. *Zeichenlabel* definiert. Es gibt an, ob ein Dreieck an der Oberfläche des Objektes liegt oder nicht, d.h. ob es bei der graphischen Darstellung gezeichnet werden muß oder nicht.

Durch diese Einteilung ist es möglich, die einzelnen Fälle zu identifizieren, und schnell — durch einfachen Bitabgleich — die Oberflächendreiecke zu extrahieren. Ebenfalls ist durch den In-Out-Label die Möglichkeit gegeben, das Volumen des tetraedisierten Körpers zu berechnen. Durch den Dreieckslabel selbst kann man unterscheiden, ob ein Dreieck zur Deckfläche oder zur Seitenfläche des Objektes gehört.

Die Labelung ist schnell und ermöglicht außerdem Zugriff auf topologisch wichtige Zusammenhänge. Zur Illustration sind im Anhang

C die Deklarationen der Header-Datei abgedruckt, wie sie der Rekonstruktionsbibliothek, die im Rahmen der vorliegenden Arbeit entwickelt wurde, entnommen wurden.

An dieser Stelle möchte ich mich ausdrücklich bei Dr. R. Eils, Dr. E. Bertin und Prof. Dr. J.-M. Chassery bedanken, bei denen ich die „Grundausbildung“ in Sachen Berechnung des PVD erhalten habe. Ihrem Originalcode, mit dem zahlreiche Applikationen in der Biophysik, Biomathematik und Bioinformatik publiziert wurden ([BMC92], [BPC93], [EDB<sup>+</sup>94], [4] und [EDB<sup>+</sup>96]), ist auch das Grundgerüst der vorgestellten Datenstruktur entnommen. Allerdings wurde der Code zur Berechnung des PVD komplett nach dem Ansatz von Watson umgeschrieben und im Hinblick auf numerische Stabilität und konsistentes Handling degenerierter Voronoikanten erweitert. Dank geht auch an Prof. Dr. B. Sakmann und Prof. Dr. M. Robert-Nicoud, die mir einen Kurzaufenthalt am *Institut Albert Bonniot* in Grenoble, Frankreich, ermöglicht haben, der für die Realisierung des beschriebenen Rekonstruktionsalgorithmus wesentlich war.

**3.3.5. Erweiterung auf hierarchisch angeordnete Konturen.** Das bisherige beschreibt den topologischen Fall, daß in einer Kontur keine weiteren Konturen verlaufen (siehe Abbildung (3.3.1)). Mit diesem Ansatz konnten topologische Änderungen des Objektes, welche von *einfachen Bifurkationen* herrührten, schon erfaßt werden. Betrachten wir jedoch den Fall, daß in einer Kontur weitere Konturen verlaufen können (siehe Abbildung (3.3.6)), die *Löcher* in einer Kontur definieren, so müssen wir die Klassifizierung der Tetraeder (siehe Punkte 1 bis 4 auf Seite 62) insofern abändern, als daß wir das Wort *Kontur* in das Wort *topologisch zusammenhängende Kontur* tauschen.

Als *topologisch zusammenhängende Konturen* werden Konturen bezeichnet, die in ein und derselben Schnittebene liegen und den Querschnitt ein und desselben Objektes definieren. Somit umfaßt die Menge  $C_i$  nun ein u.U. nicht zusammenhängendes Gebiet, was bedeutet, das die Kontur  $\partial C_i$  (der Rand des Gebietes  $C_i$ ) nicht durch eine zusammenhängende Kurve zu beschreiben ist (siehe Abbildung (3.3.6)).

Mit dieser Erweiterung werden die Fälle, die bisher nach den Kriterien 3 und 4 auf Seite 63 zu bewerten waren, den Kategorien 1 und 2 zugeordnet. Nach Anwendung des Kantenkriteriums können diese dennoch ebenso als eindeutig außen erkannt werden. Die neue Definition stellt somit eine echte Erweiterung der bisher durchgeführten Klassifizierung dar.

**3.3.6. Nicht-solide Verbindungen.** Da in diesem Algorithmus kein *a priori* Wissen über die Links der Objekte eingeht, kann es vorkommen, daß unerwünschte Verbindungen zwischen Konturen entstehen. Sind nun zwei verschiedene Konturen über einen Tetraeder verbunden, der lediglich eine Kante mit den Konturen oder gar nur einen

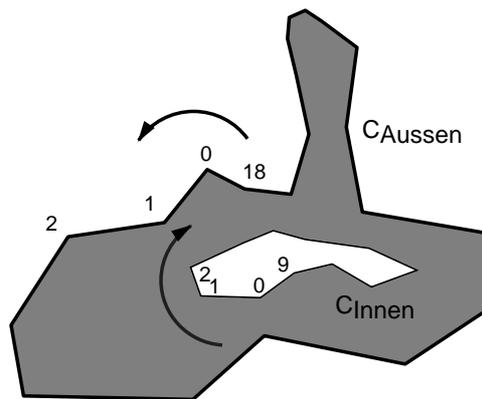


ABBILDUNG 3.3.6. Hier wird die Konturdefinition dahingehend erweitert, daß nun auch ineinandergeschachtelte Konturen durch den Rekonstruktionsalgorithmus erfaßt werden. Die durchgezogenen, schwarzen Linien definieren den Rand des Konturgebietes, welches sich in diesem Falle aus  $\partial C = \partial C_{\text{Innen}} + \partial C_{\text{Aussen}}$  zusammensetzt.

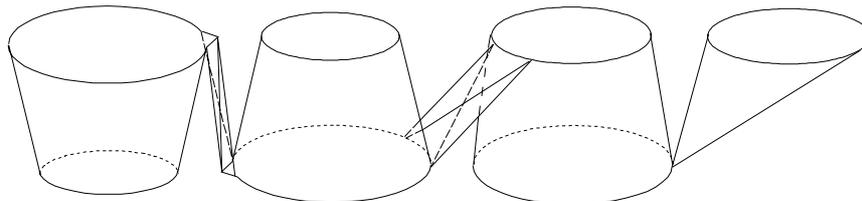


ABBILDUNG 3.3.7. Verschiedene Formen von *nicht-soliden* Verbindungen: Zwei Konturen sind nur über Tetraeder miteinander verbunden, die lediglich Kanten (Mitte) oder Punkte (rechts) mit einer der Konturen gemeinsam haben. Links sehen wir eine Verbindung, die aufgrund unbefriedigender Überlappung der Konturen und einer kleinen Störung der Konturlinie (siehe kleine Ausbuchtung links) zustande kommt, jedoch *nicht* eliminiert werden kann, da sie eine feste (*solide*) Verbindung nach Definition von Boissonat und Geiger darstellt.

Anknüpfungspunkt zu einer der Konturen hat, sprechen wir von einer *nicht-soliden* Verbindung (*Non-solid-Connection*). Beispiele sind in der Abbildung (3.3.7) zu sehen. Diese Verbindungen können durch die Nachbarschaft im Delaunaygraphen erkannt und eliminiert werden.

### 3.4. Interpolation von Zwischenschichten

Wie wir in den Abbildungen (3.4.1) und (3.4.2) sehen können, sind die vorgegebenen Konturpunkte nicht ausreichend, um die großen topologischen Veränderungen zwischen den beiden Konturen zu repräsentieren. Aus diesem Grund wurden in der Literatur einige Algorithmen

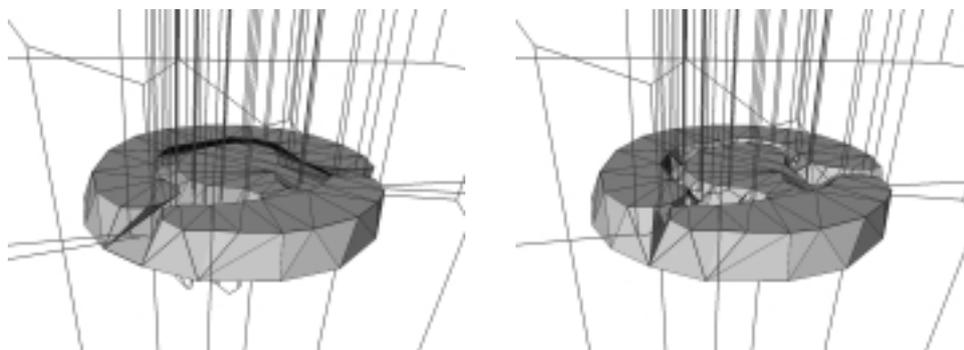


ABBILDUNG 3.4.1. Die hellgrauen Linien repräsentieren das 3D Außenskelett, welches noch die Objektoberfläche (siehe linke Seite) durchdringt. Nachdem nun interpolierende Punkte eingeführt wurden, durchdringt das Skelett die Oberfläche des Objektes nicht mehr. Ein innerer Kanal wird sichtbar (rechts).

vorgestellt, die sich mit der Interpolation von Zwischenschichten beschäftigen ([OPC96], [EPO91], [MSS92] und [CP94]). Aber auch der Boissonat'sche Ansatz, wie er 1988 zuerst publiziert wurde, wurde 1992 durch Boissonat und Geiger derart erweitert, daß auch größere topologische Änderungen erfaßbar wurden.

**3.4.1. Erweiterung des Boissonat'schen Ansatzes.** Um Rasche topologische Änderungen zwischen den Konturen erfassen zu können, haben Boissonat und Geiger 1992 angeregt, die Konturflächen, die von den Konturlinien umrandet werden (siehe Abbildungen (3.3.1) und (3.3.5)), durch sog. *innere Punkte* zu füllen. Dadurch wird es möglich, daß Oberflächenkanten oder Kanäle innerhalb der Konturen verlaufen, wodurch eine bessere Oberflächenrepräsentation des Objektes vor allem bei Bifurkationen und Löchern gewährleistet wird. Allerdings steigt auch die Anzahl der Tetraeder — und somit die Rechenzeit — u.U. enorm an, ohne daß das Einfügen zusätzlicher Punkte für die Objektrepräsentation notwendig gewesen wäre. Um ein adaptives Einfügen dieser Punkte vornehmen zu können, schlugen Boissonat und Geiger vor, das externe Skelett der jeweils gegenüberliegenden Kontur auf das Innere der Kontur selbst zu projizieren (siehe Abbildung (3.4.4)). Als Skelett bezeichnet man hierbei die *Innere Linie* (oder: *Medial Axis*) der Kontur. Diese kann im PVD durch die Verbindungslinien derjenigen Voronoikanten (allgemeiner:  $(m - 1)$ -Flächen) repräsentiert werden, die benachbarte Delaunaytetraeder (oder Simplexe), die zum Inneren des Objektes gehören, miteinander verbinden.

Die Schnittmenge zwischen den Konturflächen und den projizierten Skelettpunkten ergibt dann eine sinnvolle Auswahl an interpolierenden Punkten. Diese Punkte *müssen* beim Boissonat'schen Ansatz und der

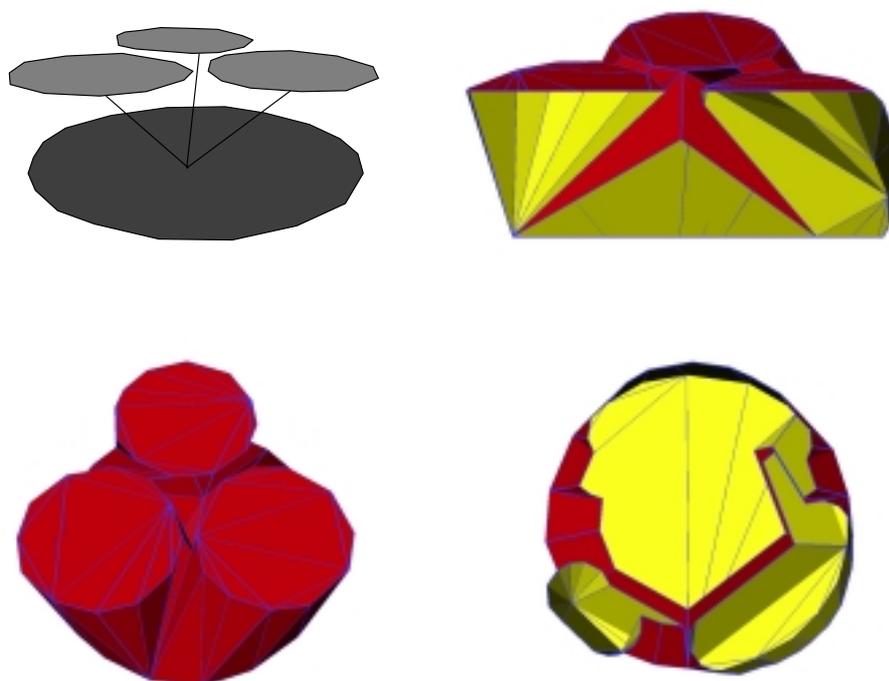


ABBILDUNG 3.4.2. Links oben sind die Eingangskonturdaten mit ihren Links zur unteren Kontur zu sehen. Es handelt sich um eine Mehrfachaufspaltung, die in der hier dargestellten Initialrekonstruktion nach Boissonat nicht in eine gute Oberflächerepräsentation überführt werden kann. Unten sind 2 Schnitte durch das 3D Objekt dargestellt, wobei rot die Außen- und gelb die Innenfläche repräsentiert.

hier eingeführten Erweiterung auf einer der Konturebenen liegen. Dies kann bei Bifurkationen dazu führen, daß die generierten Objekte im Sinne der Nachbarschaft der Delaunaytetraeder nicht miteinander verbunden sind. Somit kann eine vorher verbundene Konturfläche in mehrere Untergebiete zerfallen (siehe Abbildung (3.4.2)).

**3.4.2. Veränderungen gegenüber dem Ansatz von Boissonat und Geiger.** Um diese logische Inkonsistenz zu vermeiden (siehe Abbildung (3.4.2)), wird in der Literatur vorgeschlagen, sog. interpolierende Zwischenkonturen einzuführen (z.B. in [OPC96]). Diese Stellen dann eine Zwischenstufe der beteiligten Konturen dar. Diese Zwischenkonturen erhöhen aber gleichzeitig den Rechenaufwand wie auch die Anzahl der generierten Dreiecke. So wurde in [6] vorgeschlagen, die von Boissonat und Geiger vorgestellte Idee derart abzuwandeln, daß die Skelettpunkte auf eine Zwischenebene angehoben werden. Somit

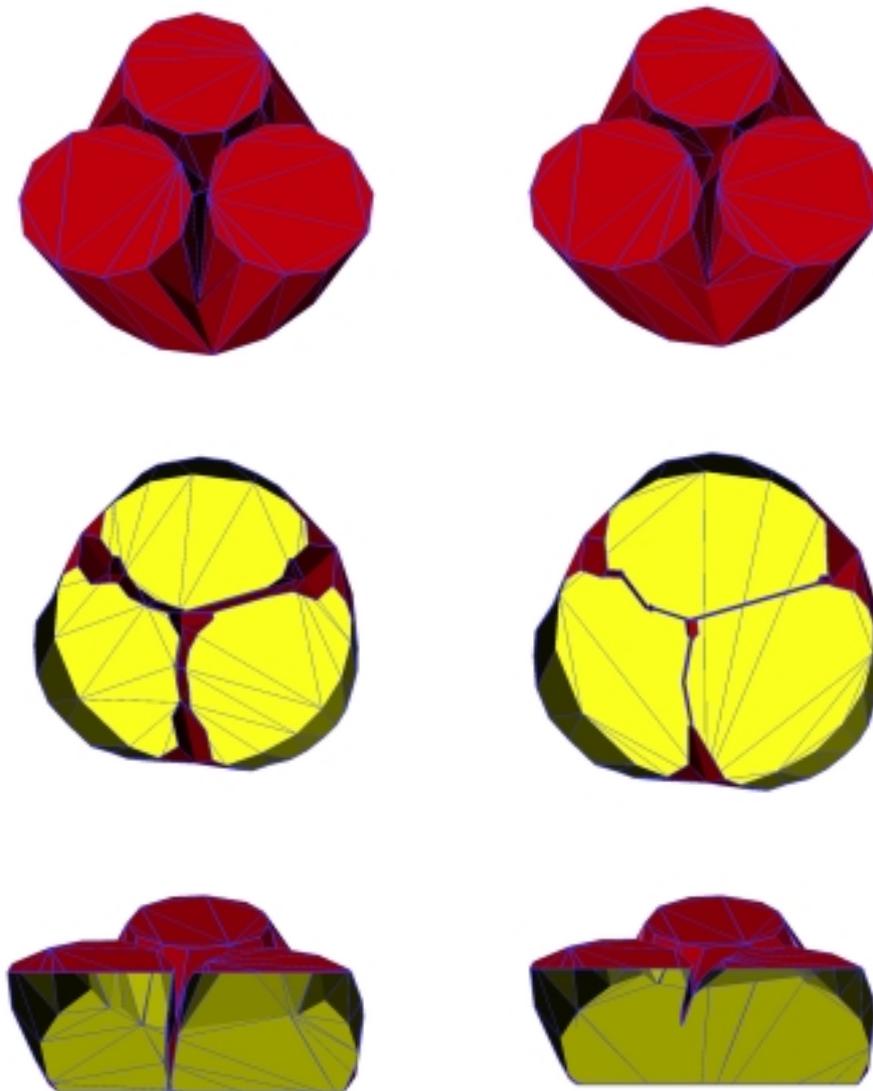


ABBILDUNG 3.4.3. In dieser Abbildung ist das Objekt von Abbildung (3.4.2) nach Einfügen von Zwischenpunkten dargestellt. Links ist das Vorgehen nach der Erweiterung von Boissonat und Geiger dargestellt, bei dem Zwischenpunkte in die untere Konturebene anhand des äußeren Skeletts der oberen Struktur eingefügt werden. Dabei zerfällt das Objekt in 3 disjunkte Regionen (siehe Schnitte durch das Objekt, links). Rechts ist dargestellt, wie die eingefügten Punkte auf eine interpolierende Zwischenschicht angehoben wurden, wie in [6] vorgeschlagen. Hier gibt es keine Abschnürungen innerhalb eines Objektes (siehe rechts unten).

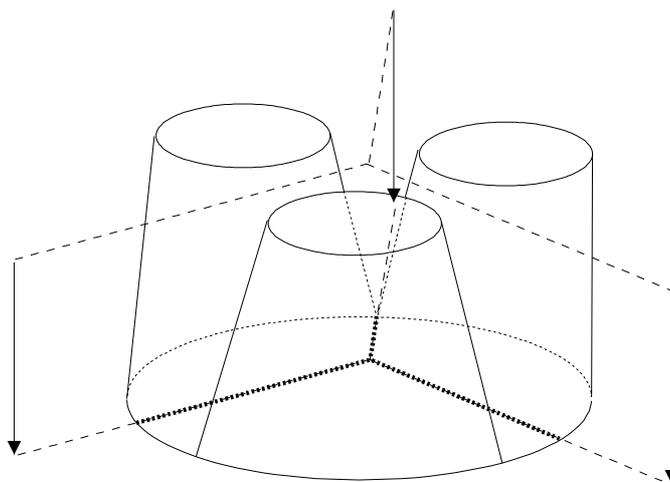


ABBILDUNG 3.4.4. Interpolation nach Boissonat und Geiger: Das zweidimensionale, externe (Punkt-)Skelett (gestrichelte Linie, oben) wird auf die jeweils gegenüberliegende Kontur projiziert (gestrichelte Linie an den Pfeilenden). Punkte, die im Innern der Kontur zu liegen kommen, werden als *innere Punkte* in die Kontur eingefügt (fett gepunktete Linie).

werden die oben angesprochenen Inkonsistenzen behoben (siehe Abbildung (3.4.2)) und man erhält im Regelfall bessere Oberflächenrepräsentationen des Objektes. Diese Einfügung der Punkte kann aber nur über einen — in diesem Falle dreidimensionalen — inkrementellen Ansatz geschehen, und nicht mehr durch die von Boissonat vorgeschlagene Verschneidung der zweidimensionalen Voronoidiagramme der Konturen<sup>8</sup> bestimmt werden.

**3.4.3. Grenzen des Rekonstruktionsansatzes.** Bearbeiten wir mit dem vorgestellten Algorithmus nun mehrere Schichten von Konturdaten, erhalten wir schichtweise eine Delaunaytetraedisierung des Objektes, die durch die Eindeutigkeit der Delaunay-Triangulierung der Deckflächen, welche durch die Konturlinien gebildet werden, zu einem dreidimensionalen Delaunay-ähnlichen Graphen zusammengefügt werden können. Somit sind die Suchalgorithmen, die für diese Graphenstruktur entwickelt wurden, weiterhin anwendbar, was u.a. eine recht effiziente Suche des nächsten Nachbarn erlaubt (siehe 3.2.24). Außerdem ist durch die Tessellierung des Objektes gewährleistet, daß wir eine gültige Oberflächenrepräsentation des Objektes erhalten, d.h. daß sich Dreiecke, die Teil der Oberfläche sind, nicht durchdringen können.

<sup>8</sup>Bis zum Schritt des Einfügens von Zwischenpunkten sollte aufgrund des günstigen Zeitverhaltens die Vortessellierung weiterhin nach dem Boissonat'schen Ansatz erfolgen.

Ob diese Oberflächenrepräsentation jedoch in jedem Falle sinnvoll erscheint, ist dabei eine andere Frage. So sind auch dem hier vorgestellten Algorithmus Grenzen gesetzt, die kurz skizziert werden sollen.

Da das diskrete Skelett des PVD's vor allem bei schlechtem Sampling der Kontur nicht — wie erwartet — innerhalb des Objektes verläuft, sondern starke Sprünge nach Außen machen kann, scheint die Wahl des diskreten Außenskeletts zur Einführung interpolierender Punkte unzureichend zu sein. So wird in [OPC96] vorgeschlagen, die Differenzgebiete, wie sie beim zweidimensionalen Verschneiden der Konturen entstehen, durch das LVD und dessen Skelett anzunähern. Ob es gelingt, diesen Ansatz auf den Boissonat'schen Ansatz zu erweitern, ist noch zu untersuchen.

Ein Nachteil dieser Erweiterung ist allerdings schon vorherzusehen: Die Einschränkung, die nur das Linken von Konturen erlaubt, die eine nichtleere Schnittmenge aufweisen, ist eine — vor allem für elektronenmikroskopische Schnitte — zu starke Einschränkung. Hier kann es bei dünnen Objekten durchaus vorkommen, daß diese keinerlei Überlappung zeigen, obwohl es aus dem Bildmaterial ersichtlich ist, daß diese Strukturen zusammengehören. Dies liegt an den großen z-Auflösungsverhältnissen, wie sie in der EM vorkommen (1:10 bis 1:20), und an den unzureichend korrigierbaren Verformungen der einzelnen Schnitte durch das Ultradünnschneiden.

Der Boissonat'sche Ansatz generiert im allgemeinen gute Rekonstruktionen der Konturdaten. Die Einführung von leichtem Rauschen und kleiner Konkavitäten in die Konturdaten, führt zum einen dazu, daß das Voronoiskelett sehr verrauscht sein kann (siehe [AM96]), wodurch die zusätzlich eingeführten, interpolierenden Punkte zu einer Verschlechterung der Oberflächerepräsentation führen können. Außerdem können durch die dadurch erzeugten, kleinen Randdreiecke Verbindungen entstehen, die als *solid* gewertet werden müssen, und daher nicht durch das im Abschnitt 3.3.6 vorgestellte Kriterium eliminiert werden können (siehe auch Abbildung (3.3.7)).

In der hier vorgestellten Implementation des Boissonat'schen Ansatzes ist daher schon eine Erweiterung des Kriteriums zur Entfernung von nicht-soliden Verbindungen vorgesehen, die das Oberflächenverhältnis der gelinkten Fläche zur Konturfläche berücksichtigt. Der Link wird dann nur überhalb eines bestimmt einstellbaren Schwellwertes aufrechterhalten. Diese Vorgehensweise bietet jedoch den Nachteil, daß die Originalkonturen nicht mehr exakt in der Rekonstruktion repräsentiert sind. Dies ist jedoch auch bei einer Reduktion der Konturdaten der Fall, wie sie im nächsten Abschnitt vorgestellt wird.

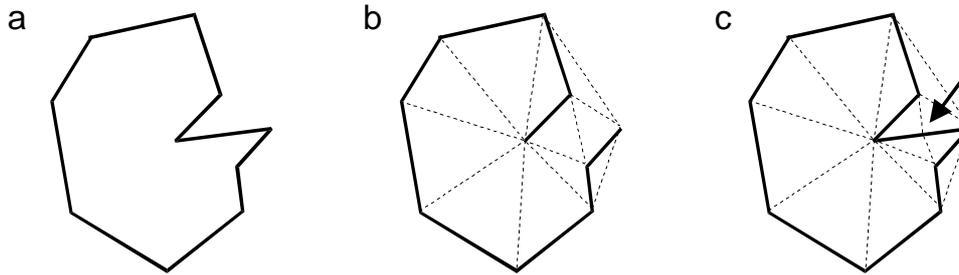


ABBILDUNG 3.5.1. Gezeigt ist ein Fall, bei dem die Konturnachbarschaft nicht im Delaunaygraphen abgebildet ist ((a) + (b)). Rechts sieht man den ersten Schritt eines iterativen Ansatzes, der es erlaubt, daß die Konturnachbarschaft eindeutig in den Delaunaygraphen abgebildet wird: Es werden auf der Konturlinie, die nicht im Delaunaygraphen enthalten ist, zusätzliche Punkte in die geometrische Mitte eingeführt. Im gezeigten Beispiel reicht ein Iterationsschritt aus (siehe Pfeil in (c)).

### 3.5. Vorverarbeitung der Konturdaten

Da im vorgestellten Rekonstruktionsverfahren (siehe Kapitel 2), die Konturdaten interaktiv erfaßt werden, ist bei der Rekonstruktion darauf zu achten, daß die Konturdaten konsistent sind. D.h. im einzelnen, daß der Umlaufsinn der Konturdaten eingerichtet ist, die Konturen sich nicht überschneiden dürfen und das Nachbarschafts-Kriterium (siehe 3.3.1) erfüllen müssen. Wie diese Unwegbarkeiten im einzelnen im Algorithmus Berücksichtigung finden, zeigt der folgende Abschnitt.

Danach soll eine einfache Methode und deren Erweiterung diskutiert werden, die es erlaubt, die Ausgangsdatenmenge an Konturpunkten, welche wesentlich die Laufzeit des Rekonstruktionsalgorithmus bestimmt, reduzieren kann, ohne die Konturen in ihrer Form stark zu verändern.

**3.5.1. Erzwingen der Startbedingungen und Konsistenzprüfung.** Abbildung (3.5.1) zeigt, daß es — je nach Vorgabe der Konturdaten — durchaus vorkommt, daß Konturnachbarn nicht gleichzeitig Nachbarn im Delaunaygraphen sind. Die Abbildung impliziert jedoch auch gleichzeitig, daß zum Erzwingen dieser für den vorgestellten Algorithmus notwendigen Bedingung (vgl. Abschnitt 3.3.1) lediglich zusätzliche Zwischenpunkte in derjenigen Kante der Kontur eingeführt werden müssen, die nicht im Delaunaygraphen repräsentiert ist.

Dies führt uns zu folgendem Algorithmus:

ALGORITHMUS 3.5.1. *Erzwingen der Startbedingungen*

Gegeben sei ein Punkt  $\mathbf{A} \in P$  der geschlossenen Kontur  $C \subset P$ .  $\mathbf{B}$  sei weiterhin ein Konturnachbar von  $\mathbf{A}$ , d.h.  $\mathbf{B} = \mathbf{A}_{-1} \vee \mathbf{B} = \mathbf{A}_{+1}$ . Führe nun die folgenden Schritte aus:

1. Betrachte die Delaunaynachbarn  $N_P(\mathbf{A})$  von  $\mathbf{A}$ . Ist nun:
  - (a)  $\mathbf{B} \in N_P(\mathbf{A})$ , so haben wir eine gültige Nachbarschaftsbeziehung  $\Rightarrow$  **Ende**
  - (b) Betrachte alle Kanten  $\kappa(\mathbf{p}_i, \mathbf{p}_j) \in \mathcal{D}(P)$  mit  $\mathbf{p}_i, \mathbf{p}_j \in N_P(\mathbf{A})$ ,  $i \neq j$ . Existiert nun eine Kante  $\kappa(\mathbf{p}_l, \mathbf{p}_m)$ , für die gilt  $\kappa(\mathbf{p}_l, \mathbf{p}_m) \cap \overline{\mathbf{AB}} \neq \emptyset \Rightarrow$  **Abbruch**
  - (c) **gehe zu 2**
2. Füge einen neuen Konturpunkt  $\tilde{\mathbf{B}} = \frac{\mathbf{A}+\mathbf{B}}{2}$  in  $\mathcal{D}(\tilde{P})$  ein, wobei  $\tilde{P} := P \cup \{\tilde{\mathbf{B}}\} \Rightarrow$  **gehe zu 1**

Mit diesem Algorithmus gelingt es, die Konturnachbarschaft — falls möglich — zu erzwingen. Gleichzeitig gibt er darüber Aufschluß, ob die Konturen inkonsistent sind, d.h. sich durchdringen oder schneiden (siehe Schritt 1b im vorgestellten Algorithmus). In diesem Falle bricht der Rekonstruktionsalgorithmus mit einem Hinweis auf die fehlerhafte Kontur und deren betroffenen Konturpunkte ab. Der Benutzer ist nun gefordert, die Kontur an der angegebenen Stelle zu „bereinigen“. Es wurde hierbei darauf verzichtet, diese Inkonsistenzen durch Heuristiken zu beheben, da diese mit Sicherheit nicht in allen Fällen greifen und das gewünschte Resultat liefern.

**3.5.2. Reduktion der Konturdaten.** Wie schon in den einleitenden Worten zu diesem Abschnitt angedeutet wurde, kann es wichtig sein, die Eingabekonturdaten zu reduzieren, sei es, um — wie bei der *Calyx von Held* — die Anzahl der Oberflächendreiecke derart zu reduzieren, daß ein flüssiges Animieren des Modells möglich wird, sei es, um die Rechenzeit bei der Generierung des 3D Modells zu senken. In der vorliegenden Arbeit wurde dabei ein einfacher Algorithmus implementiert, der durch die Einführung eines sog. *Lost-Area*-Kriteriums sinnvoll erweiterbar ist.

ALGORITHMUS 3.5.2. *Punktreduktion nach dem Winkel-Kriterium*

1. Betrachten wir den Konturpunkt  $\mathbf{A} \in C = \{\mathbf{c}_1, \dots, \mathbf{c}_n\}$  und  $\mathbf{A}_{-1}, \mathbf{A}_{+1} \in C$  die entsprechenden Konturnachbarn. Setze  $\mathbf{A} = \mathbf{c}_1$
2. Definiert  $\alpha := \angle(\mathbf{A}_{-1}, \mathbf{A}, \mathbf{A}_{+1})$  den Winkel des Kontursegmentes. Ist  $\alpha < \varepsilon_\alpha$ , dann unterdrücke den Punkt  $\mathbf{A}$  von der Konturliste  $C$
3. Setze  $\mathbf{A} = \mathbf{A}_{+1}$  und gehe zu Schritt 2, falls  $\mathbf{A}_{+1} \neq \mathbf{c}_1$

Ein Beispiel dieses Algorithmus zeigt Abbildung (3.5.3), was schon auf ein befriedigendes Ergebnis hinweist. Hauptnachteil dieser Methode ist jedoch, daß sie die Delaunaynachbarschaft nicht berücksichtigt<sup>9</sup>, was dazu führen kann, das vorher gültige Konturen zu inkonsistenten Konturen, die sich durchdringen, degenerieren können. Ein weiterer

<sup>9</sup>berücksichtigen kann, da er einen Vorverarbeitungsschritt darstellen soll

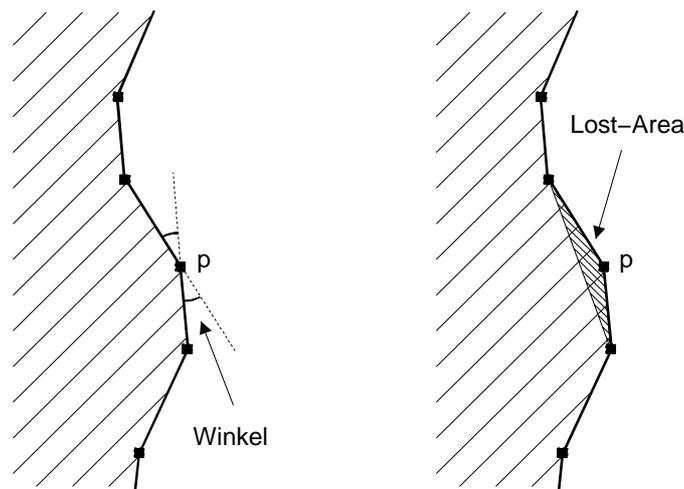


ABBILDUNG 3.5.2. Die Kriterien zur Reduktion von Konturdaten. Betrachtet wird der Punkt  $\mathbf{p}$ . Beim *Winkel*-Kriterium wird der Punkt  $\mathbf{p}$  dann unterdrückt, wenn der Winkel zu den Nachbarn kleiner als ein wählbarer Schwellwert ist (links). Beim *Lost-Area*-Kriterium wird als Bewertungsparameter die Dreiecksfläche berücksichtigt, die bei Unterdrückung des Punktes  $\mathbf{p}$  verlorengehen (oder dazukommen) würde (siehe die schraffierte Fläche, rechts).

Nachteil ist, daß neben großen Abschnitten, in denen die Kontur stark reduziert wird, kleine Abschnitte liegen, in denen die Kontur nicht ausgedünnt werden kann. Dies ist vor allem bei „verrauschten“ Eingabedaten der Fall. Das Sampling der Kontur ist demnach nicht gleichmäßig, was auch zu einer ungleichmäßigen Triangulierung der Objektoberflächen führen wird. Dies ist vor allem im Hinblick auf Simulationsrechnungen, bei denen häufig Kollisions- und Schnittberechnungen mit den Oberflächendreiecken durchgeführt werden müssen, ungünstig<sup>10</sup>.

Damit die reduzierte Kontur ein gleichmäßigeres Sampling aufweist, scheint es sinnvoll zu sein, angelehnt an die Auflösung der Aufnahmen im Präparat oder der  $z$ -Schnittdicke der Ultradünnschnitte einen neuen Parameter zu definieren, der das Winkelkriterium im oben vorgestellten Algorithmus ersetzt. Ein möglicher Parameter wäre hierbei die sog. *Lost-Area*, welche angibt, wie groß die Fläche ist, die durch Unterdrückung eines Punktes von der Originalkontur der Konturfläche verloren geht (siehe Abbildung (3.5.2)).

ALGORITHMUS 3.5.3. *Punktreduktion nach dem Lost-Area-Kriterium*

<sup>10</sup>die Dreiecke enthalten spitze Winkel

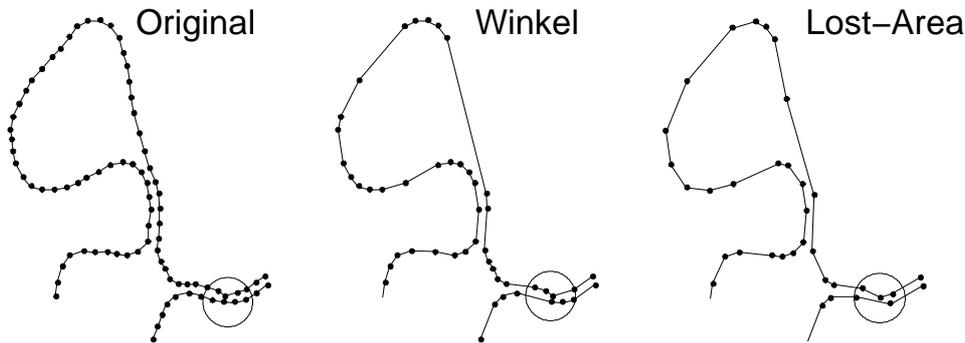


ABBILDUNG 3.5.3. Links ist die Originalkontur gezeigt, wobei die schwarzen Punkte die Eckpunkte des Polygonzuges markieren. In der Mitte ist das Ergebnis der Reduktion nach dem *Winkel*-Kriterium zu sehen; rechts das Ergebnis des *Lost-Area*-Kriteriums. Die mit einem Kreis markierte Stelle kann bei einem noch stärkeren Ausdünnen der Konturen zu einem Berühren oder Schneiden der vorliegenden Konturen führen.

1. Betrachten wir den Konturpunkt  $\mathbf{A} \in C = \{\mathbf{c}_1, \dots, \mathbf{c}_n\}$  und seien  $\mathbf{A}_{-1}, \mathbf{A}_{+1} \in C$  die entsprechenden Konturnachbarn von  $\mathbf{A}$ . Setze  $l_{\text{cum}} = 0, \mathbf{A} = \mathbf{c}_1$
2. Definiert  $l_a := \Delta(\mathbf{A}_{-1}, \mathbf{A}, \mathbf{A}_{+1})$  die Lost-Area.  
Ist  $l_{\text{cum}} + l_a < \varepsilon_{l_a}$ , dann unterdrücke den Punkt  $\mathbf{A}$  von der Konturliste  $C$   
ansonsten setze  $l_{\text{cum}} = 0$
3. Setze  $\mathbf{A} = \mathbf{A}_{+1}$  und gehe zu Schritt 2, falls  $\mathbf{A}_{+1} \neq \mathbf{c}_1$

Das Ergebnis der beiden Algorithmen im Vergleich zeigt Abbildung (3.5.3). Man erkennt, daß die abgewandelte *Lost-Area* Variante eine gleichmäßigere Approximation der Kontur liefert. Schwierigkeit dieser Variante ist es jedoch, den Parameter  $\varepsilon_{l_a}$  sinnvoll zu wählen. Es ist auch denkbar, beide Parameter in den vorgestellten Algorithmen zu kombinieren.

### 3.6. Möglichkeiten der Delaunayrekonstruktion

Durch die so vorgenommene Binarisierung des Tetraederraumes haben wir eindeutig das Objekt beschrieben. Die *inneren* Tetraeder definieren das vom Objekt eingenommene Volumen. Die Oberfläche des Objektes wird durch diejenigen Dreiecksflächen beschrieben, die den Übergang zwischen einem *inneren* zu einem *äußeren* Tetraeder beschreiben. Ebenso ist die Normalenrichtung durch die Richtung der Voronoikante vom *inneren* zum *äußeren* Tetraeder hin eindeutig definiert.

Durch diese Art der Objektrepräsentation oder Tessellierung des  $\mathbb{R}^3$  ergeben sich weitere Möglichkeiten der Gewinnung morphologischer

Daten, die über das Bestimmen von Oberflächen und Volumen, die sich trivialerweise ergeben, weit hinausgehen. Konkrete Beispiele werden in den nächsten Kapiteln anhand der Auswertung der Daten zur *Calyx von Held* vorgestellt.

Rufen wir uns in Erinnerung, daß die vorliegende Zerlegung nicht nur den euklidischen Raum des  $\mathbb{R}^3$  in Tetraeder aufteilt, sondern daß zwischen diesen Grundelementen eineindeutige Nachbarschaftsbeziehungen vorliegen. Hierdurch ist es möglich, zusammenhängende Objekte zu verfolgen und als einzelne Objekte zu identifizieren, diese zu zählen und zu quantifizieren.

Hat man die Nachbarschaft auf Ebene der Tetraeder vorliegen, so liegt uns trivialerweise auch eine Nachbarschaftsbeziehung der Oberflächenelemente der Objekte vor. Dadurch können zusammenhängende, markierte Gebiete auf der Oberfläche des Objektes erkannt, gezählt und ausgewertet werden.

Last but not least bietet die vorliegende Objektrepräsentation die Möglichkeit, ausgezeichnete Punkte des  $\mathbb{R}^3$  innerhalb der Tessellierung schnell zu lokalisieren (siehe Abschnitt 3.2). Dadurch wird eine effiziente Abstandsbestimmung dieser Punkte zur Objekt Oberfläche möglich. So ist eine Basis für *Monte-Carlo-Simulationen* geschaffen, bei denen die diffundierenden Moleküle und ihre Bewegungen zeiteffizient innerhalb des generierten Nachbarschaftsgraphen verfolgt werden können.

## KAPITEL 4

### Anwendung am Beispiel der *Calyx von Held*

In den bisherigen Kapiteln wurde eine Methode vorgestellt, die zur 3D-Rekonstruktion von elektronenmikroskopischen Serienschnitten geeignet ist. In den nun folgenden beiden Kapiteln soll das Vorgehen anhand eines konkreten Beispiels illustriert und durchgeführt werden.

Bei dem untersuchten Datensatz handelt es sich um die sog. *Calyx von Held*. Diese ist eine Riesensynapse, die im medialen Nucleus des Trapezkörpers im Hirnstamm von Säugetieren angesiedelt ist. Sie ist Teil der für das Hören und Orten von Schall zuständigen Neuronen (siehe [HA97]). Bei Betrachtung dieser Synapsen des Rattengehirns unter dem Lichtmikroskop findet man, daß diese Synapse von einer kugelförmigen postsynaptischen Zelle mit einem Durchmesser von 15 – 20  $\mu\text{m}$  und einer schalenförmig sich anschmiegenden präsynaptischen Zelle geformt wird, die einen großen Teil der postsynaptischen Zelle bedeckt (siehe [BHS95] und [HBS97]).

Da die lichtmikroskopischen Schnittbilder der mit Fluoreszenzfarbstoff markierten *Calyx* nicht die kleinen Details wie Membranen, Vesikel oder den synaptischen Spalt auflösen können, wurde die Zelle für die Untersuchung unter dem EM vorbereitet, um im Anschluß nach der in der vorliegenden Arbeit vorgestellten Methode erstmals dreidimensional rekonstruiert zu werden.

#### 4.1. Die Präparation der *Calyx*

Nach folgendem Protokoll wurden die für diese Arbeit verwendeten Dünnschnitte der *Calyx von Held* hergestellt:

- die Ratte wurde 9 Tage nach der Geburt (P9) in Übereinstimmung mit nationalen Bestimmungen enthaupet
- das Gehirn wurde entfernt und in 2.5% Glutaraldehyd über Nacht immersionsfixiert und anschließend in 0.1 M Phosphatpuffer gewaschen
- danach wurden in sagitaler Richtung mit einem Vibratom (Vibratome Series 1000 von Tpi) 100  $\mu\text{m}$  dicke Schnitte angefertigt
- die Vibratomschnitte wurden erneut in 0.1 M Phosphatpuffer gewaschen
- danach wurden die Schnitte in Osmiumtetroxid ( $\text{OsO}_4$ ) 40 Minuten osmiert

- anschließend wurden die Schnitte in einer aufsteigenden Alkoholreihe entwässert
  - zuerst jeweils 15 Minuten in 50 – 60 % Ethanol
  - dann ca. 1 Stunde in 70 % Ethanol, dem noch ca. 1 % Phosphor–Wolfram–Säure und 1 % Uranylacetat zugesetzt wurde.
  - anschließend wurden die Schnitte in 80 %, 90 % und 96 % Ethanol jeweils 15 Minuten gebadet
  - abschließend jeweils 10 Minuten in 100 % Ethanol
- nach der Entwässerung wurde der Alkohol durch zweimaliges, 10–minütiges Baden durch Propylenoxid ersetzt
- das Propylenoxid wurde durch den Kunstharz (**Durcupan**) in einem anschließenden Bad ersetzt
- nach ca. 24 Stunden (zweimaliges Wechseln des Kunstharzes) werden die Schnitte zwischen Objektträger und Deckglas gelegt
- die so flacheingebetteten Vibratomschnitte wurden bei 60° im Trockenraum für ca. 2 Tage ausgehärtet
- auf die ausgehärtete Kunstharzfolie, die das Präparat enthält, wurde ein Kunstharzzapfen aufpolymerisiert
- danach wurde ein pyramidenförmiges Stück des eingebetteten Sagitalschnittes mit einer Rasierklinge unter einem Stereomikroskop herausgetrimmt
- der so präparierte Kunstharzzapfen wurde in ein Ultramikrotom gespannt (**Ultracut S** von Reichert) und mit dessen Diamantschneide ungefähr 60 nm dicke Ultradünnschnitte angefertigt
- die Dünnschnitte wurden anschließend auf handbefilmte Einloch–Kupfergrids mit einem Haarpinsel aufgezogen

#### 4.2. Aufnahme und Digitalisierung des Bildmaterials

Die Aufnahmen erfolgten am EM 109 von Zeiss bei 80 kV Beschleunigungsspannung. Als Filmmaterial wurde ein spezieller Mittelformat–Rollenfilm (60 × 70 mm) von Agfa benutzt (**Aviphot ortho 25 PE1 PL**). Wie wir schon im Abschnitt 1.3 gesehen haben, bestimmt die eingestellte Vergrößerung am Mikroskop die Anzahl der später auszuwertenden Einzelbilder. Vergrößerung und Scandichte müssen derart aufeinander abgestimmt werden, daß ein gescannter Pixel im Präparat ungefähr halb so groß wie die maximal auflösbare Struktur sein sollte (Abtasttheorem der digitalen Bildverarbeitung, siehe [Jäh97]).

Da für die Auswertung an der *Calyx von Held* hauptsächlich die äußere Membran der prä- und postsynaptischen Seite und die Kontakte zwischen beiden Zellen interessieren, gibt es zwei Möglichkeiten, die Aufnahmen zu gestalten:

- mit großer Vergrößerung entlang der äußeren Membran aufnehmen und anschließend in niedriger Auflösung scannen oder

- mit kleiner Vergrößerung und anschließendem, hochauflösendem Scan die komplette Zelle photographieren.

Es wurden beide Verfahren erprobt, wobei letzteres bei der hier vorgestellten Auswertung zum Tragen gekommen ist. Trotzdem sollen die Schritte und entwickelten Techniken, aber auch die Schwierigkeiten bei der ersten Aufnahmemethode vorgestellt werden.

Bevor man jedoch zur Aufnahme schreiten kann, muß die Vergrößerung des Mikroskops mit Hilfe von Eichgrids bestimmt werden.

**4.2.1. Die Eichmessungen.** Am EM 109 von Zeiss wurden im Zeitraum von 1998 bis 1999 zwei Auswertungen zur Bestimmung der tatsächlichen Vergrößerung durchgeführt. Dabei können auch die Variabilitäten in der Vergrößerung erkannt und quantifiziert werden, auch wenn in biologischen Präparaten mit Sicherheit größere Abweichungen zu erwarten sind<sup>1</sup>. Die folgende Tabelle enthält die bei den Eichmessungen ermittelten Vergrößerungswerte (Gitterkonstante des Eichgrids: 463 nm, siehe Abbildung (4.2.1)).

Vergrößerungseinstellung	Messung	%
1100	1035 -	-
3500	3586 ±53	1.5
4500	4482 ±32	0.7
6600	6585 ±36	0.5
11500	11534 ±88	0.8
20400	19317 ±369	1.9
30000	29101 ±139	0.5
50000	48460 -	-

Es ist zu erkennen, daß selbst bei den Eichmessungen Variabilitäten von bis zu 2 % gemessen werden können. Diese sind bei älteren Mikroskopen meist dadurch bedingt, daß die Spannungsversorgungen noch nicht hochstabilisierend waren. Aber auch die Tatsache, daß das Grid nicht exakt an die gleiche Position im Strahlengang eingeschleust wird, kann für Vergrößerungsschwankungen sorgen (siehe [Rei97]).

**4.2.2. Aufnahmen hoher Vergrößerung.** In einem ersten Experiment wurden Aufnahmen in hoher Vergrößerung (20400) gemacht. Bei einer Negativgröße von  $60 \times 70$  mm tastet man im Präparat eine Fläche von in etwa  $2.9 \times 3.4$  µm ab. Um die ganze Zelle zu photographieren wären ca. 100 Aufnahmen pro Schnitt notwendig gewesen. Da das Interesse aber nicht dem Zellinneren, sondern den Zellmembranen gilt (siehe Kapitel 5), konnten Aufnahmen nur entlang den Membranen von post- und präsynaptischer Seite gemacht werden. Mit ausreichender Überlappung photographiert, ergaben sich trotzdem bis zu 40 Aufnahmen pro Schnittebene, die gepatcht werden mußten (siehe

<sup>1</sup>durch thermische Ausdehnung des Präparats

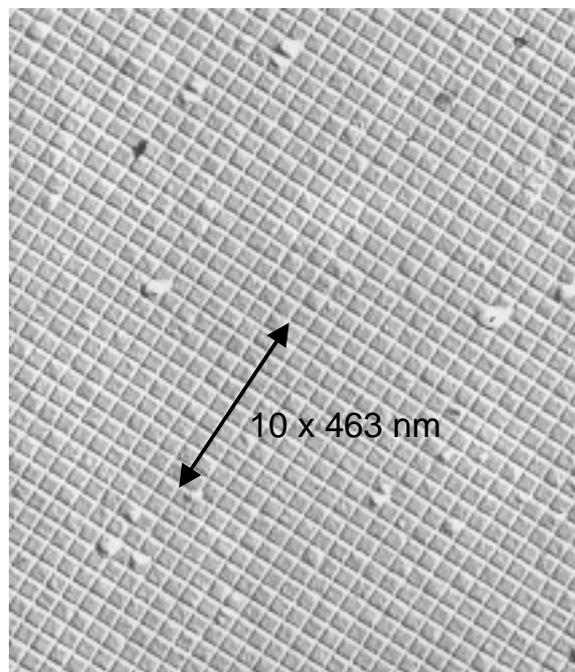


ABBILDUNG 4.2.1. Dargestellt ist das Gitter eines Eichgrids, welches für die Eichmessungen am EM 109 von Zeiss benutzt wurde.

Vergrößerung	20400	4500
Scandichte	600 <i>dpi</i>	1800 <i>dpi</i>
Dünnschnitte gesamt	317	272
Dünnschnitte pro Grid	4 – 6	10 – 16
Anzahl Bilder	6889	683 + 310
Anzahl Bilder pro Schnitt	22 (4 – 41)	4 (1 – 7)
Mittlere Bildgröße	1700 × 1450 $\approx 2.47 \text{ Mb}$	4288 × 5002 $\approx 21.4 \text{ Mb}$
Datenaufkommen	16.0 <i>Gb</i>	20.0 <i>Gb</i>
Komprimierte Daten	1.5 <i>Gb</i> $\approx 10\%$	4.2 <i>Gb</i> $\approx 20\%$

TABELLE 1. Dargestellt sind die Datenblätter der unterschiedlichen Aufnahmeverfahren, die bei der Untersuchung der *Calyx von Held* angewendet worden sind.

Abbildung (4.2.2)), was sich zu einer Gesamtbildmenge von ungefähr 7000 Aufnahmen aufsummiert hat.

Ein Blick auf das Datenblatt dieser Studie in Tabelle (1) zeigt, daß die riesige Anzahl an zu scannenden Bildern ein großer Zeitfaktor

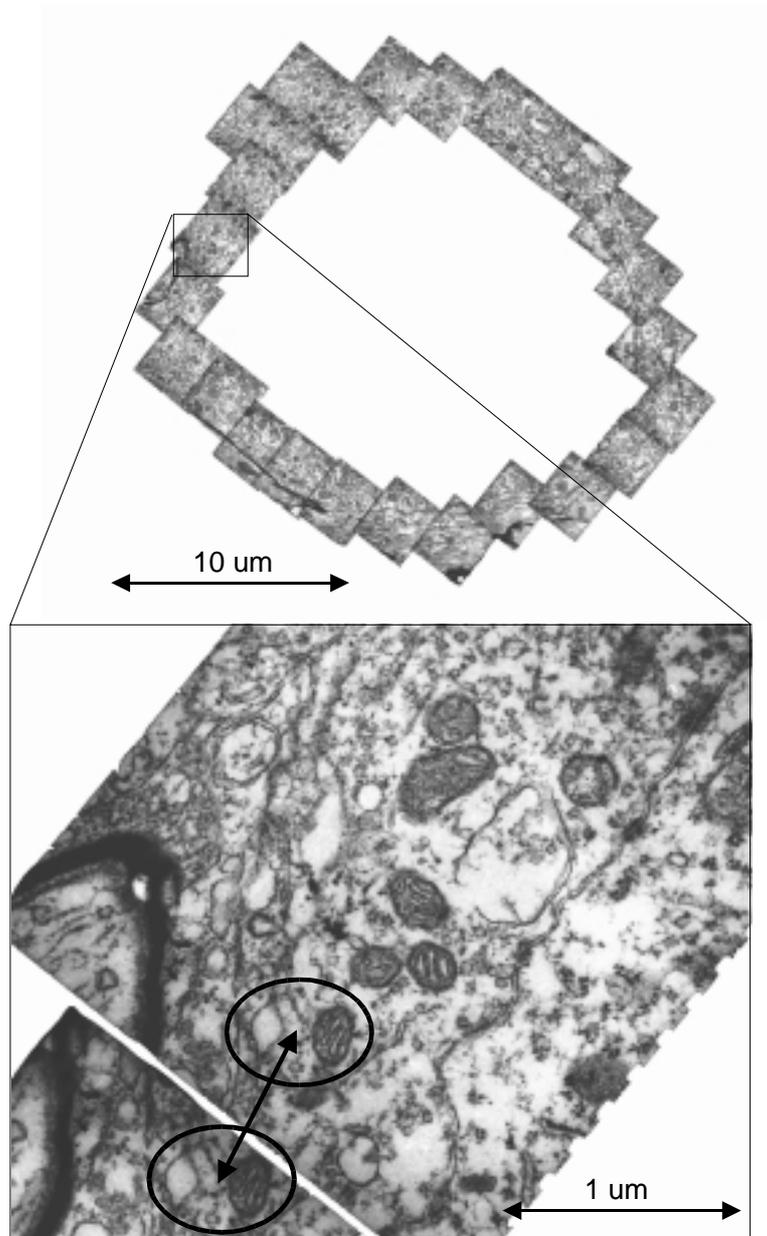


ABBILDUNG 4.2.2. Hier ist einer der Ringe durch die *Calyx von Held* zu sehen (oben), der bei Aufnahmen hoher Vergrößerung entstanden ist. Man erkennt an der Lücke (siehe Ausschnittvergrößerung unten), daß die scheinbare Vergrößerungsänderung durch Aufheizen des Präparats und die sich akkumulierenden Fehler beim Patterning dazu führen, daß sich die Ringe nicht exakt schließen lassen (siehe die als Beispiel markierte Mismatch-Region, unten). Die Aufnahmequalität und der Kontrast sind jedoch sehr gut.

darstellt. Um diesen Schritt zu vereinfachen, wurde ein simpler Algorithmus implementiert, der mehrere, auf der Scanfläche aufgebrachte Negative automatisch segmentieren, ausschneiden und numerieren kann.

Wie wir jedoch aus Abbildung (4.2.2) ersehen können, sind bei dieser Aufnahmeform Schwierigkeiten beim Pattering der Bilder derart aufgetreten, daß die entstehenden, ringförmigen Ausschnitte des Präparats nicht geschlossen werden konnten. Dies lag an zwei Fehlerquellen: Zum einen konnten sich die Fehler, die beim Pattering entstehen, durch die große Anzahl der Einzelaufnahmen und durch deren ringförmige Anordnung akkumulieren. Zum anderen hat sich das Präparat derart bei der Aufnahme erhitzt, daß es sich während des Aufnahmevorganges sichtlich ausgedehnt hat. Dies hat zu einem merkbaren Vergrößerungseffekt geführt, wodurch sich der Ring scheinbar aufweitet.

4.2.2.1. *Automatische Negativsegmentierung.* Bei dieser Aufnahmeserie konnten ungefähr 8 Negative pro Scanvorgang bei 600 dpi verarbeitet werden, was die Anzahl der notwendigen Scanvorgänge auf in etwa 900 reduziert hat. Die implementierte Vorgehensweise läßt sich in die folgenden Schritte aufteilen (siehe auch Abbildung (4.2.3)):

- Es wird zunächst, ausgehend vom Bild des gesamten Scanfeldes, ein sog. Maximumbild errechnet. Hierbei wird eine Maske der ungefähren Größe des Negatives ähnlich eines Faltungsvorganges über den Gesamtscan geschoben und in der Maskenmitte jeweils die Anzahl der Pixel zurückgeschrieben, die über einem frei wählbaren Schwellwert innerhalb des Maskenbereiches liegen.
- Im Maximumbild wird dann das globale Maximum gesucht. In einer Umgebung des gefunden Maximums, die der Negativgröße in etwa entspricht, wird davon ausgegangen, daß kein weiteres Maximum liegen kann. In den verbleibenden Pixelregionen wird diese Suche iterativ solange durchgeführt, bis eine Anzahl an Maxima gefunden wurde, die der Anzahl an gesuchten Negativen entspricht.
- Die gefundenen Maxima werden derart umsortiert, daß sie eine interne Numerierung von z.B. links oben nach rechts unten ergeben (zeilenweise sortiert), was der Anordnung der Negative auf der Scanfläche entsprechen muß.
- Ausgehend von einem rechteckigen Gebiet, was kleiner als die Negativgröße angenommen wird, wird im originalen Übersichtsscan ein Regionenwachstum durchgeführt, daß durch eine Grauwertschwelle und die maximale Negativgröße begrenzt wird.

Das vorgestellte Verfahren machte es möglich, daß nur sehr wenige, vereinzelte Negative manuell nachbearbeitet werden mußten. Da die Scans an einem HP Scanjet 4T einen nicht allzu großen Bildkontrast

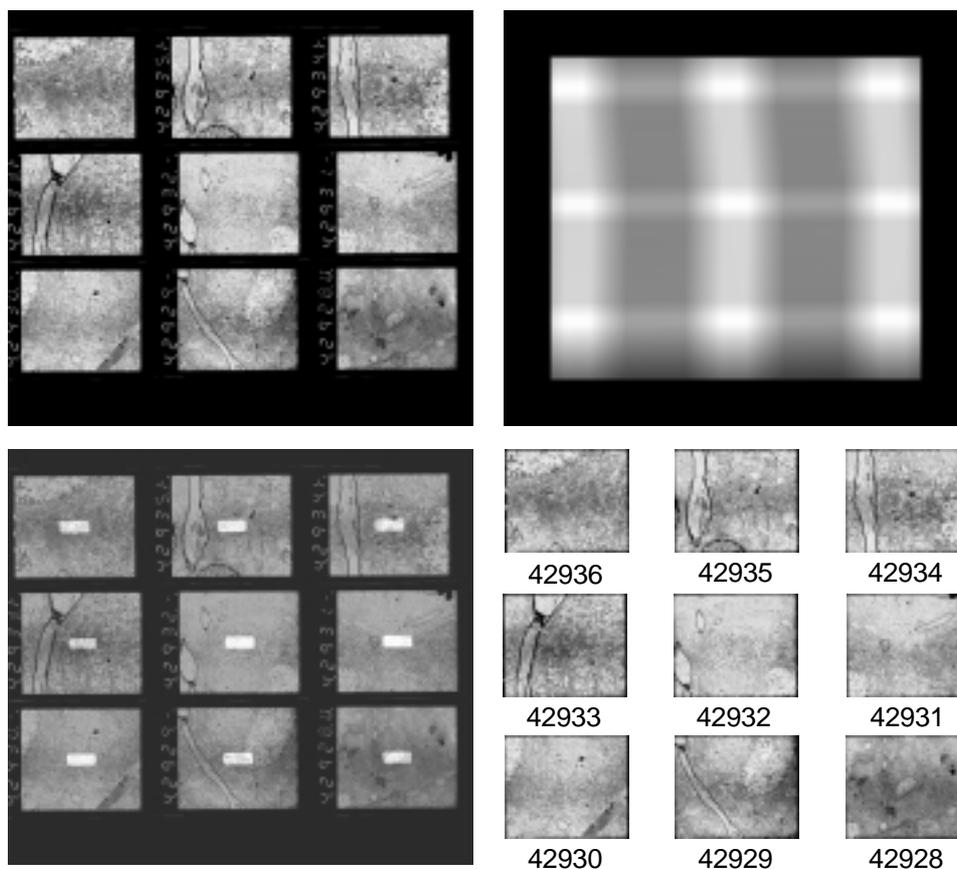


ABBILDUNG 4.2.3. Kurze Illustration zur automatisierten Negativsegmentierung: Links oben ist das Originalbild mit 9 Negativen zu erkennen. Rechts daneben ist das Ergebnis nach der Maximumfilterung zu sehen. Die Peaks im Maximumbild werden mit Hilfe des Wissens der Negativgröße herausgearbeitet und — in diesem Fall — von unten rechts bis oben links durchnummeriert (siehe Überlagerung Maximumbild/Original unten links). Rechts unten sind die erkannten und nummerierten Einzelnegative zu sehen.

lieferten, wurde in einem nächsten Schritt das Histogramm der Bilder automatisch nachbearbeitet.

4.2.2.2. *Histogrammausgleich.* Unter einem Histogrammausgleich (*Histogram Equalize*) versteht man eine pixelorientierte Operation auf Grauwertbildern, die jedem Pixel anhand einer Neuberechneten Nachschlagetabelle einen neuen Grauwert zuordnet. Die neue Nachschlagetabelle (oder auch das Zielhistogramm) wird dabei derart berechnet, daß die Grauwerte gleichmäßig über den Wertebereich des Grauwertbildes verteilt werden. Ein Beispiel zeigt dabei Abbildung (4.2.4). Eine

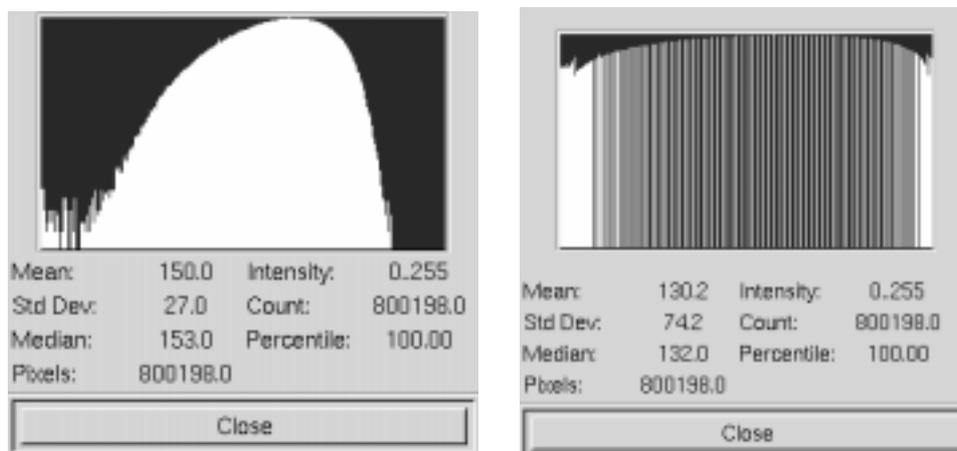


ABBILDUNG 4.2.4. Links ist das Histogramm des Originalbildes, rechts das Histogramm des modifizierten Grauwertbildes dargestellt, bei dem der Histogrammausgleich berechnet wurde. Die Grauwerte werden über den zur Verfügung stehenden Wertebereich möglichst gleichverteilt.

derartige Funktion gehört mittlerweile zum Standardumfang von Bildverarbeitungsprogrammen wie *xv* oder *gimp*. Wie wir später sehen werden, ist der Histogrammausgleich gut geeignet, um Bilder, die ein und dasselbe Objekt zeigen, aber mit verschiedenen Belichtungen aufgenommen wurden, zu normalisieren.

**4.2.3. Aufnahmen kleiner Vergrößerung.** Aufgrund der Schwierigkeiten, die bei Aufnahmen in hoher Vergrößerung des Präparats entstehen, ist es sinnvoller, dieses mit kleinstmöglicher Vergrößerung aufzunehmen. Wie klein die Vergrößerung gewählt werden darf bestimmt hierbei die maximale Scandichte, mit der das Negativ anschließend digitalisiert werden kann. Mit einem hochmodernen Flachbettscanner der High-End Klasse ist es möglich, Scandichten von  $3600 \text{ dpi}$  zu erzielen, was einer gescannten Pixelgröße von  $7 \mu\text{m}$  entspricht. Wie wir im Abschnitt 1.3 gesehen haben, würde somit eine Vergrößerung von  $2000\times$  ausreichen, um die Membranen, den synaptischen Spalt und die Vesikel aufzulösen. Bei dieser Vergrößerung könnte eine Fläche von  $35 \times 30 \mu\text{m}$  im Präparat aufgenommen werden, was bei einer Untersuchung der *Calyx von Held*, die typischerweise einen Durchmesser von  $20 \mu\text{m}$  aufweist, ideal wäre.

Leider verfügt das EM 109 von Zeiss nicht über die optimale Vergrößerung von  $2000\times$ , so daß in der vorliegenden Arbeit mit einer Vergrößerung von  $4500\times$  gearbeitet wurde. Dementsprechend konnte die Scandichte auf  $1800 \text{ dpi}$  reduziert werden, was aber immer noch einen High-End Scanner erfordert. In der vorliegenden Arbeit konnte auf den Scanner SCAI von Zeiss zurückgegriffen werden, der sowohl am EMBL,

als auch seit 1997 am MPI für medizinische Forschung in der Abteilung Biophysik in Heidelberg zu finden ist. Dieser Scanner erfaßt einen Grauwertbereich von  $2^{16}$  Grauwerten, die mittels einer Gammakorrekturkurve auf 256 Grauwerte reduziert werden. Somit ist eine optimale Ausnutzung des Grauwertbereiches von 256 Grauwerten möglich. Typische Belichtungszeiten für die verwendeten Negative lagen bei 3.3 bis 6.6 *ms* pro Pixel. Der Scan eines Negatives dauerte üblicherweise 1.5 Minuten.

Ein Blick auf Tabelle (1) gibt einen Überblick über die Größenordnung der erfaßten Daten. Da die hohe Auflösung der digitalisierten Bilder ausschließlich für die Umfahrungen benötigt wurden, konnte im Pattern-, Squeeze- und Alignment-Modus auf einen weiteren Bilddatensatz zurückgegriffen werden, der auf 1/16 der Originalgröße heruntergerechnet wurde. Diese zweiskalige Bildaufnahme erlaubt es, schnell einen Überblick über die aufgenommenen Strukturen zu bekommen, um dann bei Bedarf in die hohe Auflösungsstufe zu gehen. Dies spart bei Daten dieser Größenordnung viel Lade- bzw. IO-Zeit.

### 4.3. Die Datenerfassung

Im folgenden Abschnitt soll auf die während der Datenerfassung eingeführten Korrekturen, wie die Vergrößerungskorrektur der Einzelbilder im Pattern-Modus und die Skalierungskorrekturen im Squeeze-Modus eingegangen werden. Diese Daten geben einen Überblick über die auftretenden Schwierigkeiten bei der Rekonstruktion von elektronenmikroskopischen Serienschnitten im allgemeinen und sind nicht spezifisch für die hier betrachteten Zellen.

**4.3.1. Die Korrekturen im Pattern-Modus.** Wie wir schon in den vorigen Abschnitten diskutiert haben, treten beim EM besonders bei der Untersuchung von biologischen Präparaten Vergrößerungsfehler auf, die es zu kompensieren gilt. Hierzu werden die Bilder in ihrer Originalgröße, die sich aus der Vergrößerung am Mikroskop und der Scandichte ergibt, in die Ebenen eingebunden. CAR erlaubt nun, daß beim Einfügen eines Bildes die Vergrößerung desselben derart angepaßt werden kann, daß eine sehr gute Übereinstimmung im Überlappungsbereich erzielt wird (siehe Abbildung (4.3.1) und den Korrekturterm 2.3.1 auf Seite 30).

Diese Korrektur wurde im kompletten Datensatz der *Calyx von Held* erlaubt. Abbildung (4.3.2) zeigt einen Überblick über die Korrekturterme, wie sie bei den einzelnen aufgenommenen Bildern eingeführt wurden. Es ergibt sich, daß die Bilder im Mittel eine Schwankung von 1.8 % aufweisen, was im Rahmen der gemessenen Unstabilitäten der Vergrößerung am EM liegt. Die Bildnumerierung erfolgt in der Abbildung in chronologischer Reihenfolge, entspricht also in grober Näherung einer Zeitachse. Der eingezeichnete, schraffierte Bereich in Abbildung (4.3.2)

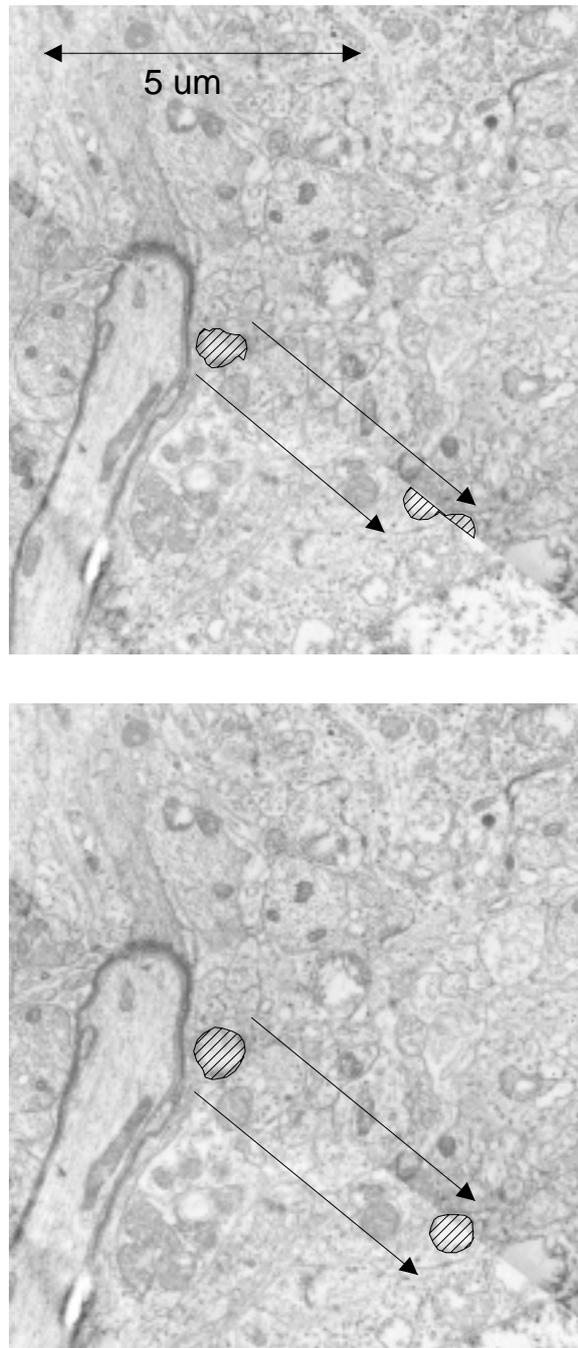


ABBILDUNG 4.3.1. Links sieht man zwei benachbarte Bilder ein und derselben Schnittebene im Präparat. Durch den Vergrößerungseffekt können die Bilder ohne Korrekturen nicht richtig passend aneinander angelegt werden (siehe markierte Strukturen). Erst durch Einführung einer Vergrößerungskorrektur erhalten wir eine gute Übereinstimmung (siehe rechts).

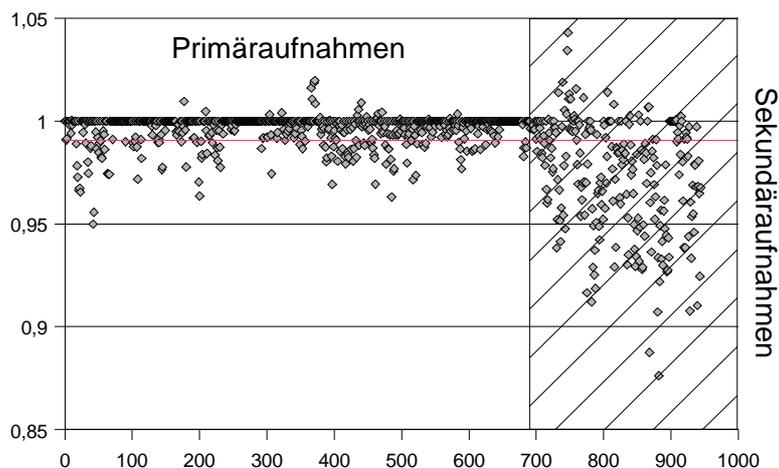


ABBILDUNG 4.3.2. Hier ist ein Plot dargestellt, der die Abweichung der eingefügten Bilder im Pattern-Modus von der Originalgröße (Wert 1) darstellt. Die Bilder weisen im Mittel nach dem Einfügen in die Ebenen eine Vergrößerung von  $99.1 \pm 1.8 \%$  ihrer Originalgröße auf. Ab der Bildnummer 694 (schraffierter Bereich) beginnen die Sekundäraufnahmen, die zu einem späteren Zeitpunkt (5 – 6 Monate nach Präparation) aufgenommen wurden.

markiert dabei den Zeitpunkt, ab dem Sekundäraufnahmen des Präparats angefertigt wurden. Dieser Aufnahmezeitpunkt liegt um 5 – 6 Monate später, als die Erstaufnahmen. Diese Zäsur spiegelt sich auch in den größeren Schwankungen der Bildvergrößerungen und deren Abweichung vom Idealwert 1 wider. Es ist daher anzunehmen, daß die Präparate durch die Lagerung oder den ersten Elektronenbeschuß noch größeren Veränderungen unterliegen.

**4.3.2. Die Korrekturen im Squeeze-Modus.** Der vorliegende Datensatz wurde in einem ersten Durchgang nur im Alignment-Modus bearbeitet, um die dreidimensionale Registrierung der Bilder vorzunehmen. Es wurden somit nur Translations- und Rotationsparameter zwischen den paarweise benachbarten Ebenen korrigiert. Ein Beispiel hierfür ist in Abbildung (4.3.3) zu sehen. Wie dieser Abbildung zu entnehmen ist, sind noch starke Abweichungen zueinander korrespondierender Strukturen zu erkennen.

Diese schlechte Korrespondenz ist Folge von Stauchungen, die im Präparat auftreten. Im Squeeze-Modus werden hier allgemeine, lineare Skalierungen eingeführt, die diese auftretenden Artefakte korrigieren können. In Abbildung (4.3.3) ist deutlich zu erkennen, daß die Korrespondenz nach der Einführung der durch die mit Pfeilen angedeutete Skalierungen deutlich verbessert wurde. Es ist also evident, daß die vorgeschlagenen Korrekturen zu einer Verbesserung des Alignments und somit zu einer Verbesserung der Rekonstruktion führen werden. Aus

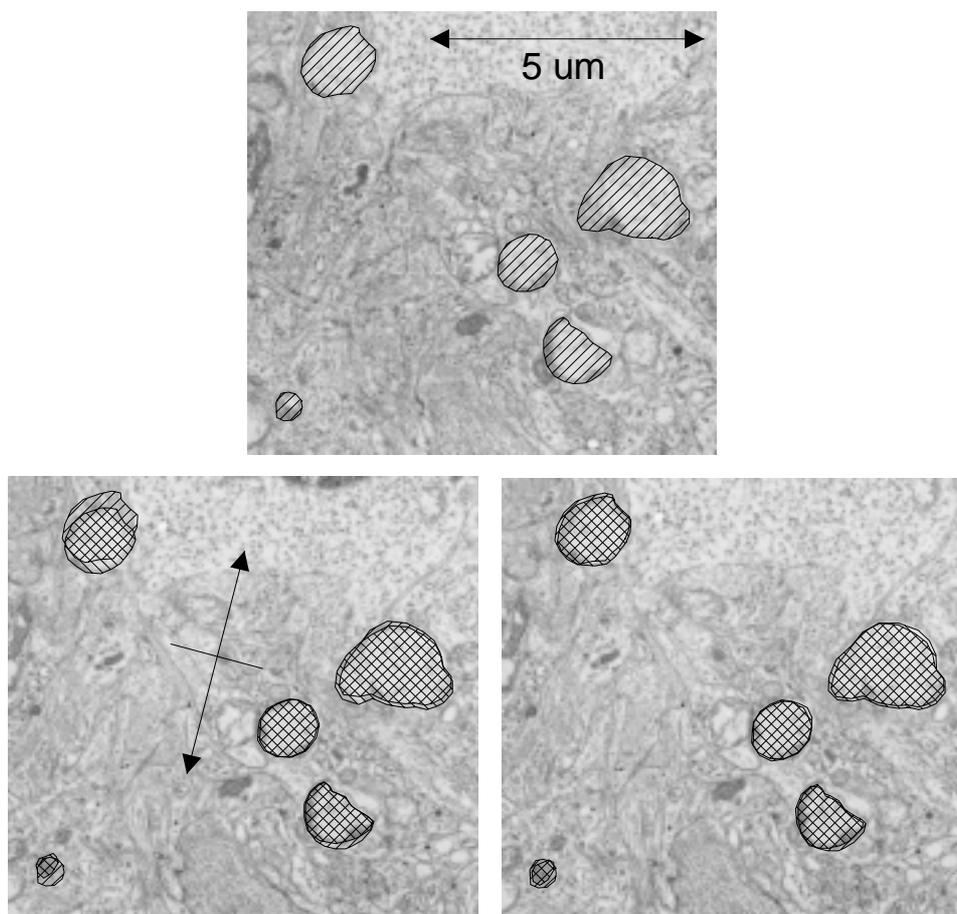


ABBILDUNG 4.3.3. In dieser Abbildung sind oben und unten zwei sukzessive Ebenenbilder dargestellt, wie sie nach Korrektur von Translation und Rotation optimal übereinandergelegt wurden. Zur Illustration des Mismatches wurden markante Strukturen in beiden Bildern markiert. Skaliert man das linke Bild in Pfeilrichtung, erhält man ein korrigiertes Bild (rechts), bei dem die Übereinstimmungen zum Ausgangsbild (oben) deutlich besser sind.

diesem Grund, sind die Skalierungskorrekturen im gesamten Datensatz derart angewandt worden, daß die Ebene  $E_k$  relativ zur Ebene  $E_{k-1}$  angepaßt wurde (siehe Abschnitt 2.4).

Um auch ein objektives Kriterium für die Verbesserung des Alignments einzuführen, greifen wir auf die im Abschnitt 1.3 eingeführte Größe der MDS zurück, welche ein Qualitätsmaß für die Ähnlichkeit von Bildern darstellt. So berechnen wir für die aufeinanderfolgenden Ebenen im Falle des Alignments ohne Korrekturen und im Falle der Einführung von linearen Korrekturen die zugehörigen MDS-Werte.

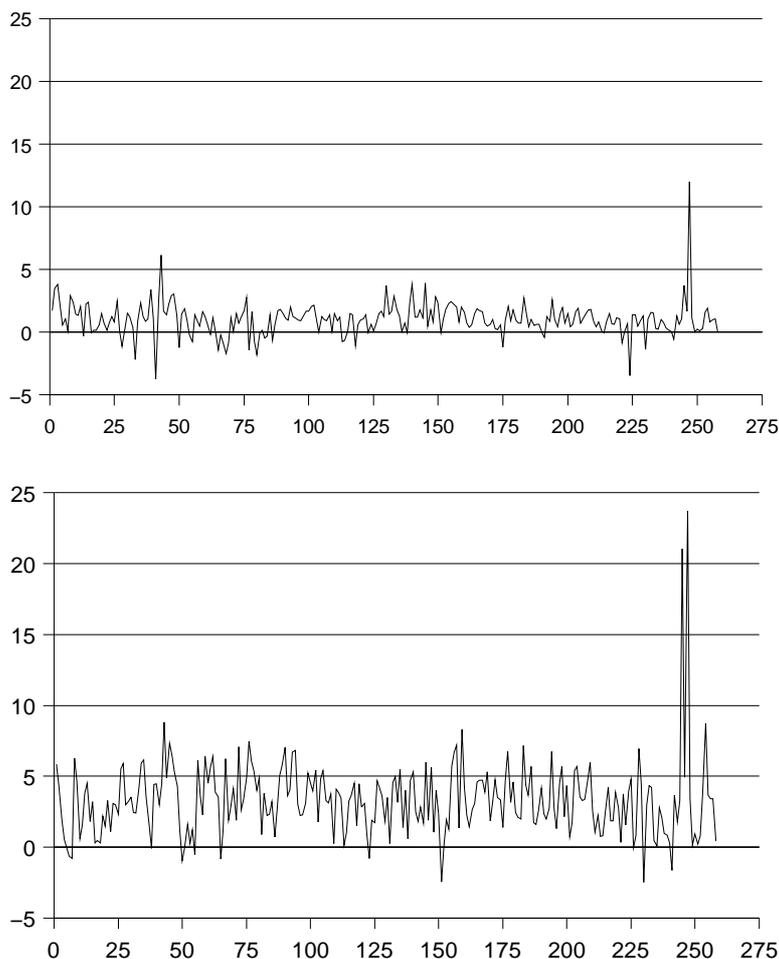


ABBILDUNG 4.3.4. In diesen Diagrammen ist der Differenzwert von  $MDS_{\text{unseq}} - MDS_{\text{seq}}$  zwischen zweier aufeinanderfolgender Bilder aufgetragen, was ein Qualitätsmaß für die Verbesserung des Alignments durch Einführung der allgemeinen, nichtlinearen Transformationen bestätigt (Werte  $> 0$ ). Deutlicher ist die Tendenz (unteres Diagramm), wenn *vor* Bestimmung der MDS-Werte ein Histogrammausgleich der Einzelbilder vorgenommen wird.

Wir erinnern uns daran, daß niedrigere MDS-Werte dabei einer höheren Ähnlichkeit zwischen den Bildern entsprechen. Aus diesem Grund sind in Abbildung (4.3.4) die Differenzwerte der MDS-Werte ohne Korrekturen ( $MDS_{\text{unseq}}$ ) und den Werten mit Korrekturen ( $MDS_{\text{seq}}$ ) betrachtet worden. In der Tendenz sehen wir, daß die MDS-Werte nach der Korrektur niedriger liegen, also der Differenzwert  $MDS_{\text{unseq}} - MDS_{\text{seq}}$  Werte  $> 0$  annimmt, was für eine Verbesserung des Alignments spricht. Noch deutlicher wird der Effekt, wenn wir vor Berechnung des MDS-Wertes einen Histogrammausgleich bei den Einzelbildern vornehmen

(siehe Abbildung (4.3.4) unten). Diese Tendenz läßt sich auch mit den PSNR–Werten bestätigen (hier nicht gezeigt).

Ob der Qualitäts- oder Ähnlichkeitsfaktor MDS (oder PSNR respektive) ein gutes Maß für einen möglichen Alignment- oder Matchingalgorithmus darstellt, bleibt dagegen abzuwarten. Auf jeden Fall sind diese einfachen, pixelorientierten Größen deutlich einfacher zu handhaben, als Korrelationswerte, bei denen erst eine geschickte Renormierung der Werte durchgeführt werden muß, bevor diese verglichen werden können. Dies liegt am unvollständigen und z.T. sehr unterschiedlichen Überlappungsbereich sukzessiver Ebenenbilder.

Es bleibt nun noch zu untersuchen, wie sich die vorgenommenen Verformungen im Datensatz verhalten: Sind diese statistisch gesehen unabhängig oder scheint es eine Vorzugsrichtung für die Deformationen zu geben? Wie wir der Abbildung (4.3.5) eindeutig entnehmen können, ist zweiteres der Fall. Lediglich in den Bereichen, in denen das Verhältnis von maximaler zu minimaler Skalierung ( $s_{\max}/s_{\min}$ ) annähernd 1 ist, gibt es keine eindeutige Tendenz für die Skalierungsrichtung. Außerdem fällt auf, daß sich zwischen dem vorderen und hinteren Bereich, in dem  $s_{\max}/s_{\min}$  deutlich von 1 abweicht, die Richtungen von  $s_{\max}$  zu  $s_{\min}$  vertauschen, also eine Winkelverschiebung von  $90^\circ$  auftritt. Eine mögliche Interpretation dieses Sachverhaltes bedarf jedoch einer weitergehenden systematischen Untersuchung.

In einer abschließenden Untersuchung wurde geprüft, ob evtl. die Verformungsrichtung, wie sie durch den kumulativen Skalierungsplot in Abbildung (4.3.5) angezeigt wird, mit der Schneiderichtung des Ultramikrotoms zusammenhängen könnte. Hierzu haben wir Schnitte betrachtet, bei denen die durch eine Unebenheit im Diamantmesser des Ultramikrotoms verursachte Schnittschärfe zu sehen ist. In Abbildung (4.3.6) ist dargestellt, daß die kumulative Richtung, die durch die Hauptachsen der Verformungsellipse eines Eichkreises gegeben ist (betrachtet wird hier Schnitt Nr. 170), nicht gut mit der Schnittrichtung des Ultramikrotoms übereinstimmt. Auch hier bedarf es einer weiteren Untersuchung, um zu prüfen, wodurch diese Kumulation der Skalierungen verursacht wird. Ein systematischer Fehler bei dieser Untersuchung scheint jedoch aus zweierlei Gründen ausgeschlossen zu sein: Einerseits bestätigen die MDS–Plots in Abbildung (4.3.4) eine Verbesserung des Alignments, andererseits weist die Rekonstruktion in Abbildung (5.2.1) keinerlei Spuren einer Unausgewogenheit der Skalierungswerte auf. Abbildung (4.3.7) zeigt eine Übersichtsaufnahme der Dünnschnitte, in der die Schnittrichtung, als auch die ungefähre Lage der untersuchten *Calyx von Held* eingezeichnet sind.

Allerdings wirft dieser Befund eine zentrale Frage auf: Wie können wir eine Skala im vorliegenden Bildmaterial festlegen? Solange die Ursache für die Verformungen nicht eindeutig geklärt werden kann, ist diese Frage nicht eindeutig zu beantworten. Aus diesem Grund

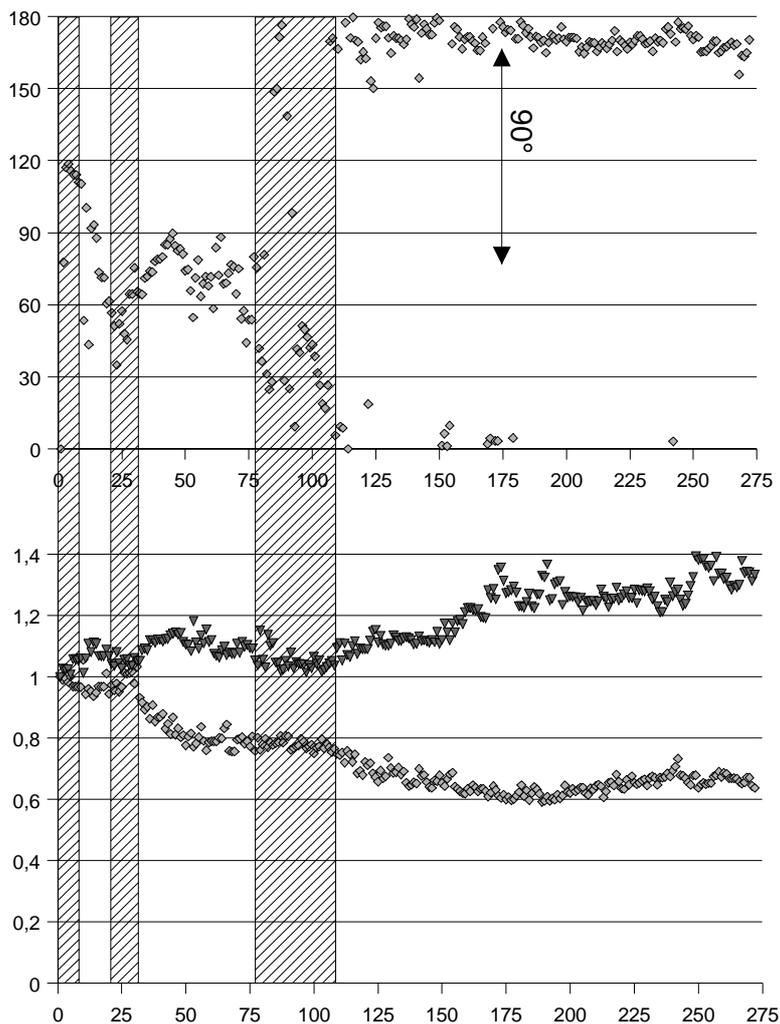


ABBILDUNG 4.3.5. Hier ist aufgetragen, wie sich die allgemeinen Skalierungen von der ersten (unverzerrten) bis zur letzten Ebene akkumulieren. Im oberen Plot ist der Winkel der Richtung aufgetragen, in der minimal skaliert wird. Die maximale Skalierungsrichtung liegt hierzu senkrecht. Der Winkel im oberen Plot hat eine Periode von  $\pi$ . Im unteren Plot sind zum einen der minimale Skalierungswert  $s_{min}$  (Rautensymbole unten), zum anderen das Verhältnis  $s_{max}/s_{min}$  aufgetragen. Man erkennt, daß in den Bereichen, in denen  $s_{max}/s_{min} \approx 1$  ist, die Winkelangabe ungenau wird (schraffierte Bereiche). Im letzten Bereich vertauschen sich die Richtungen von  $s_{min}$  und  $s_{max}$  (Verschiebung um  $90^\circ$ ).

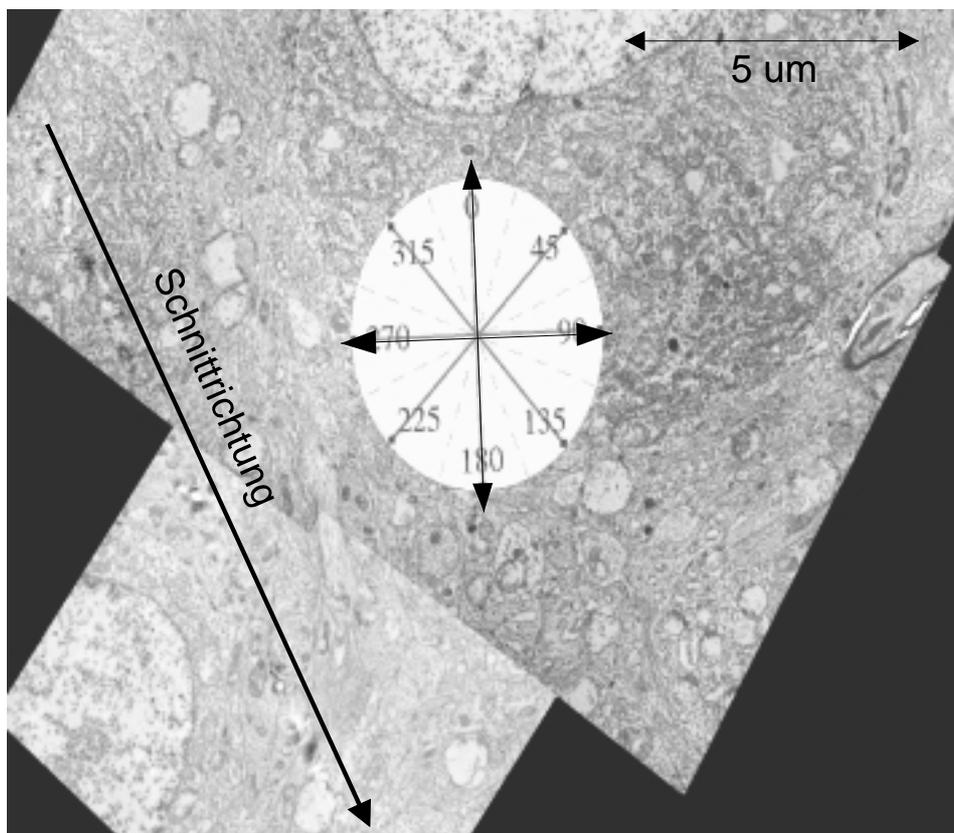


ABBILDUNG 4.3.6. In dieser Abbildung ist die Schneidrichtung des Ultramikrotoms anhand der Schnittscharfe, die durch eine Unebenheit des Diamantmesser im Präparat hinterlassen wurde, eingezeichnet (Pfeil links). Wie wir dem verformten Originalbild eines Kreises, der durch die kumulativen Skalierungen nun eine Ellipse bildet (siehe Mitte), entnehmen können, korrespondiert keine der Hauptachsenrichtungen der Skalierungsellipse hinreichend gut mit der Schnittrichtung (Bild der Schnittnummer 170, siehe auch Abbildung (4.3.5)).

werden im nächsten Kapitel, wenn die quantitativen Daten der *Calyx* vorgestellt werden, Relativwerte oder eine mögliche Abschätzung des Wertebereiches angegeben, in dem sich die morphologischen Daten bewegen können.

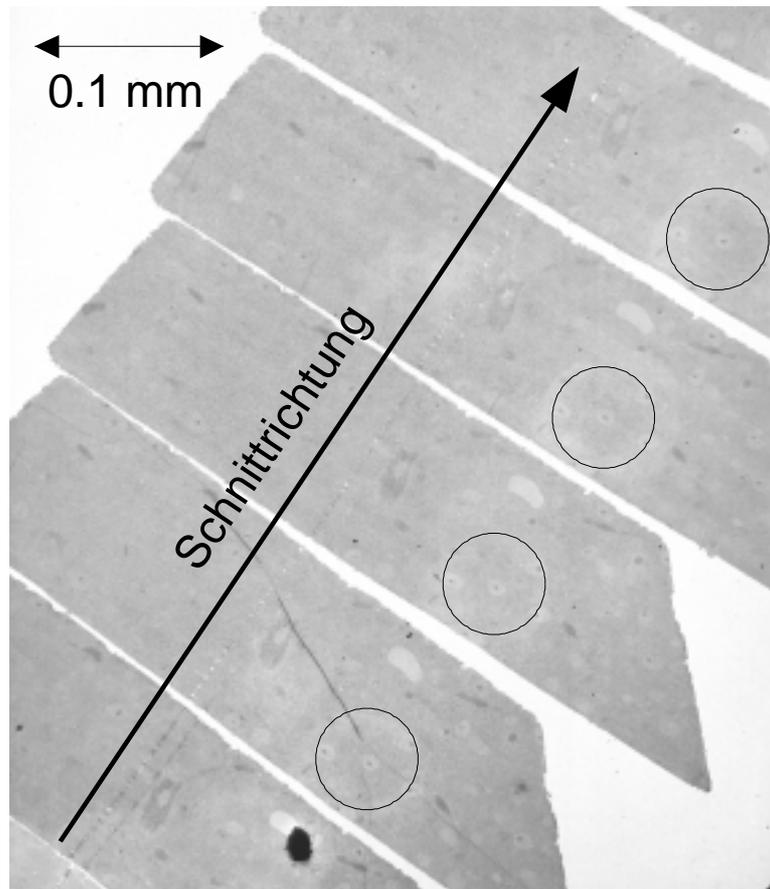


ABBILDUNG 4.3.7. Diese Aufnahme zeigt eine Übersicht über die Form der angefertigten Dünnschnitte. Es sind die Schnitttrichtung und die ungefähre Lokalisation der untersuchten *Calyx von Held* eingetragen.

## Die Rekonstruktion der *Calyx von Held*

Bevor wir nun zur Rekonstruktion und zur Auswertung der dreidimensionalen Gestalt der *Calyx von Held* kommen, die in der vorliegenden Arbeit erstmals auf der Auflösungsstufe des EM durchgeführt wurde, soll in einer kurzen Einführung herausgearbeitet werden, welche Gebiete der synaptischen Kontakte interessant im Hinblick auf Simulationen zur synaptischen Übertragung sind.

### 5.1. Die synaptische Übertragung

Der Vorgang der synaptischen Übertragung ist für die Signal- und Reizübermittlung von Neuronen entscheidend. Werden die Signale in den Neuronen selbst mittels sog. Aktionspotentiale übertragen, können sie an den Berührungs- oder Kontaktstellen der Neuronen, die auch Synapsen genannt werden, auf zwei verschiedenen Wegen weitergeleitet werden.

Zum einen gibt es die sog. elektrischen Synapsen, die es mit Hilfe spezieller Membranproteine, den sog. *Gap Junctions* ermöglichen, daß ankommende, elektrische Signale direkt zu einer Potentialänderung der Partnerzelle am Kontakt führen. Zum anderen gibt es sog. chemische Synapsen. Bei diesen werden durch die ankommenden Aktionspotentiale, die meist mit verstärktem  $\text{Ca}^{2+}$  Einstrom einhergehen, sog. *synaptische Vesikel* in der präsynaptischen Endigung dazu angeregt, an der Zellmembran anzudocken und mit dieser zu verschmelzen. Diesen Prozeß nennt man auch *Exozytose*. Durch das Verschmelzen mit der präsynaptischen Membran wird der Inhalt der Vesikel, der den sog. *Neurotransmitter* enthält in den synaptischen Spalt abgegeben. Der Neurotransmitter diffundiert im synaptischen Spalt und bindet an Rezeptoren auf der postsynaptischen Membran. Durch das Binden des Neurotransmitters werden kurzzeitig Ionen-Kanäle aktiviert, die den Einstrom bestimmter Ionen entlang des Konzentrationsgefälles zwischen extra- und intrazellulärem Raum ermöglichen. Dies führt zu einer Potentialänderung in der postsynaptischen Zelle und löst entweder ein *erregendes* (EPSP) oder ein *inhibitorisches postsynaptisches Potential* (IPSP) aus.

Die *Calyx von Held* gehört zu den chemischen Synapsen. Besonderheit der *Calyx* ist, daß die präsynaptische Zelle ausschließlich diesen einen Kontak zur postsynaptischen Zelle herstellt<sup>1</sup>. Ein Großteil des synaptischen Kontaktes (auch Terminal genannt) sitzt dabei direkt auf dem postsynaptischen Zellkörper auf. Bei Aktivierung der *Calyx* wird sehr zuverlässig ein postsynaptisches Signal induziert. Die Signalübertragung erfolgt sehr schnell und zeitlich präzise. Durch die Größe dieser Synapse von ca. 20  $\mu\text{m}$  ist sowohl die prä- als auch die postsynaptische Seite des Terminals für *Patch*-Elektroden zugänglich. Somit wird das Beobachten der oben beschriebenen Vorgänge der synaptischen Transmission in elektrophysiologischen Experimenten unter kontrollierten Bedingungen möglich (siehe u.a. [HBS97], [BS99] und [WWB<sup>+</sup>99]). Diese Tatsache macht die *Calyx von Held* für Simulationsrechnungen interessant, bei denen das Übertragungsverhalten dieser Synapsen in Modellrechnungen studiert und mit dem Experiment in Übereinstimmung gebracht werden soll.

Bevor dies jedoch gelingen kann, müssen anhand von Rekonstruktionen noch wichtige morphologische Parameter für die Diffusionsrechnung bestimmt werden. Zu diesen Parametern gehören neben Volumen und Oberfläche von prä- und postsynaptischer Seite auch die Größe der Kontaktfläche. Als Kontaktfläche wird hierbei die Membranfläche von prä- oder postsynaptischer Seite verstanden, welche direkt an Membranareale der post- oder präsynaptischen Nachbarzelle angrenzt. Durch zusätzliche Bestimmung des mittleren Abstandes von prä- und postsynaptischer Seite der Kontaktfläche erhalten wir eine Abschätzung für das Volumen des synaptischen Spaltes. Dadurch ist eine Abschätzung der Geometrie und des Volumens gegeben, in welchem die Diffusion der Transmittermoleküle stattfindet.

Die Exozytose der Vesikel und auch das Binden der Transmittermoleküle an bestimmte Rezeptoren findet jedoch nicht überall an der Kontaktfläche zwischen prä- und postsynaptischer Membran statt, sondern beschränkt sich auf sog. *aktive Zonen*. In der elektronenmikroskopischen Aufnahme sind diese Stellen durch eine Verdickung der Membran als dunkle Stellen sichtbar, was auf die veränderte Proteinstruktur der Membran an diesen Stellen zurückzuführen ist. Man nennt sie daher auch *optische Dichten*. Die aktiven Bereiche von prä- und postsynaptischer Membran liegen dabei einander gegenüber. Eine Besonderheit der *Calyx* ist, daß sie im Gegensatz zu anderen kortikalen Synapsen sehr viele dieser aktiven Zonen aufweist.

Neben den aktiven Zonen erscheinen jedoch auch die *Anhaftstellen* oder *Desmosomen* (auch *Puncta Adherentia* genannt) als verdickte, dunkle Membranstrukturen unter dem EM. Da in der vorliegenden Studie keine spezifische Antikörper-Färbung vorgenommen wurde, mit

---

<sup>1</sup>üblicherweise macht eine Zelle mehrere, kleine Kontakte zu verschiedenen anderen Nervenzellen

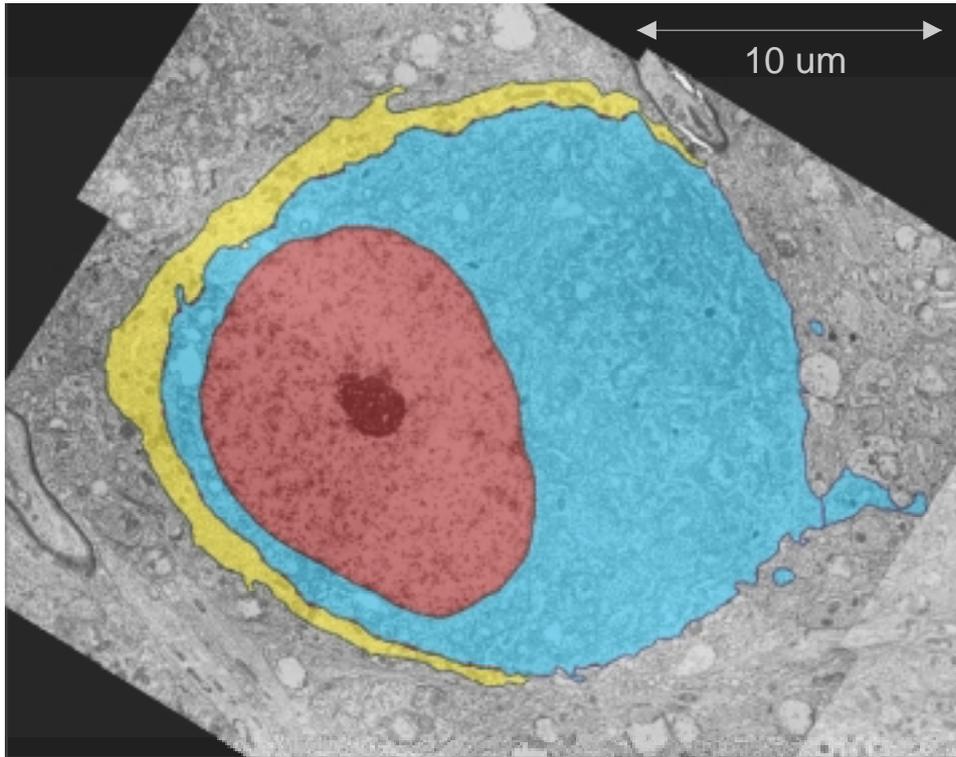


ABBILDUNG 5.1.1. Dargestellt ist eine Aufnahme der *Calyx von Held* unter dem EM, wie sie mit CAR digital zusammengesetzt wurde. Gelb ist die präsynaptische, blau die postsynaptische Zelle dargestellt. Rotbraun ist der Nucleus der postsynaptischen Zelle markiert.

der man zwischen aktiven Zonen und Desmosomen unterscheiden kann, wurde ein einfaches Kriterium zu Rate gezogen, um zwischen beiden Strukturen zu unterscheiden: Waren mehr als 2 – 3 Vesikel in der Nähe der Membranverdickung zu erkennen (in einem Abstand von bis zu 50 nm), wurde diese als aktive Zone markiert (siehe Abbildung (5.1.2)).

Neben diesen, für die Simulationsrechnungen wichtigen Größen wurde auch der Nucleus der Zelle markiert, der in vielen Schnitten, als wichtige Struktur für das Alignment fungieren konnte. So möchte ich mich an dieser Stelle beim Anatomischen Institut in Freiburg von Herrn Prof. Dr. M. Frotscher bedanken, an welchem sowohl die Präparation (Dr. J. Lübke), das Dünnschneiden und die Elektronenmikroskopie (S. Nestel), als auch die Umfahrungen (L. Söhl) gemacht wurden. Für eine Darstellung der markierten Strukturen siehe die Abbildungen (5.1.1) und (5.1.2).

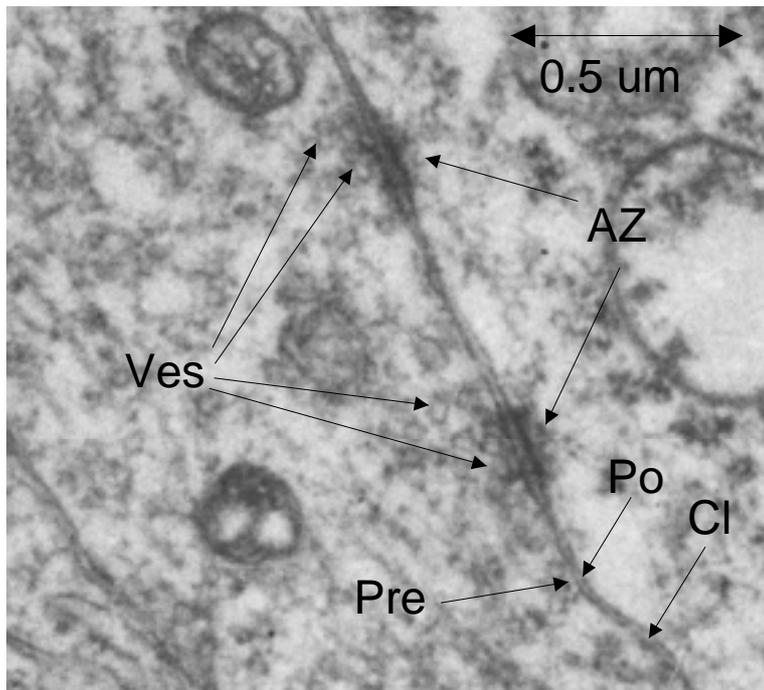
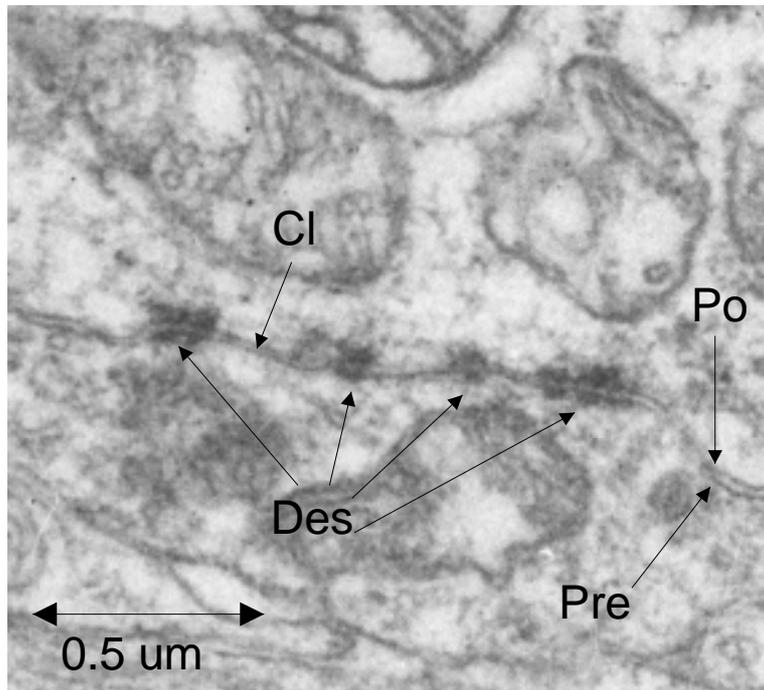


ABBILDUNG 5.1.2. Es sind zwei verschiedene Ausschnitte der in Abbildung (5.1.1) dargestellten *Calyx* gezeigt. Zur Legende: Po bezeichnet die postsynaptische, Pre die präsynaptische Membran, Ves die Vesikel, die in dieser Studie eine aktive Zone definieren, AZ bezeichnet die aktive Zone, Des die Desmosomen, Cl den synaptischen Spalt.

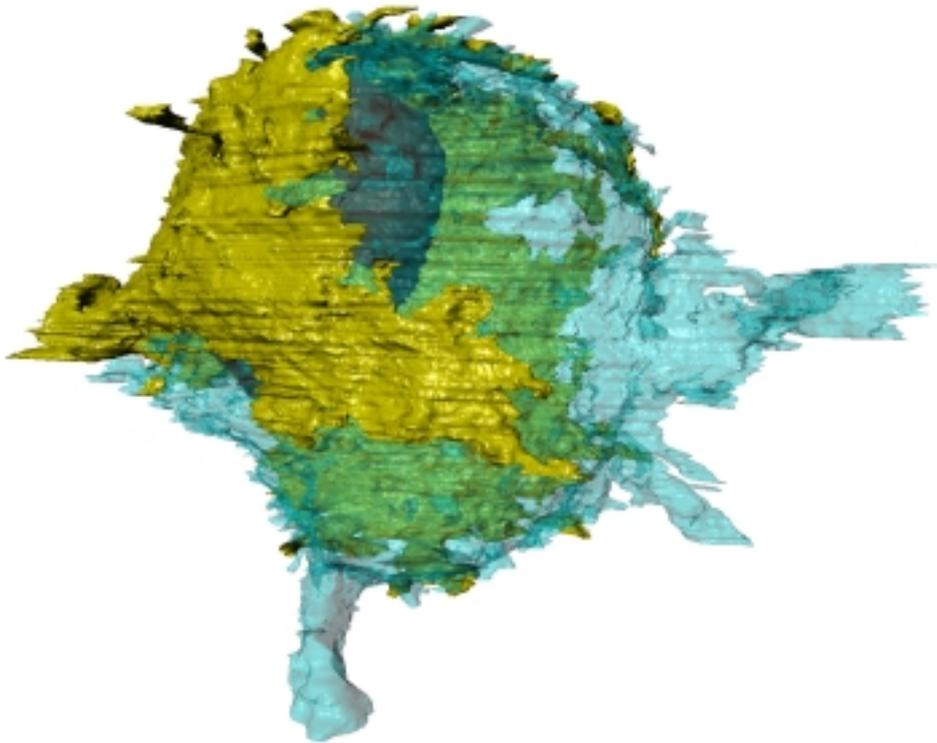


ABBILDUNG 5.2.1. Hier ist die 3D Rekonstruktion der *Calyx von Held* zu sehen. Die gelb markierte präsynaptische Zelle umschließt die blau angefarbte Postsynapse (hier transparent dargestellt). Dargestellt ist dasjenige Modell, bei dem der letzte Schnitt in der Dünnschnittserie unangetastet geblieben ist und welches eine Abschätzung für die minimalen Werte (siehe Text) liefert.

## 5.2. Volumen und Oberfläche

Wie wir bereits im Kapitel 4 gesehen haben, ist eine eindeutige Festlegung eines absoluten Maßstabes durch die auftretenden Verformungen der elektronenmikroskopischen Serienschnitte schwer möglich. Aus diesem Grund geben wir im folgenden stets zwei Werte für die bestimmten Größen an, die eine Abschätzung für den minimalen und maximalen Wert liefern sollen. Berechnet wurden diese Werte, indem wir zum einen den letzten Schnitt (minimaler Wert), zum anderen den ersten Schnitt in der Schnittserie (maximaler Wert) unberührt gelassen haben.<sup>2</sup> Die mittlere Schnittdicke wurde auf  $70\text{ nm}$  festgelegt, da die exakte Schnittdicke der Einzelschnitte nur sehr ungenau durch deren

---

<sup>2</sup>Dieser Schnitt wurde keinen linearen Skalierungen unterworfen.

Interferenzfarben bestimmt werden können (siehe Tabelle 1.2.2 auf Seite 13). Eine genauere Bestimmung der Einzelschnittdicken wurde in dieser Studie nicht durchgeführt.

Wir erhalten nach Berechnung des 3D Modells folgende Werte für Volumen und Oberfläche der *Calyx von Held*<sup>3</sup>:

	Postsynapse		Präsynapse		Zellkern	
Volumen ( $\mu m^3$ )	1860	3420	255	471	353	650
Oberfläche ( $\mu m^2$ )	1570	2520	1480	2360	312	500

Eine Oberflächendarstellung der Rekonstruktion, die eine Abschätzung der minimalen Werte der hier vorgestellten Quantifizierung ergeben hat, sieht man in Abbildung (5.2.1).

### 5.3. Kontaktfläche

Als weiteren wichtigen Parameter hatten wir die Kontaktfläche zwischen prä- und postsynaptischer Zelle zu bestimmen. Durch diese Fläche ist auch gleichzeitig die Fläche des synaptischen Spaltes gegeben. Um aus den markierten Strukturen automatisch diese Fläche herauszuarbeiten, haben wir uns folgendes Algorithmus bedient:

#### ALGORITHMUS 5.3.1. Bestimmung der Kontaktfläche

*Es seien zwei Kontur-Punktemengen  $P$  und  $Q$  und eine Distanz  $d$  gegeben. Sei weiterhin für  $P$  die 3D Rekonstruktion  $\mathcal{D}(P)$  bestimmt. Bestimme nun für jeden Punkt  $\mathbf{q} \in Q$  den nächsten Punkt  $\mathbf{p}_q \in P$  und die zugehörige Kontur  $C_{\mathbf{p}}$  (siehe 3.2.24 auf Seite 55). Markiere iterativ alle Nachbarn von  $\mathbf{p}_q \in C_{\mathbf{p}}$ , für die gilt  $\|\mathbf{n}_{\mathbf{p}_q} - \mathbf{q}\| < d$ . Markiere abschließend alle Dreiecke der Oberfläche von  $\mathcal{D}(P)$ , von denen ein Eckpunkt im vorigen Schritt markiert wurde.*

In der folgenden Tabelle sind die maximalen und minimalen Werte aufgeführt, die sich bei der Berechnung nach dem vorgestellten Algorithmus für  $d = 200 \text{ nm}$  ergeben. Dabei wurde die Fläche zum einen als Teil der präsynaptischen, zum anderen als Teil der postsynaptischen Seite bestimmt. Die Prozentangaben beziehen sich auf die Gesamtfläche der jeweiligen Membran.

	Minimum ( $\mu m^2$ )	%	Maximum ( $\mu m^2$ )	%
präsynaptisch	679	46	1040	44
postsynaptisch	712	45	1090	43

Wir erhalten somit, daß der synaptische Spalt in etwa 45 % der Gesamtfläche von prä- und postsynaptischer Seite bedeckt. Der tatsächliche prozentuale Wert könnte etwas höher liegen, wenn man bedenkt, daß Teile des Axons und der Dendriten mitrekonstruiert wurden.

<sup>3</sup>Eine Kugel des Radius  $10 \mu m$  hat ein Volumen von  $4189 \mu m^3$  und eine Oberfläche von  $1257 \mu m^2$ .

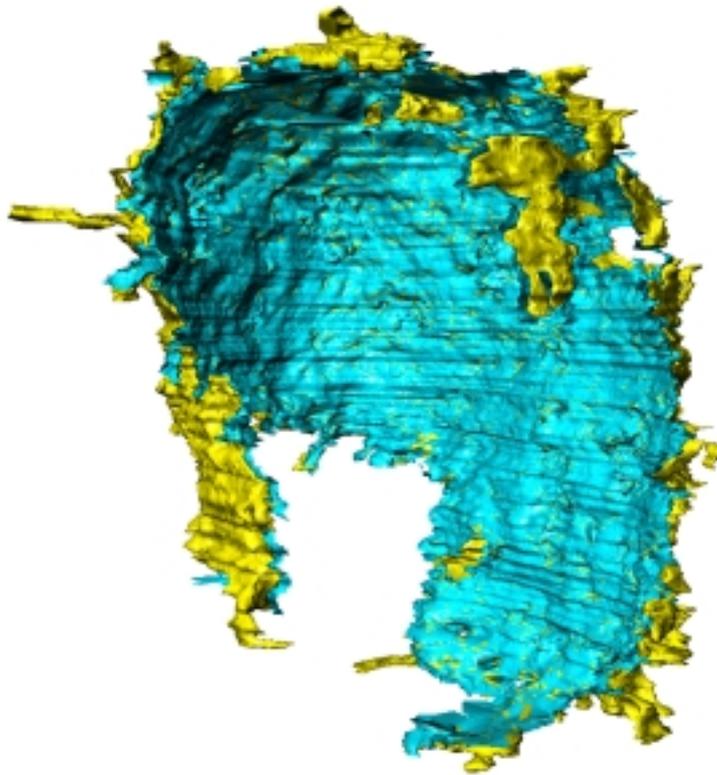


ABBILDUNG 5.3.1. In dieser Abbildung sehen wir auf die Innenseite der präsynaptischen Zelle der *Calyx von Held*. Bläulich dargestellt ist die algorithmisch bestimmte Kontaktfläche zwischen prä- und postsynaptischer Zelle.

Eine Darstellung der extrahierten Dreiecke der postsynaptischen Membran vor dem Hintergrund der präsynaptischen Zelle sehen wir in Abbildung (5.3.1).

#### 5.4. Synaptischer Spalt

Wie wir dem Algorithmus 5.3.1 entnehmen können, wird in einem ersten Schritt, der nächste Punkt  $\mathbf{p}_q \in P$  eines gegebenen Punktes  $\mathbf{q} \in Q$  bestimmt. Dieser Punkt gibt eine Abschätzung für den kleinsten Abstand im synaptischen Spalt, der von den Kontur-Punktemengen  $P$  und  $Q$  eingeschlossen wird. Betrachten wir jeden oben ermittelten Abstandswert als Messung des synaptischen Spaltes und tragen wir die Anzahl der erfolgten Messungen im Intervall  $[0 \text{ nm}, 60 \text{ nm}]$  in  $1 \text{ nm}$ -Schritten auf, so erhalten wir den in Abbildung (5.4.1) gezeigten Verlauf. Unter der Annahme, daß die Messungen annähernd um das

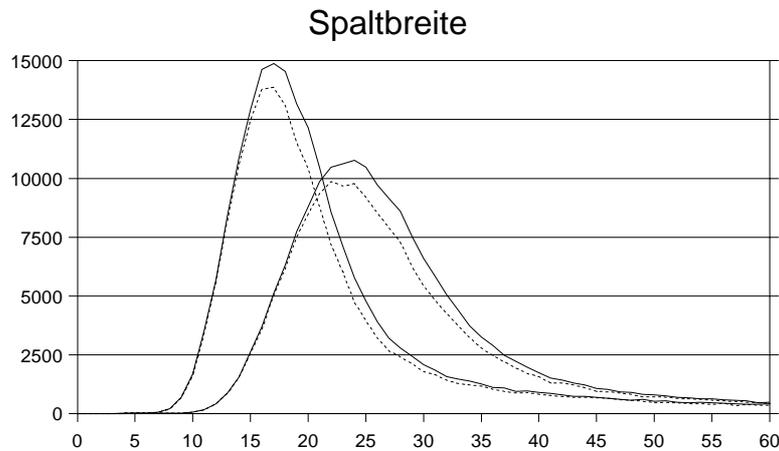


ABBILDUNG 5.4.1. Dargestellt ist die Verteilung der erfolgten Spaltmessungen. Das Maximum gibt eine gute Näherung für die mittlere Spaltbreite an, während die Breite der annähernd normalverteilten Messdaten auf Höhe 0,61 eine Schätzung für  $2\sigma$  gibt. Die prä- (gestrichelte Linie) und postsynaptisch (durchgezogene Linie) bestimmten Werte stimmen gut überein (Werte in  $nm$ ).

Maximum normalverteilt sind, bekommen wir eine Abschätzung für die Breite des synaptischen Spaltes von<sup>4</sup>:

	Minimum ( $nm$ )	Maximum ( $nm$ )
Spaltbreite prä	$17 \pm 4$	$23 \pm 5$
Spaltbreite post	$17 \pm 4$	$24 \pm 6$

### 5.5. Spezialisierte Kontaktfläche

Obwohl die Verformungen der einzelnen Dünnschnitte im Squeeze-Modus korrigiert werden konnten und sich somit das Alignment deutlich verbessert hat, sind die Überlappungen in kleinen Größenskalen wie der Breite des synaptischen Spaltes oder der aktiven Zonen unzureichend bzw. gar nicht vorhanden. Daher ist eine 3D-Rekonstruktion nach einem der hier vorgestellten Verfahren nicht denkbar. Aus diesem Grund muß zu einem anderen Verfahren gegriffen werden, das eine Abschätzung der spezialisierten Kontaktfläche, d.h. der Membranfläche, die von aktiven Zonen bzw. Desmosomen eingenommen wird, erlaubt.

<sup>4</sup>Mittelwert und Standardabweichung ergeben im Bereich 0–200  $nm$  die Werte  $40 \pm 36 nm$ . Für eine exaktere Bestimmung der Spaltbreite müßte eine spezifische Auswahl von Regionen (markierte Abschnitte auf prä- und postsynaptischer Seite) zur Spaltauswertung herangezogen werden.

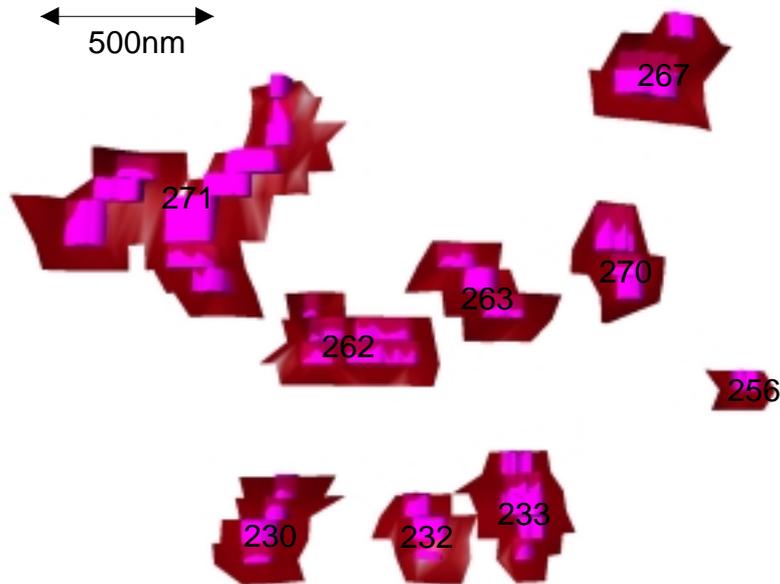


ABBILDUNG 5.5.1. Hier ist dargestellt, wie die Oberflächenregionen nach dem hier vorgestellten Algorithmus automatisch ausgezählt werden. Transparent dargestellt ist die vom Algorithmus extrahierte spezialisierte Kontaktfläche der aktiven Zonen. Die Umfahrungen der aktiven Zonen sind im Klötzchenmodell (*Brick-Modus*) den Flächen überlagert. Man erkennt, daß diese überschätzt werden. Die hier ausgezählten, als disjunkt markierten Oberflächenregionen stimmen allerdings sehr gut mit der visuellen Kontrolle überein.

Für eine Abschätzung des Maximalwertes der spezialisierten Kontaktfläche, bedienen wir uns einer leichten Abwandlung von Algorithmus 5.3.1: Wir extrahieren hier nur die Oberflächendreiecke, bei denen **alle** Eckpunkte in einem Abstand von  $d = 60 \text{ nm}$  zu den Umfahrungen der aktiven Zonen bzw. Desmosomen liegen (siehe Abbildung (5.1.2)). Wie wir sehen werden, führt dies zu einer Überschätzung der Oberflächenwerte, jedoch zu einer guten Übereinstimmung mit den visuell als zusammengehörend erkannten aktiven Zonen (siehe Abbildung (5.5.1)).

Um die erhaltenen Oberflächenregionen automatisch auszählen zu können, müssen wir auf der Menge der Oberflächendreiecke nach Zusammenhangskomponenten suchen. Dies geschieht nach dem folgenden Algorithmus:

ALGORITHMUS 5.5.1. *Bestimmung der Anzahl der Oberflächenregionen*

*Wir nehmen an, daß die 3D-Rekonstruktion  $\mathcal{D}(P)$  der Punktmenge  $P = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$  schon bestimmt sei. Bezeichne weiterhin  $T$  die Menge aller Oberflächendreiecke der Rekonstruktion und sei weiterhin*

eine Untermenge  $T_M \subset T$  gegeben, welche in unserem Falle die spezialisierte Membranoberfläche darstellt. Für ein Element  $t \in T$  bezeichne  $P_t$  die Dreieckseckpunkte, die dieses Dreieck definieren.

1. Starte mit  $i = 1$ , also mit dem Punkt  $\mathbf{p}_1$ . Setze  $n_O = 0$ .
2. Inkrementiere  $i$  solange, bis ein Punkt  $\mathbf{p}_i$  gefunden wurde, für den gilt:  $\mathbf{p}_i \in P_t$  für ein  $t \in T_M$ .  
Ist  $i = n$ , gehe zu Schritt 8.
3. Füge  $\mathbf{p}_i$  in eine Liste  $L_{\text{todo}}$  und  $t$  in eine Liste  $L_T$  ein.
4. Hole das nächste Element  $\mathbf{p}_{\text{todo}}$  aus der ToDo-Liste.
5. Ist  $\mathbf{p}_{\text{todo}}$  markiert, gehe zu 4
6. Ist die Liste  $L_{\text{todo}}$  leer, inkrementiere  $n_O$  um 1. Berechne Oberfläche und Schwerpunkt der Oberfläche, die durch die Dreiecke in der Liste  $L_T$  definiert ist. Leere die Liste  $L_T$ . Gehe zu Schritt 2.
7. Suche alle Nachbarn  $\mathbf{p}_m$  von  $\mathbf{p}_{\text{todo}}$ , für die ein  $t_m \in T_M$  existiert. Füge  $\mathbf{p}_m$  in  $L_{\text{todo}}$  ein. Füge  $t_m$  nur dann in  $L_T$  ein, falls  $t_m \notin L_T$ . Markiere  $\mathbf{p}_{\text{todo}}$ . Gehe zu 4.
8.  $n_O$  gibt nun die Anzahl der Oberflächenregionen an.  $\Rightarrow$  **Ende**

Bestimmen wir die Oberflächen nach diesem Algorithmus im Falle der *Calyx von Held*, erhalten wir folgende Tabelle, wobei auch hier wieder zwischen den Flächen auf prä- und postsynaptischer Seite unterschieden wurde. Wie schon erwähnt, müssen die Oberflächenwerte als maximale Abschätzung gelten, während die Anzahl der einzelnen Flächen eher eine untere Grenze markieren.

	Min. ( $\mu\text{m}^2$ )	#	%	Max. ( $\mu\text{m}^2$ )	#	%
AZ prä.	71.4	427	10.5	101	457	9.7
AZ post.	72.4	417	10.2	103	441	9.4
Des prä.	131	649	19.3	188	761	18.1
Des post.	131	655	18.4	188	769	17.2

Es ergibt sich eine recht gute Übereinstimmung zwischen den Anzahlen, die auf prä- bzw. postsynaptischer Seite bestimmt wurden. Da das absolute Distanzmaß von  $d = 60 \text{ nm}$  für die beiden Datensätze<sup>5</sup>, für die minimale und maximale Werte bestimmt wurden, eine unterschiedliche Bedeutung hat, erklärt sich auch, daß im maximalen Datensatz eine größere Anzahl an Flächenelementen ermittelt wurde: Es wurden weniger Dreiecke auf den Membranoberflächen als spezialisierte Oberflächen markiert, da weniger Punkte das Abstandskriterium erfüllt haben. Trotzdem geben die prozentualen Angaben eine gute Abschätzung dafür, wieviel der Fläche des synaptischen Spaltes von den spezialisierten Membranen eingenommen wird.

<sup>5</sup> $d$  wurde kleiner als die Schnittdicke  $70 \text{ nm}$  gewählt

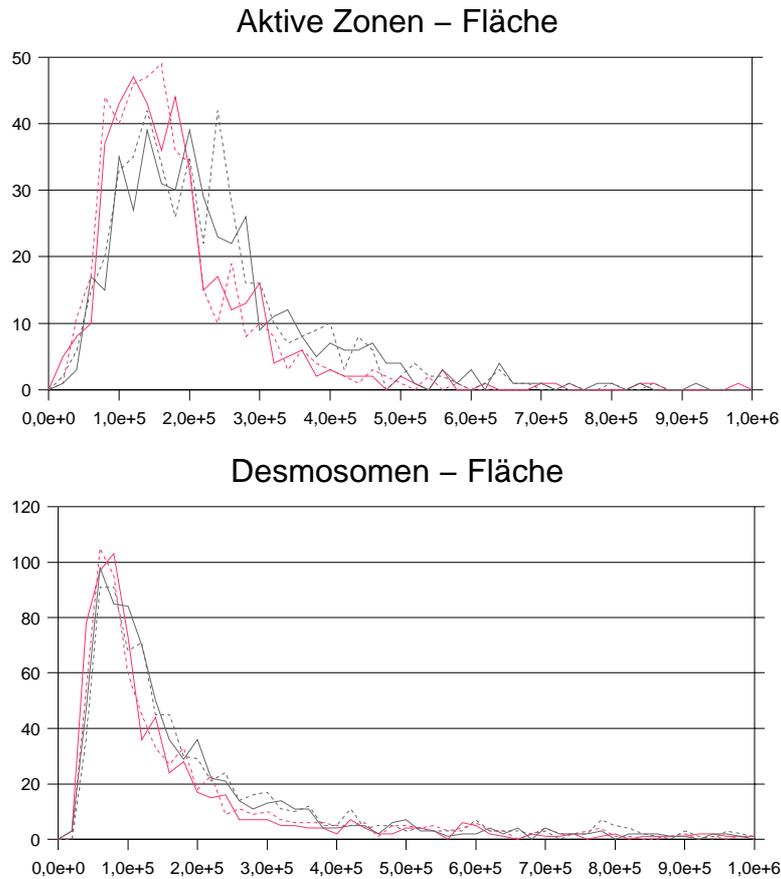


ABBILDUNG 5.5.2. Aufgetragen ist hier die Verteilung der gemessenen Oberflächengrößen von aktiven Zonen (oben) und Desmosomen (unten). Rot markiert sind die minimalen, grau markiert die maximalen Werte. Gestrichelt ist die jeweils auf der präsynaptischen Seite ermittelte Fläche, die gut mit der postsynaptischen übereinstimmt (Flächenangaben in  $nm^2$ ).

Als weitere Größe ergibt sich aus dieser automatisierten Zählung eine Abschätzung für die mittlere Größe einer aktiven Zone und eines Desmosoms. Bestimmt wurden die Werte analog zur Auswertung des synaptischen Spaltes anhand der Häufigkeitsplots der auftretenden Flächengrößen. Die bestimmten Maximalwerte und zugehörigen Varianzen sind in der folgenden Tabelle eingetragen. Es wird hier nicht zwischen den prä- und postsynaptisch bestimmten Werten unterschieden, da diese gut übereinstimmen (siehe Abbildung (5.5.2))<sup>6</sup>.

<sup>6</sup>Mittelwert und Standardabweichung ergeben Werte in den Intervallen  $[17 \pm 12, 23 \pm 18]$  für aktive Zonen und  $[20 \pm 27, 25 \pm 36]$  für Desmosomen (alle Angaben in Einheiten von  $10^{-2} \mu m^2$ ).

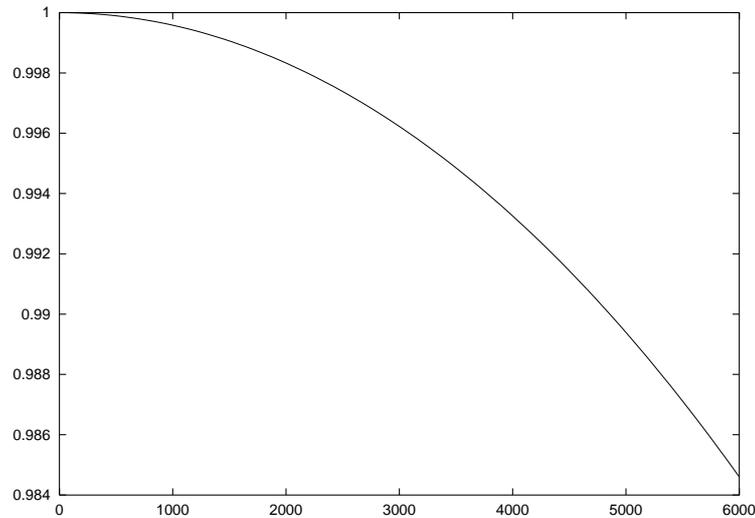


ABBILDUNG 5.5.3. In dieser Abbildung ist dargestellt, wie sich die Bogenlänge  $b = \alpha r$  im Vergleich zur Sehnenlänge  $s = 2 \cdot \sin \frac{\alpha}{2} \cdot r$  verhält. Aufgetragen ist die Funktion  $s / (\arcsin \frac{s}{2 \cdot r} \cdot r)$  mit  $r = 10000 \text{ nm}$ . In dem hier dargestellten Bereich betragen die Abweichung vom Idealwert 1 weniger als 2%.

$(10^{-2} \mu m^2)$	Minimum		Maximum	
aktive Zonen	14	$\pm 7$	17	$\pm 9$
Desmosomen	7	$\pm 4$	7	$\pm 5$

In einer letzten Auswertung können wir anhand der Schwerpunkte der Oberflächenregionen bestimmen, wie groß der minimale Abstand zur nächsten aktiven Zone ist. Erinnern wir uns an die Lösung des Problems zur Auffindung des Punktepaars mit minimalen Abstand auf Seite 55, dann können wir feststellen, daß zur Beantwortung der letzten Fragestellung die Verwendung der Voronoizerlegung der Schwerpunkte der einzelnen spezialisierten Oberflächen geeignet ist. Zwar macht man durch die Bestimmung des euklidischen Abstandes Fehler, die aber in Bereichen von 10 % des Zelldurchmessers vernachlässigbar sind (siehe Abbildung (5.5.3)). Die Auswertungen sind in Abbildung (5.5.4) dargestellt. Da der ermittelte mittlere Abstandswert von  $600 \pm 260$  für aktive Zonen und  $460 \pm 200$  für Desmosomen sehr klein gegenüber dem Zelldurchmesser der runden postsynaptischen Zelle von etwa  $20 \mu m$  ist, zeigt sich, daß der euklidische Abstand sehr wohl eine gute Näherung des realen mittleren Abstandes liefert.

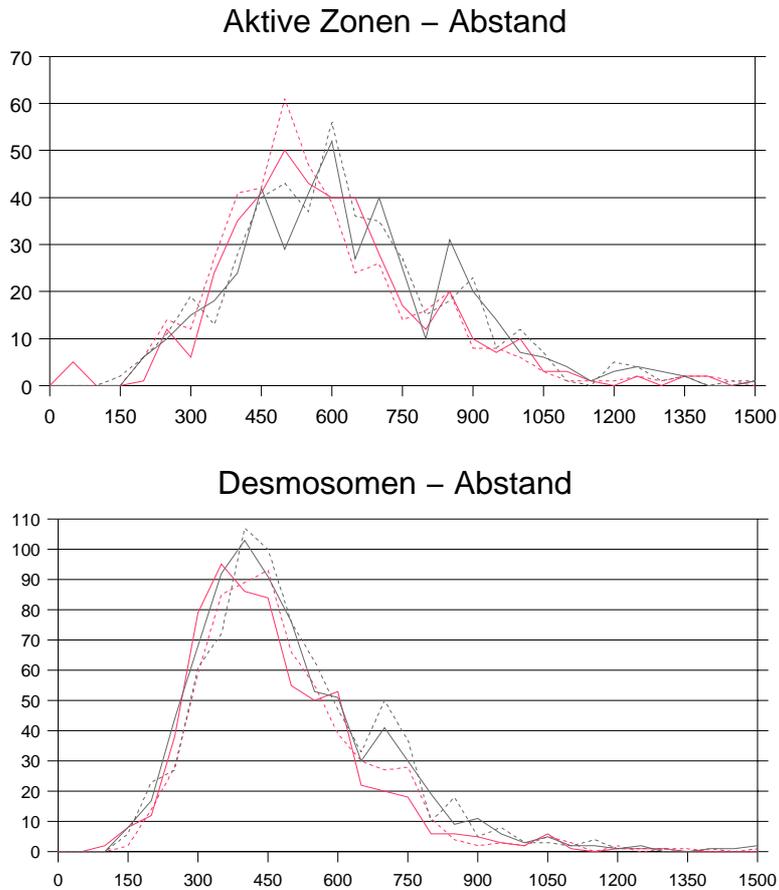


ABBILDUNG 5.5.4. Hier sind Häufigkeitsplots des mittleren Abstandes von einer aktiven Zone zur nächsten aufgetragen. Als Ordinate sind die Anzahl der aktiven Zonen (oben) oder Desmosomen (unten) aufgetragen, die in einem bestimmten Abstandsintervall (siehe Abszisse, Angaben in  $nm$ ) liegen. Die Markierungen sind entsprechend zu Abbildung (5.5.4).

## Zusammenfassung

Die vorliegende Arbeit beschäftigt sich allgemein mit dem Themenkomplex der 3D-Rekonstruktion von elektronenmikroskopischen Serienschnitten von biologischen Objekten. Im speziellen wurden Daten einer Riesensynapse im Hirnstamm einer Ratte erfaßt und ausgewertet. Diese sog. *Calyx von Held* bildet einen wichtigen Bestandteil der für das Hören zuständigen Neuronen, die für die Erfassung und Ortung von Schall verantwortlich sind. Dabei unterteilt sich die Arbeit in die drei wesentlichen Bestandteile:

- die Datenerfassung
- den Rekonstruktionsprozeß
- die Auswertung des 3D Modells

Jeder dieser 3 Bereiche weist für sich gesehen eigene Schwierigkeiten auf, die es im einzelnen zu erkennen und im Modell zu berücksichtigen galt.

### Die Datenerfassung

Bei der Datenerfassung wurden sämtliche Präparations- und Aufnahmeschritte am Anatomischen Institut der Universität Freiburg durchgeführt. Diese Schritte beinhalten die Perfusion des Gehirnmaterials mit Schwermetallen, die sich vor allem an den Zellmembranen anlagern, die Einbettung des Gewebes in Kunstharz und das anschließende Dünnschneiden, damit das präparierte Zellmaterial unter dem Elektronenmikroskop analysiert werden kann. Verwendet wurden hierfür Standardtechniken, die es aber nur dem Geübten und Erfahrenen erlauben, große Schnittserien von Ultradünnschnitten herzustellen. Diese Standardtechniken beinhalten vor allem im Hinblick auf die Rekonstruktion und die Auswertung der morphologischen Daten Nachteile, die in das Rekonstruktionsmodell mit einfließen mußten.

Primärer Nachteil sind die Schrumpfungsartefakte, die bei der Einbettung des Gewebes in Kunstharz und der damit verbundenen Entwässerung entstehen. Diese Vorgänge sind zwar bekannt, jedoch schwer quantifizierbar. Dieser Artefakt kann auch im vorgestellten Modell nicht gemessen und dementsprechend kompensiert werden. Abhilfe scheint hierbei nur die Wahl einer anderen Einbettungs- oder Fixierungsmethode wie die der *Kryofixierung* zu schaffen. Ob mit diesen

schwer handhabbaren Fixierungstechniken große Schnittserien wie im vorgestellten Datensatz möglich sind, bleibt dabei noch zu untersuchen.

Sekundärer Nachteil ist, daß durch das physikalische Durchtrennen des Objektes mechanische Stauchungen des Gewebes auftreten, die eine z.T. sehr starke, ungleichmäßige Verformung der Objekte vor allem bei sehr großen Schnittserien mit sich bringt. Dies wirft sogleich die Frage auf, wie man einen Maßstab im Gewebe finden kann, der einen Rückschluß auf die genauen Abmessungen des untersuchten Objektes erlaubt (siehe [GS86] und [LMM<sup>+</sup>99]).

Ein weiterer Nachteil ist, daß die Schnittdicke und somit die Auflösung des Objektes senkrecht zur Schnittfläche immer noch sehr groß im Vergleich zur erzielbaren Auflösung im Präparat ist. Wir haben es hier mit einem Auflösungsmaßverhältnis von 1 : 10 bis 1 : 20 zu tun. Somit weisen die Objekte teilweise sehr starke Veränderungen von einem Schnitt zum nächsten auf. Außerdem variiert die Schnittdicke mit der Raumtemperatur und der Luftfeuchtigkeit derart, daß die am Ultramikrotom eingestellte Schnittdicke durchaus von der tatsächlichen Schnittdicke abweichen kann.

### Der Rekonstruktionsprozeß

Um möglichst viele dieser Schwierigkeiten quantitativ erfassen und auswerten zu können, wurde in der vorliegenden Arbeit ein Softwarepaket entwickelt, welches die digitalisierten Bilder der elektronenmikroskopischen Aufnahmen zur Grundlage nimmt, um den Rekonstruktionsprozeß mitsamt seinen Korrekturen durchzuführen. Dieses Programm (CAR) bildet das Herzstück der vorliegenden Arbeit und deckt die zentralen Schritte von der Datenerfassung bis zur Auswertung ab. Durch die digitale Datenerfassung mit dieser speziell ausgerichteten Software ist es möglich, den kompletten Rekonstruktionsprozeß quantitativ zu erfassen, was uns zu den Ergebnissen der Auswertungen führt.

So konnte mit Hilfe von CAR festgestellt werden, daß sich die Stauchungen der Schnitte bei einer großen Schnittserie (> 150 Schnitte) signifikant akkumulieren. Es konnte dabei keine gute Übereinstimmung mit der Schnittrichtung des Ultramikrotoms festgestellt werden. Es muß daher in einer weiterführenden Studie genau untersucht werden, welcher Effekt die stetig zunehmende Verzerrung der Schnitte mit steigender Anzahl an Ultradünnschnitten bedingt. Die Skalierungsänderungen über den ganzen Datensatz betrachtet sind auf jeden Fall nicht vernachlässigbar und werfen sogleich die Frage nach einem absoluten Maßstab von elektronenmikroskopischen Aufnahmen auf, der eine Aussage darüber erlaubt, welcher der Ultradünnschnitte dem tatsächlichen Objektschnitt am nächsten kommt. In wie weit weichen die tatsächlich gemessenen Größen von den wirklichen ab? Die Beantwortung dieser Frage ist vor allem im Hinblick auf quantitative Auswertungen wichtig.

In der vorliegenden Untersuchung der *Calyx von Held* werden daher die Variationen in der Größe des Objektes durch Angabe der beiden ermittelten Extremwerte begrenzt und geben somit einen Rahmen an, in dem sich die tatsächlichen morphologischen Werte — bis auf die Schrumpfungartefakte — bewegen könnten. Daß es sich bei der Untersuchung, bei der die angesprochenen Korrekturen manuell und mit visueller Kontrolle durchgeführt wurden, um ein Versuchsfakt handelt, ist aufgrund zweier Tatsachen unwahrscheinlich: Zum einen bestätigen die Funktionsplots des *Mean-Difference-Signals* (MDS) quantitativ die tendenzielle Verbesserung des Alignments nach Einführung der linearen Korrekturen (siehe auch [LMM<sup>+</sup>99]). Zum anderen zeigen die Rekonstruktionen deutlich weniger Störungen der Oberflächenrepräsentation als ohne die Verformungen und geben darüber hinaus keinen Anlaß zur Annahme, daß eine überproportionale Skalierung in eine ausgezeichnete Richtung vorliegt.

Bevor man nun zur Quantifizierung der von einem Spezialisten markierten Zellstrukturen kommen kann, muß im Hinblick auf die auftretenden Verformungen und die zu gewinnenden Parameter ein geeignetes Verfahren gewählt werden, welches sich robust gegenüber den großen topologischen Änderungen der Objekte zeigt und gleichzeitig eine Objektrepräsentation generiert, die eine Vielzahl von Auswertemöglichkeiten bietet.

Die Wahl fiel hierbei auf eine Rekonstruktionsmethode, die die vom Spezialisten identifizierten Objektquerschnitte in eine Tetraedisierung des Raumes integriert. Durch die binäre Zerlegung in Objekt- und Außentetraeder ist eine Quantifizierung von Oberfläche und Volumen elegant und analytisch exakt durchführbar. Die gleichzeitig erzeugte Nachbarschaftsstruktur im Delaunaygraphen erlaubt darüber hinaus die effiziente Berechnung und Gewinnung von problemspezifischen Parametern, wie dies am Beispiel der *Calyx von Held* demonstriert wurde. Es wurde im wesentlichen auf das von Boissonat erstmals 1988 vorgestellte und 1992 durch Boissonat und Geiger verfeinerte Verfahren zur 3D Rekonstruktion der Konturdaten implementiert. Dabei wurde die Methode zur Einführung interpolierender Punkte dahingehend verfeinert, daß die zusätzlich eingefügten Punkte auf eine mediale Ebene angehoben werden. Hierdurch werden topologische Inkonsistenzen, die eine zusammenhängende Kontur nach Einfügen der interpolierenden Punkte in mehrere Teile zerfallen lassen, vermieden. Allerdings treten sowohl beim Boissonat'schen Ansatz, als auch bei der hier implementierten Modifikation dieses Ansatzes Schwierigkeiten auf, wenn die Konturen wenig bzw. keine Überlappung der Konturen vorweisen. Dies ist eine grundlegende Schwierigkeit, die allen 2D basierten Algorithmen zueigen ist. Aus diesem Grund scheint eine weitere Verbesserung der hier in 3D vorgestellten Verfeinerung den richtigen Weg zu weisen.

## Die Auswertung

An der untersuchten *Calyx* wurden nebst Volumina und Oberflächen von post- und präsynaptischer Seite auch deren Kontaktfläche bestimmt. Diese Kontaktfläche wiederum enthält spezialisierte Membranabschnitte, die sog. *Desmosomen* oder *optischen Dichten*, welche zum einen für eine Fixierung von post- zu präsynaptischer Seite sorgen. Zum anderen gibt es unter diesen, im Elektronenmikroskop dunkel gefärbten Membranabschnitten sog. *aktive Zonen*, an welchen nach einem Aktionspotential Neurotransmitter ausgeschüttet wird, der dann in den synaptischen Spalt einströmen kann. Dort diffundieren diese Moleküle in das vom synaptischen Spalt begrenzte Volumen, bis sie auf der postsynaptischen Seite ein weiteres Aktionspotential auslösen.

Genau diese synaptische Kopplung ist Ziel elektrophysiologischer Untersuchungen. Diese Untersuchungen liefern mit Hilfe der *Patch-Clamp*-Technik globale Verhaltensmuster dieser synaptischen Kontakte. Durch Verfeinerung dieser Techniken gelingt es in Einzelfällen, mikroskopische Parameter, wie z.B. die verfügbare Menge an Vesikeln abzuschätzen (*Flash-Photolyse*-Experimente). Eine Kenntnis weiterer mikroskopischer Parameter scheint jedoch wichtig, um das Verhalten von synaptischen Kontakten in Simulationsrechnungen nachbilden zu können. Wichtige Parameter für diese Modellrechnungen sind die Anzahl, mittlere Größe und Verteilung der *aktiven Zonen*, sowie eine Abschätzung zum Volumen des synaptischen Spaltes. All diese Parameter konnten in der vorliegenden Arbeit an der Auswertung eines exemplarischen Kontaktes, der *Calyx von Held*, bestimmt werden.

Aufgrund der noch nicht gelösten Frage nach einem absoluten und exakten Maßstab in den aufgenommenen Schnitten und der Tatsache, daß mit Sicherheit eine Variabilität in der Morphologie der synaptischen Kontakte, auch zu einem ähnlichen Entwicklungsstadium wie der hier untersuchte Kontakt (9 Tage postnatal), zu erwarten ist, sind die hier vorgestellten Werte nur als Anhaltspunkte und Richtlinien zu werten. Sie bieten jedoch die Möglichkeit, diese bisher z.T. unbekannt und nur grob abschätzbaren Werte deutlich präziser zu bestimmen oder erstmals für derartige Untersuchungen zugänglich zu machen.

Es gilt jedoch noch einige Stufen zu erklimmen, bevor die Zusammenführung von morphologischen und elektrophysiologischen Daten in Simulationsrechnungen erfolgreich sein wird. Welche möglichen Verbesserungen und Verfeinerungen zu der hier vorgestellten Methode zur Gewinnung morphologischer Daten aus elektronenmikroskopischen Schnittserien gangbar scheinen, soll im folgenden Ausblick kurz umrissen werden.

## Ausblick

Diese Arbeit hat auf ihrem Weg von der Datenerfassung bis hin zur Datenauswertung von synaptischen Kontakten auf der elektronenmikroskopischen Auflösungsebene nur einen Weg vorzeichnen können, der sich in der dabei entwickelten Software **CAR** manifestiert. Aufgrund der Komplexität der Fragestellung war es allerdings nicht möglich, auftauchende Seitenwege näher und detaillierter zu untersuchen. Hauptaugenmerk für die Bewertung dieser alternativen und weiterführenden Lösungswege muß dabei auf die Verkürzung der sehr zeitraubenden, manuellen Arbeitsschritte gelegt werden. Dabei sollte selbstverständlich die in der vorliegenden Arbeit erzielte Präzision als Maßstab genommen und möglichst übertroffen werden. Ansatzpunkte für eine derartige Weiterentwicklung sind hierbei einige gegeben, die in diesem Ausblick diskutiert und zusammengefaßt werden sollen.

Einer der zeitaufwendigsten Arbeitsschritte stellt hierbei das Erfassen der Objektumrandungen dar. Im Beispiel der *Calyx von Held* war dies ein Zeitaufkommen von ca. einem halben Jahr Vollzeitbeschäftigung für einen Spezialisten. Eine besondere Schwierigkeit im Hinblick auf eine Automatisierung dieses Arbeitsschrittes ist, daß das zu untersuchende Zellmaterial in vielen Fällen nicht spezifisch gefärbt werden kann. Ist eine Färbung der Zelle möglich, ist in der Elektronenmikroskopie meist die Schwierigkeit gegeben, daß diese nicht homogen und gleichmäßig ist. Dieses Problem läßt sich nur mit Hilfe von intelligenten und adaptiven Filtern, wie dem Diffusionsfilter lösen (siehe [KM95] und [Wei98]). Diese Filter fassen die Grauwertverteilung der Bilder als Konzentrationsverteilung mobiler Stoffe auf. Der Diffusionsprozeß bedingt nun, daß sich die Grauwerte durchmischen und einen homogenen Eindruck vermitteln. Der Diffusionsprozeß kann derart gesteuert werden, daß hohe Grauwertgradienten wie eine undurchdringbare Membran für den diffundierenden Stoff wirken. Diese Filtertechniken scheinen für das elektronenmikroskopische Bildmaterial geeignet zu sein. Allerdings bedarf es gezielter Studien zur Parameterfindung der richtigen Diffusionskonstanten. In einem derartig vorverarbeiteten Bildmaterial ist es anschließend einfacher, Objektränder zu extrahieren (siehe [TBM<sup>+</sup>99]). Im Falle von unspezifischen Färbungen müßten kantenorientierte Segmentierungsalgorithmen eingesetzt werden, die lokale Orientierungen, wie die Membranrichtungen, berücksichtigen und gegebenenfalls kurze Unterbrechungen überbrücken (siehe [MLB98]).

Ein ebenfalls sehr zeitraubender und im Hinblick auf die Exaktheit der morphologischen Daten entscheidender Schritt ist derjenige des Alignments. Um einen absoluten Maßstab in den einzelnen Schichten der Ultradünnschnitte finden zu können, muß es gelingen, die realen Abmessungen der Dünnschnitte zu ermitteln. Ein möglicher Weg wäre dabei, den zu schneidenden Block mittels stereoskopischer Methoden vor und während des Schneidevorganges zu vermessen, wobei das Wissen über die vereinfachte Pyramidenform des zugeschnittenen Kunstharzblockes eine entscheidende Vereinfachung darstellen kann. Anschließend müßte die Schnittdicke mittels Interferenzmessungen bestimmt werden (siehe [Cos60] und [ZE60]). Weiterhin müßte es gelingen, die Schnitte in einer Auflösungspyramide aufzunehmen, um die makroskopischen Orientierungen und Verzerrungen der Gewebeschnitte zur Verbesserung des Alignments auf mikroskopischer Ebene verwenden zu können.

Um den Alignmentprozeß in CAR zu vereinfachen, zu beschleunigen und evtl. exakter zu machen, muß als nächster Schritt das Problem gelöst werden, wie man markante Stellen im Zellmaterial, wie z.B. senkrecht zur Schnittebene verlaufende Membranen und Strukturen, die auf mehreren Ebenen wiederzufinden sind, aufeinander anpassen kann. Dabei muß als erster Schritt diejenige lineare Transformation, die Translationen, Rotationen und allgemeine Skalierungen erlaubt, gefunden werden, welche einen Satz von vorgegebenen Punkten ( $n > 3$ ) optimal im Sinne vom mittleren quadratischen Abstand zueinander orientiert. In einer logischen Fortführung dieses Gedankens müßte untersucht werden, ob die Objektumfahrungen in einem iterativen Prozeß für diese Optimierung herangezogen werden könnten. In einem abschließenden Schritt, der die konsequente Weiterführung des punktorientierten Alignments darstellt, sollte noch untersucht werden, ob man einen guten Bewertungsmaßstab finden kann, der die Pixelinformation der Grauwertbilder lokal (vgl. PSNR und MDS) oder global (Korrelationsfunktionen) verwendet und Auskunft über die Qualität des Alignments gibt. Bei diesen Minimierungsversuchen der beschriebenen und noch zu findenden Funktionale ist es mit Sicherheit hilfreich, daß mit Hilfe von CAR ein guter Startwert interaktiv generiert werden kann. Eine globale Lösung des Matchingproblems scheint dabei ein fast unerreichbares Ziel zu sein.

Untersucht man, wie bei der *Calyx von Held*, biologische Objekte größerer Ausdehnung, bei denen sich das Objekt über mehrere Aufnahmen erstreckt, kann eine Erweiterung der oben skizzierten Matching-Algorithmen auf das *Pattern*-Problem angewendet werden. In Kombination mit einem an das Mikroskop gekoppelten, digitalen Bildaufnahmesystem sollte gar ein automatisches Zusammensetzen der einzelnen Detailaufnahmen zum Gesamtbild des Objektquerschnittes möglich werden. Allerdings scheinen auch die Bildverarbeitungssysteme,

wie sie z.B. von der Firma SiS mit den Philips Mikroskopen mitgeliefert werden und ein derartiges Modul anbieten (*MIA - Multiple Image Alignment*), mit dem hier speziell verwendeten Bildmaterial Probleme zu haben. Und dies, obwohl diese Systeme eine recht gute Vorpositionierung der Bilder durch das Mikroskop geliefert bekommen. Auch hier scheint noch Bedarf an weiterführenden Studien zu bestehen.

Im Hinblick auf die anschließenden Simulationsrechnungen der Transmitterfreisetzung im synaptischen Spalt z.B. bei der *Calyx von Held* muß jedoch nicht nur an der Verbesserung des Alignments gearbeitet werden, sondern auch die Rekonstruktionsalgorithmen derart erweitert werden, daß topologische Änderungen nach lokalen Oberflächen- oder Krümmungskriterien besser erfaßt und beschrieben werden können. Der zentrale Bestandteil dieser Simulationsrechnungen in Form des synaptischen Spaltes, der von der an den *Calyx* beteiligten post- und präsynaptischen Zellen geformt wird, zeichnet sich durch seine großen Abmessungen entlang der Membranoberflächen und seinem extrem kleinen Durchmesser senkrecht dazu aus. Dies führt zu einer z.T. unzureichenden Überlappung, wodurch die meisten der hier kurz vorgestellten Rekonstruktionsalgorithmen zu unzureichenden Rekonstruktionsergebnissen führen werden. Als einzig vielversprechender Weg zur Behebung dieser Schwierigkeit scheint hier die Erweiterung des Boissonat'schen Ansatzes und der in dieser Arbeit vorgestellten Verfeinerungen zu sein. Nur dieser Ansatz liefert eine räumliche Zerlegung des Objektes, welche eine schnelle Lokalisation und Verfolgung von Transmittermolekülen in dreidimensionalen Monte-Carlo-Simulationen ermöglicht. Somit kann die gesamte Information, die im Laufe des Rekonstruktionsprozesses gesammelt wurde, in die Simulationsrechnungen mit einfließen.

Zur Erhöhung der Präzision und Exaktheit der bestimmten, morphologischen Daten ist jedoch nicht nur auf der Auswerteseite — wie bisher dargestellt — an Verbesserungen zu denken, sondern muß schon beim Bildaufnahme-prozeß und der Präparation versucht werden, Verfahren zu verwenden, die über die Standardtechniken, die bisher hauptsächlich zur qualitativen Beurteilung des Gewebematerials herangezogen wurden, weit hinausgehen. Eine dabei vielversprechendes Verfahren wurde in [LMM<sup>+</sup>99] vorgestellt: Das Präparat wird mittels *Schockgefrierens* für die Elektronenmikroskopie vorbereitet, was weniger Präparationsartefakte erwarten läßt. Anschließend werden die Schnitte (*Gefrierbrüche*) in einem speziellen Elektronenmikroskop untersucht, in welchem sich das Präparat um einen größeren Winkelbereich drehen läßt ( $\pm 60^\circ$ ). Diese Bildserie wird danach mit tomographischen Rekonstruktionsverfahren (*gefilterte Rückprojektion*) bearbeitet (siehe auch [Fra96]), wobei wir einen dreidimensionalen, registrierten Bilddatensatz erhalten, der annähernd isotrope Auflösung im Präparat bietet (in [LMM<sup>+</sup>99] wurden 7 nm isotrope Auflösung angegeben).

Die Einschränkung, daß die Auflösung in der Elektronenmikroskopie mit der Dicke des Präparates abnimmt, ist natürlich auch bei dieser Aufnahmeform gegeben und kann nur begrenzt durch die Verwendung höher energetischer Elektronen in der HVEM (*High Voltage Electron Microscopy*) kompensiert werden. Dies bedeutet, daß auch bei diesen Verfahren größere Zellen in mehrere, physikalische Schnitte zu unterteilen sind, was die Aktualität der hier vorgestellten Verarbeitungsschritte auch in Zukunft gewährleisten wird. Ob die jüngsten Weiterentwicklungen in der Lichtmikroskopie (z.B. die *Stimulated Emission Detection* (kurz STED) von Hell et al., in Vorbereitung), die Auflösungen von unter 100 nm versprechen, die vorliegenden Probleme in dicken Gewebeschnitten lösen und die Elektronenmikroskopie gar ablösen könnten, ist zu diesem Zeitpunkt noch nicht absehbar und eher fraglich.

Abschließend soll nicht unerwähnt bleiben, daß es sich bei CAR um ein recht umfangreiches Softwarepaket von ca. 35.000 Zeilen Source-Code handelt, in denen die während der Umstrukturierungs- und Entwicklungsphasen entworfenen und wieder verworfenen Programmzeilen keine Berücksichtigung finden. Eine derart komplexe Software muß, damit sie nicht nur weiterbenutzt, sondern auch weiterentwickelt werden kann, nach modernsten Gesichtspunkten des Softwareengineering gestaltet werden, ansonsten ist dieser Software nur eine begrenzte Lebensdauer beschieden. Daher wird die vorliegende Software grundlegend umstrukturiert werden müssen, um den hier vorgestellten möglichen Weiterentwicklungen in CAR einen fruchtbaren Boden zu bereiten. Dies soll in einem nächsten Schritt, evtl. mit Hilfe eines industriellen Lizenznehmers realisiert werden. Damit verbinde ich den Wunsch, daß die Forschungseinrichtungen langsam aber sicher erkennen mögen, daß nicht nur die Softwareentwicklung, sondern auch deren Wartung und Pflege einen wichtigen Bestandteil der wissenschaftlichen Forschung und Arbeit der Zukunft bilden wird.

## Literaturverzeichnis

- [AM96] D. Attali and A. Montanvert. Modeling noise for a better simplification of skeletons. In *Proceedings of the International Conference on Image Processing*, volume III, pages 13–16, 1996.
- [BBJ99] W. Berghorn, T. Boskamp, and K. Jung. Schlanke Bilder. *c't — Magazin für Computertechnik*, 26:184–185, 1999.
- [BBSS99] W. Berghorn, T. Boskamp, S. Schönfeld, and H.G. Stark. Winzig mit Wavelets. *c't — Magazin für Computertechnik*, 26:186–197, 1999.
- [BCL96] C.L. Bajaj, E.J. Coyle, and K. Lin. Arbitrary topology shape reconstruction from planar cross sections. *Graphical models and image processing*, 58(6):524–543, 1996.
- [Ber73] E.R. Berger. Two morphologically different mitochondrial populations in the rat hepatocyte as determined by quantitative three-dimensional electron microscopy. *Journal of Ultrastruct. Research*, 45:303–327, 1973.
- [BG92a] J.D. Boissonnat and B. Geiger. Three-dimensional reconstruction of complex shapes based on the Delaunay triangulation. Technical Report No. 1697, INRIA, May 1992.
- [BG92b] C. Bron and P. Gremillet. 3D reconstruction by image-processing of serial sections in electron microscopy. In A. Kriete, editor, *Visualization in Biomedical Microscopies*, pages 75–105. VCH, Weinheim, New York, Basel, Cambridge, 1992.
- [BHBS98] J.H. Bollmann, F. Helmchen, J.G.G. Borst, and B. Sakmann. Postsynaptic calcium influx mediated by three different pathways during synaptic transmission at a calyx-type synapse. *Journal of Neuroscience*, 18(24):10409–10419, 1998.
- [BHS95] J.G.G. Borst, F. Helmchen, and B. Sakmann. Pre- and postsynaptic whole-cell recordings in the medial nucleus of the trapezoid body of the rat. *Journal of Physiology*, 489(3):825–840, 1995.
- [BHS<sup>+</sup>96] J. Bradl, M. Hausmann, B. Schneider, B. Rinke, and C. Cremer. Resolution effects in practical light microscopy and the application of a  $2\pi$ -tilting device for resolution enhancement. In *3D Imaging sciences in microscopy*. The Royal Microscopical Society, April 1996.
- [BMC92] E. Bertin, R. Marcepoil, and J.-M. Chassery. Morphological algorithms based on voronoi delaunay graphs: microscopic and medical applications. In *Image algebra and morphological image processing III*, pages 356–357, San Diego, Juli 1992. SPIE.
- [Boi88] J.D. Boissonnat. Shape reconstruction form planar cross-sections. *Computer Vision, Graphics, and Image Processing*, 44:1–29, 1988.
- [Bow81] A. Bowyer. Computing dirichlet tessellations. *The Computer Journal*, 24:162–166, 1981.
- [BPC93] E. Bertin, F. Parazza, and J.-M. Chassery. Segmentation and measurement based on 3D voronoi diagram: Application to confocal microscopy. *Comput. Med. Imag.Graph*, 17(3):175–182, 1993.

- [Bra79] G.J. Brakenhoff. Imaging modes of confocal scanning light microscopy. *Journal of Microscopy*, 117:219–232, 1979.
- [Bro79] K.Q. Brown. Voronoi diagrams from convex hulls. *Information Processing Letters*, 9(5):223–228, 1979.
- [BS58] L. Bachmann and P. Sitte. Über Schnittdickenbestimmung nach dem Tolansky–Verfahren. In *IV. Internationaler Kongress der EM*, volume II, page 75, Berlin, 1958.
- [BS96] J.G.G. Borst and B. Sakmann. Calcium influx and transmitter release in a fast CNS synapse. *Nature*, 383:431–434, 1996.
- [BS98] J.G.G. Borst and B. Sakmann. Calcium current during a single action potential in a large presynaptic terminal of the rat brainstem. *Journal of Physiology*, 506(1):143–157, 1998.
- [BS99] J.G.G. Borst and B. Sakmann. Depletion of calcium in the synaptic cleft of a calyx–type synapse in the rat brainstem. *Journal of Physiology*, 521(1):123–133, 1999.
- [Buc88] C.E. Buckley. A divide–and–conquer algorithm for computing 4–dimensional convex hulls. In *Lecture Notes in Computer Science*, volume 333, pages 113–135, Berlin, March 1988. Springer–Verlag.
- [BWY80] J.L. Bentley, B.W. Weide, and A.C. Yao. Optimal expected time algorithms for closest point problems. *ACM Transactions on Mathematical Software*, 6:563–580, 1980.
- [CC78] C. Cremer and T. Cremer. Considerations on a laser–scanning–microscope with high resolution and depth of field. *Microscope Acta (Stuttgart)*, 81:31–44, 1978.
- [CF88] M.A. Casey and M.L. Feldman. Age–related loss of synaptic terminals in the rat medial nucleus of the trapezoid body. *Neuroscience*, 24:189–194, 1988.
- [Cla97] U. Claussen. *Programmieren mit OpenGL*. Springer–Verlag, Berlin, Heidelberg, New York, Barcelona, Budapest, Hongkong, London, Mailand, Paris, Santa Clara, Singapur, Tokio, 1 edition, 1997.
- [Cos60] A. Cosslett. Some application of the ultraviolet and interference microscopy in electron microscopy. *Journal of the Royal Microscopical Society*, 79:263 ff., 1960.
- [CP94] Y.-K. Choi and K.H. Park. A heuristic triangulation algorithm for multiple planar contours using an extended double branching procedure. *The Visual Computer*, 10:372–387, 1994.
- [CS89] K.L. Clarkson and P.W. Shor. Applications of random sampling in computational geometry. *Journal of Discrete and Computational Geometry*, 2:387–421, 1989.
- [DES+98] S. Dietzel, R. Eils, K. Sätzler, H. Bornfeld, A. Jauch, C. Cremer, and T. Cremer. Evidence against a looped structure of the inactive human X–chromosome territory. *Exp. Cell Res*, page 239, 1998.
- [DSW90] W. Denk, J.H. Strickler, and W.W. Webb. Two–photon fluorescence scanning microscopy. *Science*, 248:73–75, 1990.
- [EBS+95] R. Eils, E. Bertin, K. Saracoglu, B. Rinke, E. Schröck, F. Parazza, Y. Usson, M. Robert–Nicoud, E.H.K. Stelzer, J.-M. Chassery, T. Cremer, and C. Cremer. Application of laser confocal microscopy and 3D–voronoi diagrams for volume and surface estimates of interphase chromosomes. *Journal of Microscopy*, 177:150–161, 1995.
- [EDB+94] R. Eils, S. Dietzel, E. Bertin, E. Schröck, Y. Usson, M. Robert–Nicoud, T. Cremer, and C. Cremer. 3D–Voronoidiagramme zur quantitativen Bildanalyse in der Interphase–Cytogenetik. In W. Kropatsch, editor, *Mustererkennung 4*, Heidelberg, 1994. Springer–Verlag.

- [EDB+96] R. Eils, S. Dietzel, E. Bertin, E. Schröck, M.R. Speicher, T. Ried, M. Robert-Nicoud, C. Cremer, and T. Cremer. Three-dimensional reconstruction of painted human interphase chromosomes: active and inactive X-chromosome territories have similar volumes but differ in shape and surface structure. *J. Cell Biol.*, 135(6):1427–1440, 1996.
- [Ede87] H. Edelsbrunner. *Algorithms in Computational Geometry*. Springer-Verlag, Berlin, 1987.
- [EPO91] A.B. Ekoule, F.C. Peyrin, and C.L. Odet. A triangulation algorithm from arbitrary shaped multiple contours. *ACM Transactions on Graphics*, 10(2):182–199, 1991.
- [ES99] R. Eils and K. Sätzler. Shape reconstruction from volumetric data. In B. Jähne, H. Haußecker, and P. Geißler, editors, *Handbook of Computer Vision and Applications*, volume 2, pages 791–815. Academic Press, San Diego, London, Boston, New York, Sydney, Tokyo, Toronto, 1 edition, 1999.
- [ESM+95] R. Eils, K. Saracoglu, C. Münkler, J. Imhoff, K. Sätzler, E. Bertin, S. Dietzel, E. Schröck, T. Ried, T. Cremer, and C. Cremer. Three-dimensional imaging approaches and monte carlo simulations: development of tools to study the morphology and distribution of chromosome territories and subchromosomal targets in human cell nuclei. In P.C. Cheng, editor, *Zoological Studies*, volume 34, pages 7–10, New Jersey, London, Singapore, 1995. World Scientific Publishing Co., Ltd.
- [FKU77] H. Fuchs, Z. Kedem, and S.P. Useton. Optimal surface reconstruction from planar contours. *Commun. ACM*, 20(10):693–702, 1977.
- [Fra96] J. Frank. *Three-Dimensional Electron Microscopy of Macromolecular Assemblies*. Academic Press, San Diego, New York, Boston, London, Sydney, Tokyo, Toronto, 1996.
- [FvDFH96] J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes. *Computer Graphics — Principles and Practice*. Addison-Wesley, Reading (Massachusetts), Menlo Park (California), New York, Don Mills (Ontario), Wokingham (England), Amsterdam, Bonn, Sydney, Singapore, Tokyo, Madrid, San Juan, Paris, Seoul, Milan, Mexico City, Taipei, 2 edition, 1996.
- [GD82] S. Ganapathy and T.G. Dennehy. A new general triangulation algorithm for planar contours. *ACM Transactions in Computer Graphics*, 16(3):69–75, 1982.
- [GG78] P.A. Gaunt and P.N. Gaunt. *Three dimensional reconstruction in biology*. Pitman Medical, London, 1978.
- [GS78] P.J. Green and R. Sibson. Computing dirichlet tessellations in the plane. *The Computer Journal*, 21:168–173, 1978.
- [GS86] J.C. Gilkey and L.A. Staehlin. Advances in ultrarapid freezing for the preservation of cellular ultrastructure. *Journal of Electron Microscopy*, 3:177–210, 1986.
- [HA97] R.H. Helfert and A. Aschoff. Superior olivary complex and nuclei of the lateral lemniscus. In G. Ehret and R. Romand, editors, *The Central Auditory System*, pages 193–258. Oxford University Press, New York, 1997.
- [HBS97] F. Helmchen, J.G.G. Borst, and B. Sakmann. Calcium dynamics associated with a single action potential in a CNS presynaptic terminal. *Biophysics Journal*, 72:1458–1471, 1997.
- [HL92] J. Hoschek and D. Lasser. *Grundlagen der geometrischen Datenverarbeitung*. Teubner, Stuttgart, 2 edition, 1992.

- [HS95] K.M. Harris and P. Sultan. Variation in the number, location and size of synaptic vesicles provides an anatomical basis for the nonuniform probability of release at hippocampal CA1 synapses. *Neuropharmacology*, 34(11):1387–1395, 1995.
- [HSH95] S.W. Hell, J. Soukka, and P.E. Hänninen. Two- and multiphoton detection as an imaging mode and means of increasing the resolution in far-field light microscopy: a study based on photon-optics. *Bioimaging*, 3:64–69, 1995.
- [HSLC94] S.W. Hell, E.H.K. Stelzer, S. Lindek, and C. Cremer. Confocal microscopy with an increased detection aperture: type-b 4pi confocal microscopy. *Optics Letters*, 19:222–224, 1994.
- [Jäh97] B. Jähne. *Digital Image Processing*. Springer-Verlag, 4 edition, 1997.
- [Kep75] E. Keppel. Approximating complex surfaces by triangulation of contour lines. *IBM J. Res. Develop.*, 19:2–11, 1975.
- [KM95] J. Kacur and K. Mikula. Solution of nonlinear diffusion appearing in image smoothing and edge detection. *Applied Numerical Mathematics*, 17:47–59, 1995.
- [LCS96] S. Lindek, C. Cremer, and E.H.K. Stelzer. Confocal theta fluorescence microscopy with annular apertures. *Applied Optics*, 35:126–130, 1996.
- [Lic79] K.G. Lickfeld. *Elektronenmikroskopie*. Verlag Eugen Ulmer, Stuttgart, 1979.
- [LMM<sup>+</sup>99] M.S. Ladinsky, D.N. Mastronarde, J.R. McIntosh, K.E. Howell, and L.A. Staehelin. Golgi structure in three dimensions: Functional insights from the normal rat kidney cell. *Journal of Cell Biology*, 144(6):1135–1149, 1999.
- [LR66] N.J. Lenn and T.S. Reese. The fine structure of nerve endings in the nucleus of the trapezoid body and the ventral cochlear nucleus. *American Journal of Anatomy*, 118:375–390, 1966.
- [LSH95] S. Lindek, E.H.K. Stelzer, and S.W. Hell. Two new high-resolution confocal fluorescence microscopies (4pi, theta) with one- and two-photon excitation. In J.B. Pawley, editor, *Handbook of Biological Confocal Microscopy*, pages 417–430. Plenum Press, New York, 1995.
- [ML92] M. Marko and A. Leith. Contour-based 3D surface reconstruction using stereoscopic contouring and digitized images. In A. Kriete, editor, *Visualization in Biomedical Microscopies*, pages 45–73. VCH, Weinheim, New York, Basel, Cambridge, 1992.
- [MLB98] H. Mayer, I. Laptev, and A. Baumgartner. Multi-scale and snakes for automatic road extraction. In *5th European Conference on Computer Vision*, volume I, pages 720–733, 1998.
- [Mos92] V.A. Moss. Acquisition and visualization of serial section images. In A. Kriete, editor, *Visualization in Biomedical Microscopies*, pages 19–43. VCH, Weinheim, New York, Basel, Cambridge, 1992.
- [MSS92] D. Meyers, S. Skinner, and K. Sloan. Surface from contours. *ACM Transactions on Graphics*, 11(3):228–258, 1992.
- [MV98] J.B.A. Maintz and M.A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2:1–36, 1998.
- [OBS92] B. Okabe, B. Boots, and K. Suguhara. *Spatial tessellations — Concepts and Applications of Voronoi Diagrams*. John Wiley, New York, 1992.
- [OPC96] J.-M. Oliva, M. Perrin, and S. Coquillart. 3D reconstruction of complex polyhedral shapes from contours using a simplified generalized voronoi diagram. In *Eurographics '96*, pages C 397–408, 1996.

- [Pea58] L.D. Peachey. A study of section thickness and physical distortion produced during microtomy. *Journal of biophysical and biochemical Cytology*, 4:233 ff., 1958.
- [PG82] W.J. Perkins and R.J. Green. Three-dimensional reconstruction of biological sections. *Journal of Biomedical Engineering*, 4:37–43, 1982.
- [PS88] J.P. Preparata and M.I. Shamos. *Computational Geometry, an Introduction*. Springer-Verlag, New York, 1988.
- [Rei67] L. Reimer. *Elektronenmikroskopische Untersuchungs- und Präparationsmethoden*. Springer-Verlag, Berlin, Heidelberg, New York, 2 edition, 1967.
- [Rei97] L. Reimer. *Transmission Electron Microscopy*. Springer-Verlag, Berlin, Heidelberg, New York, 4 edition, 1997.
- [Rus90] J.C. Russ. *Computer-Assisted Microscopy*. Plenum Press, New York, London, 1990.
- [SE96] K. Sätzler and R. Eils. 3D-Bildfolgen zur Auflösungsverbesserung in der Fluoreszenzmikroskopie. In *Mustererkennung 1996*, pages 92–105, Berlin, Heidelberg, New York, Barcelona, Budapest, Hongkong, London, Mailand, Paris, Santa Clara, Singapur, Tokio, September 1996. Deutsche Arbeitsgemeinschaft für Mustererkennung, Springer-Verlag.
- [SE97] K. Sätzler and R. Eils. Resolution improvement by 3D reconstructions from tilted views in axial tomography and confocal theta microscopy. *Bioimaging*, 5:171–182, 1997.
- [Sei90] R. Seidel. Linear programming and convex hulls made easy. In *6th Annual ACM Symposium on Computational Geometry*, pages 211–215, 1990.
- [SH75] M.I. Shamos and D. Hoey. Closest point problems. In *Proc. 16th Ann. IEEE Symposium on Foundations of Computer Science*, pages 151–162, 1975.
- [SH98] G.M.G. Shepherd and K.M. Harris. Three-dimensional structure and composition of CA3→CA1 axons in rat hippocampal slices: Implications for presynaptic connectivity and compartmentalization. *Journal of Neuroscience*, 20(18):8300–8310, 1998.
- [Sjö58] F.S. Sjöstrand. Ultrastructure of retinal rod synapses of the guinea pig eye as revealed by three-dimensional reconstructions from serial sections. *Journal of Ultrastruct. Research*, 2:122–170, 1958.
- [SP58] F.S. Sjöstrand and A. Polson. Macrocrystalline patterns of closely packed poliovirus particles in ultrathin sections. *Journal of Ultrastruct. Research*, 1:365 ff., 1958.
- [SS97] T. Schikorski and C.F. Stevens. Quantitative ultrastructural analysis of hippocampal excitatory synapses. *Journal of Neuroscience*, 17(15):5858–5867, 1997.
- [TBM<sup>+</sup>99] W. Tvaruskó, M. Bentele, T. Misteli, R. Rudolf, C. Kaether, D.L. Spector, H.H. Gerdes, and R. Eils. Time-resolved analysis and visualization of dynamic processes in living cells. *Proceedings of the National Academy of Science, USA*, 96:7950–7955, 1999.
- [Wat81] D.F. Watson. Computing the n-dimensional delaunay tessellation with application to voronoi polytopes. *The Computer Journal*, 24:167–172, 1981.
- [Wat00] A. Watt. *3D Computer Graphics*. Addison-Wesley, Harlow, 3 edition, 2000.
- [Wei98] J. Weickert. *Anisotropic Diffusion in Image Processing*. Teubner, Stuttgart, 1 edition, 1998.

- [Wer94a] J. Wernecke. *The Inventor Mentor*. Addison–Wesley, Reading (Massachusetts), Menlo Park (California), New York, Don Mills (Ontario), Wokingham (England), Amsterdam, Bonn, Sydney, Singapore, Tokyo, Madrid, San Juan, Paris, Seoul, Milan, Mexico City, Taipei, 1994.
- [Wer94b] J. Wernecke. *The Inventor Toolmaker*. Addison–Wesley, Reading (Massachusetts), Menlo Park (California), New York, Don Mills (Ontario), Wokingham (England), Amsterdam, Bonn, Sydney, Singapore, Tokyo, Madrid, San Juan, Paris, Seoul, Milan, Mexico City, Taipei, 1994.
- [Wil90] T. Wilson. Confocal microscopy. In T. Wilson, editor, *Confocal Microscopy*, pages 1–64. Academic Press Inc., New York, 1990.
- [WK55] R.C. Williams and F. Kallman. Interpretation of electron micrographs of single and serial sections. *Journal of biophysical and biochemical Cytology*, 1:301 ff., 1955.
- [WND98] M. Woo, J. Neider, and T. Davis. *OpenGL Programming Guide*. Addison–Wesley, Amsterdam, 1998.
- [WWB<sup>+</sup>99] L.G. Wu, R.E. Westenbroek, J.G.G. Borst, W.A. Catterall, and B. Sakmann. Calcium channel types with distinct presynaptic localization couple differentially to transmitter release in single calyx–type synapses. *Journal of Neuroscience*, 19(2):726–736, 1999.
- [YF87] K. Yamamoto and H.D. Fahimi. Three–dimensional reconstruction of a peroxisomal reticulum in regenerating rat liver: evidence of interconnections between heterogeneous segments. *Journal of Cell Biology*, 105:713–722, 1987.
- [ZE60] T. Zelander and R. Ekholm. Determination of the thickness of electron microscopical sections. *Journal of Ultrastruct. Research*, 4:413 ff., 1960.

## Lebenslauf

Name	Sätzler
Vorname	Kurt
geboren am	12.05.1969
Geburtsort	Schwaigern/Württemberg
Eltern	Kurt Sätzler Erna Sätzler, geb. Weissenstein
Geschwister	Rita Hall, geb. Sätzler Rainer Sätzler
1975-1979	Besuch der Sonnenberg-Grundschule in Schwaigern/Württemberg
1979-1988	Besuch des Robert-Mayer-Gymnasiums in Heilbronn am Neckar
3. Mai 1988	Abitur erfolgreich abgelegt
3. Oktober 1988	Beginn der Bundeswehrzeit in Tauberbischofs- heim
2. November 1988	Immatrikulation an der Ruprecht-Karls-Uni- versität in Heidelberg
8. Dezember 1988	Zurückstellung von der Bundeswehr nach dia- gnostizierter Kreuzbandruptur
18. Juli 1990	Diplom-Vorprüfung im Fach Physik an der Ruprecht-Karls-Universität in Heidelberg er- folgreich abgelegt
2. Dezember 1990	Tätigkeit als Freier Mitarbeiter der Firma IBM in Heidelberg
13. August 1991	Bekanntgabe der Wehrdienstunfähigkeit
30. September 1992	Ablegung des Sprachzeugnisses in der Sprache Spanisch am Zentralen Sprachlabor der Ru- precht-Karls-Universität in Heidelberg
1. Oktober 1992	Anstellung als ungeprüfte Hilfskraft am Inter- disziplinären Zentrum für wissenschaftliches Rechnen der Ruprecht-Karls-Universität in Heidelberg
12. April 1995	Diplomprüfung im Studiengang Physik an der Ruprecht-Karls-Universität in Heidelberg er- folgreich abgelegt

1. Mai 1995 Anstellung als geprüfte Hilfskraft am Interdisziplinären Zentrum für wissenschaftliches Rechnen der Ruprecht–Karls–Universität in Heidelberg
16. Juni 1995 Heirat mit Nilda Santacruz–Sätzler, geb. Santacruz Benitez
1. September 1995 Anstellung als geprüfte Hilfskraft am Institut für Humangenetik der Ruprecht–Karls–Universität in Heidelberg
1. Dezember 1995 Antritt des Promotionsstipendiums der Max–Planck–Gesellschaft in der Abteilung Zellphysiologie in Heidelberg
- Mai bis Juli 1996 Aufenthalt am Insitut Albert Bonniot in Grenoble, Frankreich, im Rahmen des Human Capital Mobility Projektes der EU
1. Oktober 1996 Anstellung am Interdisziplinären Zentrum für wissenschaftliches Rechnen der Ruprecht–Karls–Universität in Heidelberg im Rahmen des Projektes *Computergestützte Operationsplanung*
- April 1998 Präsentation von CAR auf der Analytica in München
- Oktober 1999 Präsentation von CAR auf der begleitenden Ausstellung des Neuroscience Meetings in Miami, USA

## Publikationen

1. H. Bornfleth, **K. Sätzler**, R. Eils, and C. Cremer. High precision distance measurements and volume conserving segmentation of objects near and below the resolution limit in three-dimensional confocal fluorescence microscopy. *Journal of Microscopy*, 189:118–136, 1998.
2. H. Bornfleth, D. Zink, **K. Sätzler**, R. Eils, and C. Cremer. Modellgestützte Segmentierung von Replikationsdomänen in dreidimensionalen konfokalen Mikroskopiebildern. In *Mustererkennung 1996*, pages 408–419, Berlin, Heidelberg, New York, Barcelona, Budapest, Hongkong, London, Mailand, Paris, Santa Clara, Singapur, Tokio, September 1996. Deutsche Arbeitsgemeinschaft für Mustererkennung, Springer-Verlag.
3. S. Dietzel, R. Eils, **K. Sätzler**, H. Bornfeld, A. Jauch, C. Cremer, and T. Cremer. Evidence against a looped structure of the inactive human X-chromosome territory. *Exp. Cell Res*, page 239, 1998.
4. R. Eils, K. Saracoglu, C. Munkel, J. Imhoff, **K. Sätzler**, E. Bertin, S. Dietzel, E. Schröck, T. Ried, T. Cremer, and C. Cremer. Three-dimensional imaging approaches and monte carlo simulations: development of tools to study the morphology and distribution of chromosome territories and subchromosomal targets in human cell nuclei. In P.C. Cheng, editor, *Zoological Studies*, volume 34, pages 7–10, New Jersey, London, Singapore, 1995. World Scientific Publishing Co., Ltd.
5. R. Eils, K. Saracoglu, and **K. Sätzler**. Morphological parameters, medial lines and skeletons computed in the 3D-voronoi diagram with applications to 3D CLSM images. In P.C. Cheng, P.P. Hwang, J.L. Wu, G. Wang, and H. Kim, editors, *Focus on Multidimensional Microscopy*, pages 34–56. World Scientific Publishing, Singapore, 1999.
6. R. Eils and **K. Sätzler**. Shape reconstruction from volumetric data. In B. Jähne, H. Haußecker, and P. Geißler, editors, *Handbook of Computer Vision and Applications*, volume 2, pages 791–815. Academic Press, San Diego, London, Boston, New York, Sydney, Tokyo, Toronto, 1 edition, 1999.
7. R. Eils, S. Uhrig, **K. Sätzler**, K. Saracoglu, A. Bolzer, J.-M. Chassery, M. Ganser, and M.R. Speicher. An optimized and fully

- automated system for fast and accurate identification of chromosomal rearrangements by multiplex-FISH (M-FISH). *Cell Genetics and Cytogenetics*, 82:160–171, 1998.
8. M. Grabenbauer, **K. Sätzler**, E. Baumgart, and H.D. Fahimi. Three-dimensional ultrastructural analysis of peroxisomes in HepG2 cells: Absence of a peroxisomal reticulum but evidence of close spatial association with the endoplasmic reticulum. *Cell. Biochem. Biophys.*, 2000. in press.
  9. J. Lübke, A. Roth, O. Ohana, **K. Sätzler**, M. Frotscher, and B. Sakmann. Three-dimensional reconstruction of synapses onto layer 5 pyramidal neurons in rat somatosensory cortex from serial thin sections. In *Göttingen Neurobiology Report*, volume II, page 635, 1999.
  10. J.H.R. Lübke, **K. Sätzler**, L. Söhl, A. Roth, R. Eils, M. Frotscher, and B. Sakmann. The calyx of held: 3D reconstruction and quantitative analysis of its morphology. In *Neuroscience Meeting 2000*. Society of Neuroscience, 2000. submitted.
  11. **K. Sätzler** and R. Eils. 3D-Bildfolgen zur Auflösungsverbesserung in der Fluoreszenzmikroskopie. In *Mustererkennung 1996*, pages 92–105, Berlin, Heidelberg, New York, Barcelona, Budapest, Hongkong, London, Mailand, Paris, Santa Clara, Singapur, Tokio, September 1996. Deutsche Arbeitsgemeinschaft für Mustererkennung, Springer-Verlag.
  12. **K. Sätzler** and R. Eils. Resolution improvement by 3D reconstructions from tilted views in axial tomography and confocal theta microscopy. *Bioimaging*, 5:171–182, 1997.

## Danksagung

Am Ende einer jeden Arbeit dürfen natürlich all die Personen nicht unerwähnt bleiben, die — jede auf ihre Weise — zum Gelingen der vorliegenden Arbeit beigetragen haben.

Als erstes möchte ich mich bei Prof. Dr. B. Sakmann bedanken, der diese Arbeit mit seiner Zielsetzung, der dreidimensionalen Rekonstruktion von synaptischen Kontakten, und deren Fernziel, diese morphologischen Auswertungen mit Simulationsrechnungen zu verbinden, geprägt und initiiert hat. Schon die Erfahrung, am Alltag einer Arbeitsgruppe partizipieren zu dürfen, die sich absolut der Grundlagenforschung verschrieben hat, die unglaublich international besetzt und mit vielen hoffnungsvollen Nachwuchswissenschaftler gesegnet ist, ist ein wichtiges Erlebnis, für das ich sehr dankbar bin. Dieses Umfeld ermöglichte mir auch zahlreiche Auslandsaufenthalte, die ich allesamt nicht missen möchte. Besonderen Dank gilt in der Gruppe von Prof. Dr. B. Sakmann ebenfalls Dr. G. Borst, K. Bauer und A. Roth. Dr. G. Borst hat nicht nur die Arbeit durch seine elektrophysiologische Expertise speziell bei der *Calyx von Held* mitbetreut, sondern war auch durch seine ruhige Art vor allem in schwierigeren Zeiten ein wichtiger Ansprechpartner. K. Bauer, der die technischen Fäden hinter den Kulissen ruhig zusammenhält, danke ich für seine Unterstützung beim Kauf all der technischen Geräte, die wir für dieses Mammutprojekt benötigt haben. Und *last but not least* danke ich A. Roth, der einer der ersten, und damit einer der wichtigsten Benutzer von CAR ist, welcher durch sein stetes Interesse und seine konstruktiven Kritiken auch ein wichtiger, motivierender Faktor ist.

Bedanken möchte ich mich auch bei Prof. Dr. Dr. C. Cremer, der sich als Spezialist in Sachen Lichtmikroskopie dieses etwas exotischen Themas angenommen hat, und somit diese interdisziplinäre Arbeit ermöglicht hat. Er ist für mich einer der wichtigen Wegbereiter, die solche interdisziplinären Forschungsrichtungen wie die Biophysik und Bioinformatik propagiert und ermöglicht haben.

Interdisziplinarität ist auch eines der großen Verdienste, um die sich die nächste Person, der mein Dank gilt, verdient gemacht hat. So möchte ich mich bei Prof. Dr. Dr. h.c. mult. W. Jäger dafür bedanken, daß er nicht nur diese Doktorarbeit mit großem Interesse wohlwollend verfolgt hat, sondern auch mit dem Interdisziplinären Zentrum für Wissenschaftliches Rechnen eine Institution ins Leben gerufen hat, dessen

Struktur und Konzeption Vorbild für viele wissenschaftliche Institute der Zukunft sein wird. In diesem fruchtbaren Umfeld war dieser Arbeit ein guter und tragfähiger Boden bereitet, um zum Erfolg gelangen zu können. An dieser Stelle möchte ich mich bei der Supportgruppe des IWR, besonders bei M. Ridinger, S. Schnadt und B. Przywara dafür bedanken, daß sie durch ihre exzellente Arbeit erst viele Probleme überbrückbar und lösbar gemacht haben.

Einem Sproß, der diesem Umfeld des IWR entsprungen ist, gilt mein nächster Dankesgruß. So danke ich Dr. R. Eils dafür, daß er nicht nur meine Diplom- und Doktorarbeit mitbetreut, sondern auch als Freund stets ein offenes Ohr für meine Anliegen gefunden hat. Ich habe mit Freude beobachtet, wie sich die kleine Arbeitsgruppe, die Roland Eils zu Anfängen meiner Diplomarbeit leitete, und die im wesentlichen aus den zwei Diplomanden K. Saracoglu und mir bestand, in den letzten Jahren zu einer eigenständigen Abteilung des DKFZ Heidelbergs gemausert hat. Ich wünsche dieser Gruppe und Dr. R. Eils weiterhin viel Erfolg.

Gerne denke ich auch an meine Kurzaufenthalte am Institut Albert Bonniot in Grenoble, Frankreich, zurück, wo ich zusammen mit Dr. R. Eils den Grundstein zur Voronoibibliothek gelegt habe, die nicht nur zur Umsetzung des 3D Rekonstruktionsalgorithmus wichtig war, sondern auch den Boden für die Skelettierungsalgorithmen und den ersten Farbsegmentierungsalgorithmus für das *M-Fish*-Projekt bereitet hat. Dabei gilt mein besonderer Dank Prof. Dr. M. Robert-Nicoud, der diesen Aufenthalt organisiert und ermöglicht hat.

Ganz wesentlich für die Umsetzung dieses Projektes war die Zusammenarbeit mit Dr. J. Lübke, der die Kontakte zwischen dem Max-Planck-Institut für medizinische Forschung in Heidelberg und dem Anatomischen Institut von Prof. Dr. M. Frotscher in Freiburg geknüpft hat, sowie die Präparation, Bildaufnahme und Umfahrung der äußeren Umrandungen der *Calyx von Held* mit Hilfe von CAR organisiert und geleitet hat. Mein Dank gilt aber nicht nur den beiden Leitern, sondern auch der technischen Assistentin S. Nestel und dem cand. med. L. Söhl. So hat S. Nestel mit ihrer ruhigen Art und der unglaublichen Gabe, derartige Serienschnitte im wahrsten Sinne des Wortes in Serie herstellen zu können, den Grundstein zu dieser Untersuchung gelegt. L. Söhl danke ich für seine Ausdauer bei der Bedienung und seine Anregungen zur Verbesserung von CAR, die aus diesem Programm eine gebrauchsfertige und benutzerfreundliche Software werden lassen. Außerdem denke ich immer gerne an die angenehmen Kurzaufenthalte, die ich im Gästezimmer des Anatomischen Institutes verbringen durfte, zurück. Diese Phasen gehörten zu den produktivsten Phasen des Projektes.

Weiterhin danke ich Dr. R. Schröder für seine Hilfestellungen in puncto Elektronenmikroskopie von biologischen Präparaten, sowie für den erfolgreichen Großgeräteantrag für den Scanner SCAI von Zeiss

der Abteilung Biophysik am Max-Planck-Institut für medizinische Forschung in Heidelberg, von dem auch dieses Projekt profitieren konnte.

Prof. Dr. H. Schwegler danke ich für das erste Bildmaterial, mit dem wir CAR getestet und entwickelt haben, ohne welches diese Arbeit vielleicht nie zustande gekommen wäre. Auch für sein stetes Interesse am Fortgang dieses Projektes bin ich ihm sehr dankbar.

J. Fahlke und K. Saracoglu danke ich dafür, daß sie mir einen Großteil der mühsamen Scanarbeiten des Negativfilmmaterials abgenommen haben. Diese Zusammenarbeit hat stets viel Spaß und Freude bereitet.

Der Verwaltung des Max-Planck-Institutes für medizinische Forschung in Heidelberg danke ich für die einfach unglaublich schnellen Reisekostenabrechnungen und die stete Hilfsbereitschaft.

Dank geht auch an die Ruprecht-Karls-Universität Heidelberg, die es ermöglicht hat, CAR auf der Analytica 1998 in München präsentieren zu können. Gleichfalls danke ich dem Team der Firma CHROMA, die es uns im Sommer 1999 beim Neuroscience Meeting in Miami ermöglicht haben, eine Ecke ihres Ausstellungsstandes für die Präsentation von CAR nutzen zu dürfen, wofür uns SGI unbürokratischer Weise zwei Octane-Workstations zur Verfügung gestellt hat.

*Last but not least* danke ich allen Familienmitgliedern, Freunden, und vor allem meiner Frau für die stete Bereitschaft, mir ein offenes Ohr zu spenden, was für mich eine sehr wertvolle Stütze über den gesamten Verlauf des Projektes war. All denjenigen sei gesagt: Das nächste 100+X-Festival kommt bestimmt!

## ANHANG A

### Die OpenInventor–Bibliothek

`OpenInventor` ist eine Klassenbibliothek, die in der im Rahmen dieses Projektes entwickelten Software intensiv eingesetzt wurde, nicht nur, um die erzeugten 3D Geometrien darzustellen, sondern auch um die allgemeinen Korrekturtransformationen zu erfassen und in einer Datei abzuspeichern. Zum besseren Verständnis der angewandten Transformationen soll hier eine Kurzeinführung in den Aufbau und die Funktionsweise von `OpenInventor` gegeben werden.

`OpenInventor` ist eine objektorientierte Klassenbibliothek, die auf `OpenGL` aufsetzt. `OpenGL` ist eine 3D–Graphikbibliothek, die einen ca. 250 Befehle umfassenden Satz an Bibliotheksfunktionen zur Verfügung stellt, welche eine Schnittstelle zur Hardware von Graphikkarten bilden. `OpenGL` erlaubt dadurch eine hardwarenahe Programmierung von Graphikroutinen (siehe [Cla97] und [WND98]).

Im Gegensatz zu `OpenGL`, das im wesentlichen dreidimensionale Graphikprimitiven zur Verfügung stellt, liegt der Schwerpunkt von `OpenInventor` auf der Organisation und Beschreibung von 3D Objekten. Der Benutzer von `OpenInventor` beschäftigt sich also i.a. nicht mit dem konkreten Zeichenvorgang, sondern versucht 3D Objekte zu organisieren und anzordnen.

`OpenInventor` verfügt über 4 Grundelemente:

1. Basisobjekte, die in einer Datenbank organisiert sind
2. Basistemplates (*Nodekits*), die eine Erweiterung der Datenbank zulassen
3. Manipulatoren
4. Editoren und Komponenten

#### Die Inventor Datenbank

Als Basisobjekte werden 5 Gruppen von Objekten zur Verfügung gestellt. Diese sind im Einzelnen

1. Geometrien (*Shapes*)
2. Eigenschaften (*Properties*)
3. Gruppen (*Groups*)
4. Motoren (*Engines*)
5. Sensoren (*Sensors*)

Die Geometrien (*Shapes*) dienen dazu, dreidimensionale Objekte, wie Punkte, Linien, Kugeln, Quader, Kegel, triangulierte Oberflächen und

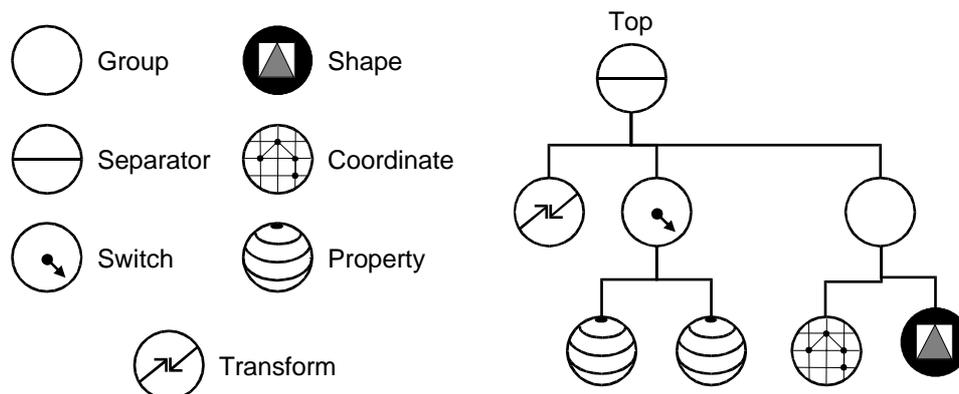


ABBILDUNG A.0.1. Dargestellt sind die Symbole der Inventor-Knoten, die in CAR zum Einsatz gekommen sind. Rechts ist ein Beispiel für einen Szenegraphen dargestellt, der von oben nach unten und von links nach rechts von Aktionen abgearbeitet wird.

Freiformflächen zu beschreiben. Die *Properties* beschreiben dabei Eigenschaften, die auf das Zeichnen und Darstellen der Geometrien wesentlichen Einfluß hat. Sie beschreiben Materialeigenschaften, wie Farben und Texturen, aber auch Transformationen. All diese Angaben beeinflussen den Zustand (*State*) des OpenGL Graphikkontextes. Die Gruppenknoten dienen hauptsächlich zur Organisation der ungeordneten Inventorobjekte in einen azyklischen, gerichteten Graphen. Motoren (*Engines*) erlauben es, Animationen in das Konzept der Objektbeschreibungen mit aufzunehmen. Sensoren (*Sensors*) erlauben die Reaktion und Interaktion der Objekte auf gewisse Änderungen in der Inventor-Datenbank.

In Abbildung (A.0.1) sehen wir diejenigen Basiselemente, die in CAR zum Einsatz kommen. Zum einen sind es natürlich Geometrien, die die Markierung, die Ebenenbilder und die Rekonstruktionen darstellen. Die wichtigsten Objekte für die Einführung der allgemeinen linearen Korrekturen sind die linearen Transformationen, deren Einfluß durch die Art der Einordnung in den Szenegraphen (*Scenegraph*) dosiert werden kann (siehe Kapitel 2).

Als Szenegraphen verstehen wir den gerichteten Graphen, der entsteht, wenn die Datenbankelemente von Inventor in Gruppenknoten eingeordnet werden. Der Szenegraph legt nun eine Abarbeitungsordnung für Aktionen (*Actions*) fest. Als Aktionen, die entlang eines Szenegraphen abgearbeitet werden können, verstehen wir u.a. das Selektieren und Zeichnen der Objekte. Wege innerhalb des Graphen werden als Pfade (*Paths*) bezeichnet.

Wir unterscheiden 3 Hauptformen von Gruppenknoten:

**Group:** unter diesem werden weitere Knoten als Kinder (*Childs*) angeordnet, die alle Einfluß auf den aktuellen Zustand des

OpenGL-Graphikkontextes haben können. Darin ausgeführte Transformationen haben also Einfluß auf Knoten, die dem *Group*-Knoten nachgestellt sind. Aktionen werden an alle Kinder des Gruppenknotens weitergereicht.

**Separator:** diese Form des Gruppenknotens speichert den Zustand von OpenGL ab, der vor der Ausführung einer Aktion in der Gruppe bestanden hat, und stellt diesen nach Verlassen der Gruppe wieder her. Nachfolgende Knoten bleiben also von den Aktionen, die im *Separator* ausgeführt werden, unbeeinflusst.

**Switch:** Der Switch verhält sich wie ein *Group*-Knoten, was den Zustand des OpenGL-Graphikkontextes betrifft. Allerdings kann gewählt werden, welches der Kinder eine Aktion ausführen soll (keines, eines oder alle).

Der mittels der Gruppen-Knoten organisierte Szenegraph legt eine eindeutige Reihenfolge für das Abarbeiten von Aktionen fest. Die Aktionen werden von oben nach unten und von links nach rechts abgearbeitet (siehe Abbildung (A.0.1)).

### Nodekits

Nodekits stellen bestimmte Anordnungen von Basisobjekten zur Verfügung, die als Schablonen oder Templates in eigenen Weiterentwicklungen von Inventor eingesetzt werden können. Nähere Informationen hierzu findet man im *Inventor Toolmaker* (siehe [Wer94a]).

### Manipulatoren

Manipulatoren sind 3D Objekte, die im Szenegraphen von OpenInventor dargestellt werden können. Besonderheit dieser Objekte ist es, daß sie die Interaktion mit dem Benutzer erlauben. Manipulatoren sind daher eng mit anderen 3D Objekten verknüpft und erlauben die Interaktion mit Datenfelder der Objekte im allgemeinen. Im besonderen werden die Transformationsparameter, wie Skalierungen, Translationen und Rotationen mit den Manipulatoren eingestellt (siehe Abbildung (2.7.3) auf Seite 40).

### Komponenten

Komponenten sind für das Zusammenspiel von OpenInventor mit dem Fenstersystem zuständig und bilden daher den einzigen, betriebssystemabhängigen Anteil von OpenInventor. Die OpenInventor-Klassenbibliothek garantiert jedoch eine Vereinheitlichung des Aufbaus dieser Komponenten. Zu den wesentlichen Komponenten zählen die *Viewer*, die einfache Manipulationen der Kameraobjekte zulassen. Dadurch ist ein einfaches und sehr intuitives Navigieren in drei Dimensionen möglich (siehe das Hauptfenster von CAR in Abbildung (2.7.1))

auf Seite 38). Andere hochentwickelte Komponenten sind die Editoren, die es erlauben, per Interface die Datenwerte von Materialien oder Transformationen zu manipulieren.

Diesen hochentwickelten Komponenten sind 3 wesentliche Basis-komponenten zugrunde gelegt, die für den Einsatz auf anderen Fenstersystemen wichtig sind. Dies ist zum einen das Zeichenfenster (*Render Area*), in welchem die `OpenGL`-Befehle ausgeführt und die Szene dargestellt werden kann. Damit dieses Zeichenfenster im Mehrfensterbetrieb eingesetzt werden kann, wird ein Interface zu den Userinteraktionen, die *Event Translations*, und eine unendlich laufende Schleife implementiert (*Mainloop*), die das stetige Abarbeiten der Userinteraktionen erlaubt.

Zum Abschluß sei noch darauf verwiesen, daß `OpenInventor` sehr wesentlich den Standard von VRML geprägt hat. Beiden Bibliotheken liegt somit das gleiche Konzept zugrunde. Für ein vertiefendes Studium der `OpenInventor`-Bibliothek seien die Bücher ([**Wer94a**] und [**Wer94b**]) empfohlen.

## Schnelle Rendertechniken

In den Anfängen der Computergraphik wurden realistische Szenen stets mit Softwarerendertechniken, wie dem Raytracing oder dem Radiosity generiert (siehe [Wat00]). Realistische Darstellungen von größeren Datenmengen auf Polygonbasis war zu den Anfängen nur auf speziellen Graphikkarten vor allem von Silicon Graphics möglich. Sie verwendeten spezielle Prozessoren für die Ausführung der Transformationsberechnungen und die Rasterung des Bildes.

Außerdem stellten nur die High-End Graphikkarten zu dieser Zeit speziellen Texturspeicher zur Verfügung, der über einen schnellen Bus an das Rasterboard angebunden war. Durch das Verfahren des sog. Texturemappings wurde es möglich, realistische Darstellung in Echtzeit zu generieren. Es wurden sogar Schatten und Reflexionen mit Texturemapping-Tricks nachempfunden, was bisher nur aufwendigen Softwarerendern vorbehalten war. Diese Techniken verwenden zu können, bedeutete gleichzeitig tief in den Geldbeutel greifen zu müssen.

Erst mit der Einführung der 02 von Silicon Graphics, konnte durch die sog. *Crossbar Architecture* auf die Verwendung von sündhaft teurem Texturspeicher verzichtet werden; hier wird der Hauptspeicher als erweiterbarer Texturspeicher verwendet. Diese Architektur findet man ebenfalls noch bei der *NT Visual Workstation* von Silicon Graphics vor (Stand 1999). Ähnliche Ziele hat man bei der Einführung des *Advanced Graphics Port* (kurz: *AGP*) verfolgt.

Getrieben durch die Anforderungen im Spielesektor, die immer realistischere 3D Landschaften anbieten wollen, halten nun im Low-End Graphikkartenmarkt erste spezialisierte Graphikprozessoren Einzug, die die CPU entlasten sollen. Außerdem ist durch das Sinken der Speicherpreise ein mehr an Texturspeicher kein Thema mehr. So ist abzusehen, daß CAR — zumindest für mittlere Bilddatenaufkommen — auch auf Standard-PC's zum Einsatz kommen kann. Die Leistungsfähigkeit heutiger Graphikkarten zeigen die anhand des öffentlich zugänglichen Graphiktestes *Viewperf 6.1.1* erzielten Bildwiederholraten (siehe Abbildung B.0.1).

In der vorliegenden Arbeit ging es nicht primär um die Echtzeitdarstellung von 3D Geometrien, sondern um die Darstellung großer bis größter Bilder, die allgemeinen linearen Transformationen ausgesetzt sind (siehe Kapitel 2). Um hierfür die Power der Graphikkarten voll

## Performace Overview – Viewperf 6.1.1.

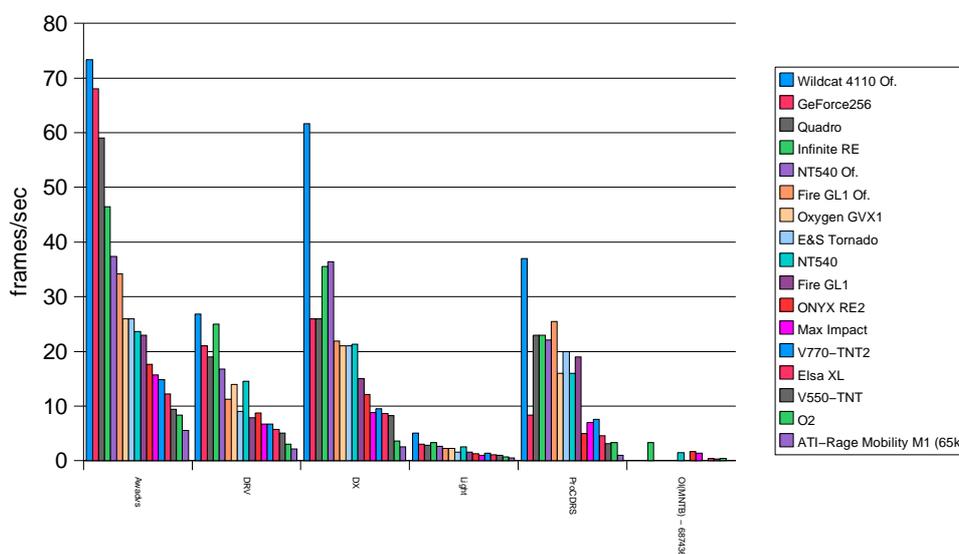


ABBILDUNG B.0.1. In diesem Diagramm sind die *Specs* der *Viewperf 6.1.1* Graphiksuite mit den bis zum Zeitpunkt dieser Arbeit verfügbaren Graphikkarten am IWR selbst getestet worden. Die neue Generation von Graphikkarten mit dem **GeForce256**-Graphikprozessor scheinen hierbei die etablierten und extrem teuren Graphikkarten des High-End Bereiches von Silicon Graphics auszubooten. Höchste Gprahikleistung für den Heim-PC scheint erschwinglich geworden zu sein.

ausnutzen zu können, wurde der betreffende Code in **OpenGL** programmiert (siehe Kapitel A).

In **OpenGL** ist das Rendern von Bildern durch Verwendung sog. *Texture Maps* möglich. Um jedoch auch große Bilder in Echtzeit rendern zu können, muß man sich mehrerer Techniken bedienen, die eine sinnvolle Auswahl der zu rendernden Bilder derart treffen, daß der Betrachter am Bildschirm möglichst keinerlei qualitative Einbußen wahrnimmt.

Da **OpenGL** nur Texturen zuläßt, deren Bilddimensionen Potenzen von 2 sind und eine (hardwareabhängige) maximale Obergrenze für Texturen festlegt, muß man sich des sog. *Tilings* bedienen (siehe Abbildung (B.0.2(a))). In diesem Verfahren werden die Bilder in kleinere Subbilder unterteilt, die den Nebenbedingung von **OpenGL** für Texturen Rechnung tragen. Hat unser digitales Ausgangsbild  $f(i, j)$  die Bildgröße  $N \times M$  und beträgt die maximale Obergrenze für Texturen  $2^{o_{\max}}$  ( $i, j \in \mathbb{N}_0$  und  $o_{\max} \in \mathbb{N}$ ). Sei weiterhin  $b(k), k \in \mathbb{N}_0$  die Binärdarstellung der Zahl  $k$  derart, daß  $k = \sum_i b_i(k) 2^i$ . So bestimmt sich die

Anzahl der Bilder  $n_{\text{tile}}$  nach folgender Formel:

$$n_{\text{tile}} = B(N) \cdot B(M) \text{ mit } B(k) = \sum_{i \geq o_{\text{max}}} b_i(k) 2^{i-o_{\text{max}}} + \sum_i b_i(k)$$

Bei einem Bild der Größe 511x511 führt dies schon zu einer Anzahl  $n_{\text{tile}} = 8 \cdot 8 = 64$  Subbildern, sofern  $o_{\text{max}} \geq 8$  ist.

Die nächste wichtige Technik bedient sich der sog. Bildpyramiden oder des sog. *Mipmappings* (siehe Abbildung (B.0.2(b))): Bei diesem Vorgehen werden zu jedem Bild herunterskalierte Bilder verschiedener Auflösungsstufen generiert. Die neuen Pixelwerte können dabei durch verschiedene Interpolationsverfahren errechnet werden (*Nearest Neighbour*, *Bilinear Interpolation*). Für welche der Interpolationsverfahren man sich in der Praxis entscheidet, hängt hauptsächlich vom Vorverarbeitungsaufwand und der daran gekoppelten Rechenzeit ab. Die Anzahl der generierten *Mipmaps*  $n_{\text{mip}}$  bestimmt sich bei einem Bild der Dimension  $2^N \times 2^M$  wie folgt:

$$n_{\text{mip}} = 2^{\max(N,M)}.$$

Nun wird abhängig von der Größe des Bildes auf dem Bildschirm diejenige *Mipmap* ausgewählt, bei der der Betrachter keinerlei Qualitätsverlust zum Original mehr ausmachen kann. Die Pixelgröße der *Mipmap* sollte hierbei in etwa der Pixelgröße der Bildschirmdarstellung entsprechen.

Die letzte Technik (siehe Abbildung (B.0.2(c))), die für eine schnellere Bildschirmdarstellung sorgen kann, ist das sog. *Clipping*. Hierbei wird anhand des Betrachterschnittes bestimmt, welche der Bilder und Subbilder in diesem zu sehen sind. Liegen Bilder außerhalb des Betrachterfeldes (*Viewports*), werden diese nicht an die Renderengine weitergeleitet und sparen somit Rechenzeit.

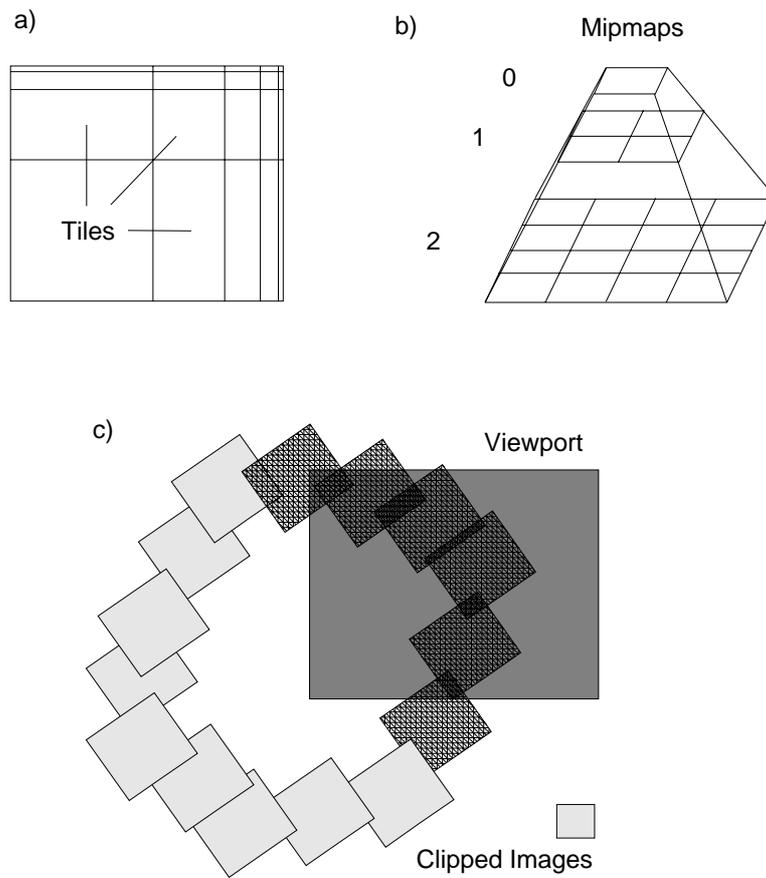


ABBILDUNG B.0.2. Symbolische Darstellung der 3 Rendertechniken: (a) *Tiling*, (b) *Mipmapping* und (c) *Clipping*.

## ANHANG C

### Die Datenstruktur der Voronoibibliothek

```
#ifndef _KSSLICEDEL_H
#define _KSSLICEDEL_H

#include <vorlist.h>

#ifdef __cplusplus
extern "C" {
#endif

/* Defines to access labels as bitfields */
#define KsGetKsCntStDLabel( n, pt)      (((3<<(2*(n)))&((pt)->lbl))>>(2*(n)))
#define KsGetKsCntStSumLabel( pt)      (((15<<8)&((pt)->lbl))>>8)
#define KsGetKsCntStTriLabel( n, pt)   (((3<<(16+2*(n)))&((pt)->lbl))>>(16+2*(n)))
#define KsGetKsCntStIOLabel( pt)      (((3<<24)&((pt)->lbl))>>24)
#define KsSetKsCntStDLabel( val, n, pt) ((pt)->lbl)=(((pt)->lbl)&\
(~(3<<(2*(n))))|(((val)&3)<<(2*(n))))
#define KsSetKsCntStSumLabel( val, pt) ((pt)->lbl)=(((pt)->lbl)&\
(~(15<<8))|(((val)&15)<<8))
#define KsSetKsCntStTriLabel( val, n, pt) ((pt)->lbl)=(((pt)->lbl)&\
(~(3<<(16+2*(n))))|(((val)&3)<<(16+2*(n))))
#define KsSetKsCntStIOLabel( val, pt) ((pt)->lbl)=(((pt)->lbl)&\
(~(3<<24))|(((val)&3)<<24))

/* Modified VLIB struct for triangulation of 2D contours */
typedef struct _kscntpt{
    double point[3];          /* coordinates */
    int num_s;               /* number of a corresponding delaunay element */
    struct _kscntlblpt *pub; /* label for triangulation */
} KSCNTPT;

/* different cases for different values of sn and cn:
    sn>0, cn>0, pr,nxt != NULL => regular contour point
    sn>0, cn>0, pr,nxt == NULL => inner point of contour
    sn>0, cn=0, pr,nxt == NULL
        => interpolating point between slices sn, sn+1
    sn>0, cn<0, pr,nxt == NULL => point outside of contour
    sn==0 for lowest slice, sn==ns for top most slice
    sn<0 for outside points of bounding box
*/
typedef struct _kscntlblpt{
    KSCNTPT *pr, *nxt; /* pointer to previous/next point on contour */
    long cn;          /* contour number */
    long sn;          /* slice number */
    short lu;         /* connectivity label (lowup) */
    short lbl;        /* label used during reconstruction process */
} KSCNTLBLPT;

/* Delaunay-(neighbourhood-)structure */
typedef struct _kscntst{
    double point[3];          /* coordinates of circumcircles */
    KSCNTPT *pt_gc[4];       /* points forming the (empty) circumcircle */
}
```

```

    struct _kscntst *pt_sv[4]; /* neighbour delaunay elements */
    int    block;             /* for blocking delaunay elements */
    struct _kscntlblst *pub; /* label for delaunay elements */
} KSCNTST;

typedef struct _kscntlblst{
    long lbl;
    long on;
} KSCNTLBLST;

typedef struct _kscnt{
    KSCNTPT    *pt;
    int        n_pt;
    int        n_pt_max;
    KSCNTST    *ps;
    int        n_ps;
    int        n_ps_max;
    PT_LISTE   psh;
    double     box[3][2];
    PT_LISTE   cslst; /* list of all contours (per slice) = concatenated list! */
    double     zBox[3][2];
    double     *zAdd;
    PT_LISTE   zval; /* list of the zvalues for computing the slice diagramm */
    int        nc, ns;
    int        addStart, /* starting index of added points */
              nadd;     /* number of added points */
} KSCNT;

/* ... some more code is skipped here ... */

#ifdef __cplusplus
} /* END of extern */
#endif

```

Die oberen Definitionen und Deklarationen im abgedruckten Source-Code zeigen, wie die Bitfelder (siehe Abbildung (3.3.5) auf Seite 68) gesetzt werden. Die duale Beziehung von Voronoidiagramm zu Delaunaygraph spiegelt sich auch hier wieder: `KSCNTPT` bezeichnet die Punktemenge  $P$ , für die  $\mathcal{D}(P)$  und dementsprechend  $\mathcal{V}(P)$  berechnet werden sollen. `num_s` ist dabei der Index, der von der Knotenstruktur des Delaunaygraphen (`KSCNTPT`) in die Kantenstruktur des Voronoidiagramms (`KSCNTST`) zeigt und somit einen Link zwischen den beiden dualen Strukturen darstellt. Die Voronoikantenstruktur spiegelt dabei die Aussage 3.2.19 auf Seite 54 wieder: Jeder Voronoipunkt hat in 3 Dimensionen 4 Kantennachbarn und 4 erzeugende Punkte. Durch die Zeiger (`*pub`) in der jeweiligen Punkt- und Kantenstruktur ist es möglich, spezielle Label, die *nicht* für die Berechnung des PVD, aber für die weiterführende Verarbeitung und Auswertung derselben<sup>1</sup> benötigt werden, in die Datenstruktur „einzuhängen“.

---

<sup>1</sup>in diesem Falle die Erzeugung der 3D Rekonstruktion von Konturdaten