



FAKULTÄT FÜR
INFORMATIK

Interaktive 3D- und Graphvisualisierung von Protein-Clustern in Toponomdaten

Masterarbeit
von

Paul Klemm

Betreuer:

Prof. Dr.-Ing. Bernhard Preim

HD Dr. med. Walter Schubert

Dr.-Ing. Steffen Oeltze

Dipl.Ing. Reyk Hillert

Gutachter:

Prof. Dr.-Ing. Bernhard Preim

Prof. Dr.-Ing. Andreas Nürnberger

Vorgelegt am 21.12.2011

Für Heidi und Hubert, die mein Kompass sind.

Danksagung

Ich möchte allen Menschen danken, die mir im Laufe dieser Arbeit unterstützend zur Seite standen. Walter Schubert für seine außergewöhnliche Arbeit und seine inspirierenden Worte; seiner Arbeitsgruppe 'Molekulare Mustererkennung' für die herzliche Aufnahme in ihrer Mitte; Bernhard Preim für seine Hilfs- und Einsatzbereitschaft für seine Studenten; meiner Familie und meinen Freunden für mein Glück. Besonders möchte ich Steffen Oeltze danken, der mir kompetent, hilfsbereit und beruhigend den Weg gewiesen hat.

Erklärung der Selbstständigkeit

Hiermit versichere ich, Paul Klemm, die vorliegende Arbeit allein und nur unter Verwendung der angegebenen Quellen angefertigt zu haben.

Magdeburg, 21.12.2011

Paul Klemm

Inhaltsverzeichnis

1	Einführung	6
1.1	Motivation	7
1.2	Zielsetzung	7
1.3	Gliederung	8
2	Biomedizinische Grundlagen	9
2.1	Molekulare Netzwerke	9
2.1.1	Proteine	9
2.1.2	Toponom	10
2.1.3	Leitproteine	11
2.2	Fluoreszenzmikroskopie	12
2.3	Automatisierte Multi-Epitope-Ligand-Kartographie	14
2.4	Combinatorial Molecular Phenotypes	16
2.5	Zusammenfassung	19
3	Grundlagen und verwandte Arbeiten	20
3.1	Direkte Volumenvisualisierung	20
3.1.1	Daten	21
3.1.2	“Volume ray casting“-Algorithmus	22
3.2	3D-Interaktion	23
3.2.1	Wahrnehmungsprobleme in 3D	23
3.2.2	Navigation und Selektion von Objekten	24
3.2.3	Eingabegeräte für 3D-Interaktion	25
3.3	Graphvisualisierung von Toponomdaten	26
3.3.1	Graphentheoretische Grundlagen	27
3.3.2	Graphlayout	27
3.3.3	Interaktive Graphvisualisierung von Toponomdaten	28
3.4	Perzeptuell unterscheidbare Farben	29
3.5	Zusammenfassung	31
4	Entwurf der 3D- und Graphvisualisierung	32
4.1	Analyse der Forschungsabläufe	32
4.2	Anforderungsanalyse	33
4.2.1	Aktueller Stand der Arbeiten	34
4.2.2	Anforderungen und Zielstellungen	34
4.3	Vorhandene Visualisierung	35

4.4	Entwicklung der 3D-Visualisierungstechnik	37
4.4.1	Generierung des 3D-Volumens	37
4.4.2	Visualisierungsparameter und -variablen	38
4.4.3	Explorationswerkzeuge und Nutzung des Zeigers	41
4.4.4	Navigation im 3D-Raum	46
4.5	Epitop-Graphvisualisierung	47
4.5.1	Generierung des Graphen	48
4.5.2	Anordnung und Parameter	49
4.5.3	Interaktion mit dem Graphen	51
4.5.4	Generierung eines Vergleichsgraphen	52
4.6	Algorithmische Farbgenerierung	54
4.6.1	Entwicklung des Algorithmus	54
4.6.2	Anwendung	56
4.7	Zusammenfassung	57
5	Implementation der Visualisierungen	58
5.1	Dateiformat	58
5.2	Programmierungswerkzeuge	58
5.3	Einbindung der Visualisierungen in MultiCompare	60
5.3.1	Stand und Erweiterung der Software	60
5.3.2	Probleme von MultiCompare	61
5.4	User-Interface-Konzept	63
5.4.1	Entwurf der Benutzeroberfläche	63
5.4.2	Interaktion zwischen den Tools	65
5.4.3	Anordnung der Werkzeuge und Multimonitorlösungen	67
5.5	Zusammenfassung	68
6	Zusammenfassung und Ausblick	69
6.1	Bewertung der Ergebnisse	70
6.2	Folgen für die Forschung	71
6.3	Zukünftige Problemfelder	72
6.4	Schlusswort	75
	Literatur	79

1 Einführung

Während die Entschlüsselung des Erbguts des Menschen abgeschlossen ist, stellt die Dekodierung der molekularen Netzwerke der Zelle Biotechnologen weltweit weiterhin vor große Herausforderungen. Innerhalb dieser Muster verbirgt sich die Verschlüsselung aller Zellfunktionen [Schubert, 2004, S.47]. Das Wissen um die Beschaffenheit von Eiweißstrukturen in Zellen und Geweben im Kontext ihrer räumlichen Anordnung und zeitlicher Abfolge ist die Voraussetzung für eine Erkennung von krankhaften Veränderungen. Es ist nicht bekannt, wie die einzelnen Zellen miteinander in Verbindung treten und sich vernetzen, sodass komplexe hochspezialisierte funktionelle Strukturen bis hin zu einzelnen Organen entstehen. Wie wahren diese Komplexe ihre strukturelle Integrität? Wo und wie wird ihr Funktionsplan geschrieben?

Durch einen Bauplan für bestimmte Gewebearten können diagnostische Verfahren und Therapieformen entwickelt werden, die speziell jene Moleküle adressieren, die ausschlaggebend für eine Funktionsstörung sind. Um die einzelnen Bestandteile von komplexen Zellbereichen verstehen zu können, müssen diese explorierbar gemacht werden.

Eine Herangehensweise an dieses Problem wurde in Form einer neuen mikroskopischen Technologie als bildgebendes Verfahren entwickelt, das es erlaubt, eine große Zahl Eiweißmoleküle (*Proteinen*) in einzelnen Zellen oder sogar in ganzen Gewebeschnitten und Zellkulturen sichtbar zu machen [Schubert, 1990, 2003; Schubert et al., 2006; Schubert, 2010; Schubert et al., 2011]. Hieraus ergibt sich die Möglichkeit, eine Vielzahl von unterschiedlichen Proteinverknüpfungen in ihrer natürlichen Lage zu beobachten. Die Analyse dieser Daten haben ergeben, dass bei gleicher optischer Präsentation gesunde und kranke Gewebe unterschiedliche Proteinnetzwerke aufweisen. Es konnten so Proteine ermittelt werden, die auslösend für bestimmte Erkrankungen sind. So können diese gezielt bekämpft werden [Schubert et al., 2006].

Bislang wurden durch dieses neue bildgebende Verfahren hauptsächlich Zellen, Gewebsschnitte, Körperflüssigkeiten (Blut) und Zellkulturen analysiert. Mit steigendem Verständnis für die Daten und Fortschreiten der Technologie geht die Tendenz zu dreidimensionalen Darstellungen, die sich aus einer Anzahl von zweidimensionalen optischen Ebenen zusammensetzen, die einen festen kleinen Abstand zueinander haben. Die Visualisierung dieser Daten ist entscheidend für die Entschlüsselung von Proteinnetzwerken. So können die Wechselmechanismen zwischen molekularen Systemen mit der Zelloberfläche verstanden werden [Schubert et al., 2006; Friedenberger et al., 2007].

Die resultierenden Daten sind derart komplex, dass eine direkte Abbildung aller Informationen für Menschen durch eine intelligente Visualisierungen und interaktive Selektionsverfahren erfolgen muss, zu denen sich auch neue Herangehensweisen gesellen, die mit bisherigen Darstellungskonventionen brechen, um einen alternativen Einblick auf die Strukturen zu bieten [Oeltze et al., 2011].

1.1 Motivation

Die Analyse der generierten Daten bedarf neben einem geschulten Spezialisten (im Folgenden als Biologe bezeichnet) auf dem Gebiet der Gewebe- und Zellforschung einer geeigneten Visualisierung. Durch die große Anzahl unterschiedlicher Proteinnetzwerke ist es ebenso wichtig, eine gute Skalierungsmöglichkeit der angezeigten Strukturen bereit zu stellen. Die Berücksichtigung der mentalen Auslastung des Nutzers bei der Verarbeitung der Daten und dem Ziehen von biologischen Schlüssen ist ein Kernaspekt bei der Entwicklung einer geeigneten Darstellungsweise. All diese Faktoren führen mit der Messung von Gewebeproben in mehreren Schichten eine neue Dimension, mit der auch neue Fragen hinsichtlich der Navigation innerhalb des Datenraums aufgeworfen werden.

Es wurden bereits dreidimensionale Proteinnetzwerke dargestellt [Schubert et al., 2006]. Diese Visualisierung zeichnete sich jedoch durch eine sehr komplexe, nicht alltagstaugliche Aufbereitung der Daten aus und erfolgte unter Verzicht aller bis dahin geschaffenen Werkzeuge für die Analyse dieser Muster [Friedenberger et al., 2007]. Dies ist der Ausgangspunkt der vorliegenden Arbeit.

Neben einer geeigneten Darstellung von dreidimensionalen Proteinstrukturen ist die Interaktion mit den dargestellten Objekten ein weiterer wichtiger Aspekt bei der Analyse. Hierbei treten nicht nur zunächst trivial erscheinenden Aufgaben wie die Bewegung der Kamera auf einen bestimmten Ausschnitt des Gewebes in den Fokus, sondern auch komplexere Eingaben wie Selektion mehrerer Muster und deren kontextuelle Analyse.

Darüber hinaus bieten neue Visualisierungstechniken die Möglichkeit, neue Sichtweisen auf die komplexe Struktur zu entwickeln. Wenngleich bei der interaktiven Darstellung darauf geachtet werden muss, dass der Nutzer nicht mit zu vielen gleichzeitig dargestellten Informationen überlastet wird, gilt gleichermaßen, dass unterschiedliche Methoden und Werkzeuge zur Verfügung gestellt werden müssen, um Einsicht in die Funktionsweise und Organisation von Zellen und Strukturen zu bekommen.

1.2 Zielsetzung

Aus der Motivation ergeben sich unterschiedliche Ziele, die in dieser Arbeit erreicht werden sollen - im Wesentlichen lassen sich diese auf zwei Teilaspekte aufteilen:

- Erstellung einer geeigneten Visualisierung von 3D-Proteinnetzwerkdaten mit Fokus auf eine interaktive Skalierbarkeit der dargestellten Muster durch den Nutzer
- Alternative Darstellung der Daten durch ein neues Graph-basiertes Verfahren ([Oeltze et al., 2011]) und Implementierung in die für die Toponomanalyse verantwortliche Software “MultiCompare”.

Das Ergebnis dieser Arbeit ist das Implementieren zusätzlicher Werkzeuge, welche die Biologen bei der Analyse der hochkomplexen Daten unterstützen und ihnen durch neue interaktive Komponenten eine selektive Darstellung in kürzester Zeit ermöglichen sollen.

Zur Erfüllung dieser Ziele soll auf eine bereits zum Zweck der Analyse dieser speziellen Datensätze geschriebene Software aufgebaut werden. Neben einer einheitlichen Bedienung, die mit bisherigen Designentscheidungen übereinstimmen, steht, wie in der Motivation angedeutet, die Berücksichtigung der kognitiven Last des Nutzers im Vordergrund, da eine Vielzahl unterschiedlicher Werkzeuge schnell durch ihre Informationsflut die Aufnahmefähigkeit beeinträchtigen können. An dieser Stelle müssen neue effektive Techniken erdacht werden.

1.3 Gliederung

Die Arbeit ist in folgende Kapitel unterteilt:

- **Kapitel 2 - Biomedizinische Grundlagen**

Für das Verständnis des biomedizinischen Aspekts werden zunächst einige grundlegende Funktionsweisen von molekularen Netzwerken beschrieben, bevor mit der Fluoreszenzmikroskopie die Basis für das bildgebende Verfahren gelegt wird. Dieses wird abschließend zusammen mit den erhobenen Daten erklärt.

- **Kapitel 3 - Grundlagen und verwandte Arbeiten**

Auf welchen informatischem Wissen die Arbeit aufbaut, ist Gegenstand dieses Kapitels. Grundlagen und Stand der Forschung zu der direkten Volumenvisualisierung werden kurz zusammen mit der 3D-Interaktionskomponente und der Graphvisualisierung vermittelt. Die vor allem für die 2D- und 3D-Visualisierung nötige Generierung von perzeptuell unterscheidbaren Farben bildet den letzten Abschnitt des Kapitels.

- **Kapitel 4 - Entwurf**

Die Analyse der Forschungsabläufe bei der Analyse von Proteinnetzwerken und die darauf folgende Anforderungsanalyse bilden das Fundament für die 3D- und Graphvisualisierung, die in diesem Kapitel entwickelt werden. Wie die interaktive Komponente für beide Visualisierungen aussieht, wird ebenso diskutiert, wie ein Algorithmus zur Generierung von perzeptuell unterscheidbaren Farben und sein Einsatz im Kontext.

- **Kapitel 5 - Implementation**

Ausgewählte Details bei der Implementierung sowie die modulare Einbindung der Visualisierungen in die bereits vorhandene Software, werden in diesem Kapitel behandelt. Die Beschaffenheit des User-Interface-Konzepts wird abschließend besprochen.

- **Kapitel 6 - Zusammenfassung und Ausblick**

Im letzten Kapitel wird beleuchtet, wie die Ergebnisse zu bewerten sind und welche Auswirkungen die Arbeit auf die mittelbare Forschung haben wird. Es folgen außerdem eine Einschätzung und Vorschläge, wie die Arbeit auf diesem Gebiet in Zukunft aussehen kann.

2 Biomedizinische Grundlagen

Einführend sollen in diesem Kapitel die grundlegenden Funktionen von Proteinen in Geweben veranschaulicht werden. Anschließend wird beschrieben, wie Proteine und ihre Funktionen unter dem Begriff *Toponom* in Zusammenhang mit ihrer Lage und Wechselwirkung mit anderen Proteinen gebracht werden. Es wird motiviert, warum die Funktion eines einzelnen Proteins im Gewebe von der molekularen Funktion getrennt analysiert werden muss. Die Funktion von Proteinen innerhalb der Zelle ergibt sich aus der Anordnung und der Hierarchie mit- und untereinander [Schubert, 2010]. Wenn die charakteristischen Strukturen von Multi-Proteinkomplexen in Toponomdatensätzen für einen gesunden Gewebssausschnitt bekannt sind, können abnorme, pathologische¹ Veränderungen daraus detektiert werden.

Als bildgebendes Verfahren für Proteine wird im Anschluss die Fluoreszenzmikroskopie erklärt. Um die von Proteinen gebildeten komplexen molekularen Netzwerke und deren Architektur darstellen zu können, wurde die 'automated Multi-Epitope Ligand Cartography/toponome imaging system' (*MELC/TIS*, deutsch: Multi-Epitope²-Ligand³-Kartographie/Toponom Abbildungssystem - *MELK/TIS*) entwickelt, die Gegenstand des folgenden Abschnitts ist. Die nähere Beleuchtung der Daten, die dieses Verfahren erzeugt - formal als 'Combinatorial Molecular Phenotypes' (*CMPs*, deutsch: kombinatorischer molekularer Phänotyp⁴) bezeichnet [Schubert et al., 2006] [Friedenberger et al., 2007] - bildet den Abschluss des Kapitels.

2.1 Molekulare Netzwerke

Die molekulare Struktur von Zellen bedingt komplexe Eigenschaften, wie Morphologie sowie stoffwechselbedingte Ressourcen und ist ebenso verantwortlich für krankhafte Veränderungen. Wenn man Daten analysieren und verstehen möchte, was die Morphologie der Eiweißstruktur in einem bestimmten Gewebe bedeutet, muss man die Funktion dieser Struktur verstehen können. Sind diese Strukturen für zelluläre Funktionen zuständig, werden sie als molekulares Netzwerk bezeichnet.

2.1.1 Proteine

Molekülgruppen in Zellen bestehen im wesentlichen aus Proteinen. Die verschiedenen Funktionen, die von Proteinen ausgeübt werden, sind schwer zu benennen, wenn

¹Pathologisch: krankhaft; vom gesunden Zustand abweichend.

²Epitop: Kleiner Bereich (Molekülabschnitt) einer immunreaktion-auslösenden Substanz (*Antigen*), gegen den das Immunsystem Antikörper (Antigen-erkennendes Eiweiß) bildet.

³ligand (vom lateinischen *ligare* (binden)): Stoffeigenschaft; Stoff besitzt reizempfindliche Moleküle die auf bestimmte Moleküle einer Zielzelle reagieren.

⁴Phänotyp: Summe aller Merkmale eines Individuums.

man sich vor Augen führt, dass allein auf der Zelloberfläche über 8.000 unterschiedliche Proteine eine Vielzahl von Funktionen ausüben (wie etwa die Kontaktvermittlung zwischen Außen- und Innenwelt der Zelle bzw. Steuerung der Zellbewegung; Überwachung von Signalkaskaden zur Informationsverarbeitung innerhalb und in der umgebenden Außenwelt der Zelle; ...) [Boyd et al., 1998].

Proteine sind Grundbausteine von Zellen. Sie erleichtern als zentrale katalytische Einheit⁵ chemische Prozesse, die sonst nur in einer lebensfeindlichen Umgebung, wie viel Druck oder hohe Temperaturen, möglich wären. Diese Proteine werden Enzyme genannt. Die Katalyse ist in der Zelle eine Funktion, die nahezu jede chemische Reaktion begleitet. Da es viele verschiedene Prozesse gibt, existieren dementsprechend viele unterschiedliche Proteine - einige Tausend in der menschlichen Zelle. Darüber hinaus üben Proteine Funktionen wie Regulation, Bewegung (z.B. als Teil von Muskelzellen) und Transport aus. Eine ebenso wichtige Funktion übernehmen Proteine als intra- und extrazelluläres Strukturelement.

Proteinmoleküle besitzen eine sensitive physikalische und chemische Balance. Werden sie gestresst, verlieren sie ihre charakteristische Struktur und damit ihre molekularen Funktionen [de Duve, 1989, S.28-29].

Das Erbgut enthält den Bauplan für jedes Protein, das für sich jeweils eine Aneinanderreihung verschiedener Aminosäuren ist. Komplexe Funktionen und Abläufe innerhalb einer Zelle ergeben sich aus Wechselwirkungen zwischen diesen Proteinen. Wie dieses Netzwerk für jede Zelle auszusehen hat, bestimmt sie in Wechselwirkung mit den umgebenden Zellen. Die räumliche Lage und Anordnung kodieren komplexe Zellfunktionen [Schubert, 2003, S.190-191]. Die Gesamtheit aller Proteine innerhalb einer Zelle oder Gewebes zu einem Zeitpunkt ohne Berücksichtigung ihrer Wechselwirkung mit anderen Netzwerken wird als *Proteom* bezeichnet [Wilkins et al., 1996, S.61]. Allgemeiner formuliert ist das Proteom die Momentaufnahme aller Proteine zu einem Zeitpunkt [Schubert, 2003, S.190].

2.1.2 Toponom

Im Unterschied zum Proteom berücksichtigt man bei der Betrachtung des Toponoms (vom griechischem *tópos*, „Ort“ und *nómos*, „Gesetz“) den topologischen Aufbau des Proteinnetzwerkes. Das Toponom ist also die Gesamtheit der Proteine und deren Netzwerkstrukturen, die in einer einzelnen Zelle in ihrer natürlichen Umgebung unter Berücksichtigung von anderen Zellen *in situ*⁶ ausgelesen wird.

Um die Unterschiede zwischen Proteom und Toponom zu verdeutlichen, wurden sie in Abbildung 2.1 für gesundes und krankhaftes Gewebe gegenübergestellt. Unterschiedliche Proteine werden als verschiedene Symbole dargestellt. Eine rein quantitative Analyse macht keinen Unterschied zwischen den Geweben deutlich - erst eine topologische Untersuchung veranschaulicht Unterschiede zum gesunden Zustand.

Bei der Analyse des Toponoms ist wesentlich, pathologische Veränderungen im Pro-

⁵Katalyse (vom griechischem *kata* für hinab und *lysis* für Auflösung) ist die Fähigkeit bestimmter Substanzen (Katalysatoren), chemische Umsetzungen zu erleichtern, ohne dabei selbst verbraucht zu werden [de Duve, 1989, S.28].

⁶In situ: In natürlicher Lage, im Körper.

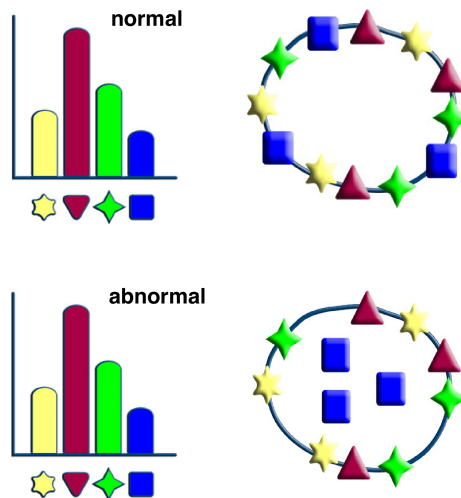


Abbildung 2.1: Vergleich von innerzellulären Proteom-Profilen (Proteom) und Mustern (Toponom). Die einzelnen Symbole repräsentieren unterschiedliche Proteine. Im linken Teil ist die als *Proteom* bezeichnete absolute Häufigkeitsverteilung der Proteine zu sehen, rechts deren topologische Anordnung im Gewebe [Schubert, 2004, S.50, Abb. 2]. Verdeutlicht wird, dass die Häufigkeit von Proteinen in normalen und pathologischem Gewebe gleich sein kann, während die topologische Anordnung Unterschiede aufweist.

teilmuster zu erkennen und zu markieren. Um dies zu ermöglichen müssen die zugrunde liegenden Proteine einzeln sichtbar gemacht werden.

Für Funktionen verantwortliche Proteinkombinationen werden als *Proteinkomplexe* oder *Proteincluster* bezeichnet. Als kleinste semantische Einheit des Toponoms kodieren sie für sich oder in Kooperation mit anderen Komplexen unterschiedliche Zellzustände. Die Komplexe zu identifizieren und darzustellen, die für bestimmte Funktionen verantwortlich sind, ist der Kernaspekt bei der Toponomanalyse. Verändern sich Struktur und Zusammensetzung der Proteine, verändern sich die Funktionen ihres Clusters, während die relative Proteinanzahl in der Regel konstant bleibt [Schubert, 2010].

2.1.3 Leitproteine

Leitproteine sind solche Proteine, die in allen Proteinclustern eines Gewebes zu finden sind. Die bisherige Forschung auf diesem Gebiet deutet darauf hin, dass sie für die hierarchische Steuerung von Abläufen in der Zelle verantwortlich sind [Schubert et al., 2006; Schubert, 2010]. Anders ausgedrückt spricht zur Zeit vieles dafür, dass die korrekte zelluläre Funktionsweise eines Proteinclusters direkt von den Leitproteinen abhängig ist. Fehlen sie, ist der Cluster nicht mehr funktionsfähig.

Die Betrachtung der Leitproteine ist besonders dann interessant, wenn eine Topono-

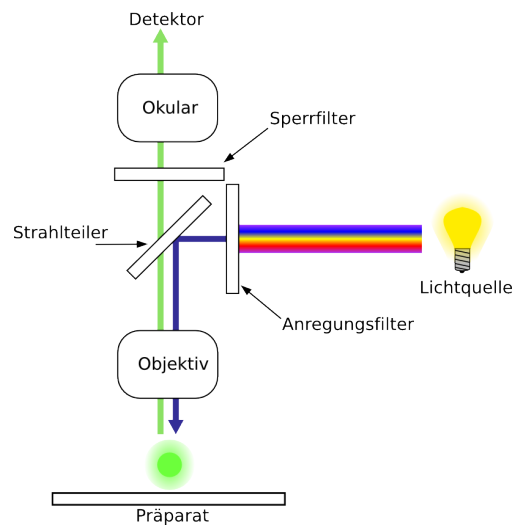


Abbildung 2.2: Schematische Darstellung der Funktionsweise eines Fluoreszenzmikroskops. Anregungslicht wird durch Filter auf das Objektiv gegeben. Fluorochrome emittieren Licht, das durch den Strahlteiler zum Okular gelangt und vom Detektor in ein Bild umgesetzt wird [Flu, 2011].

analyse von gesundem und pathologischem Gewebe desselben Typs unterschiedliche Leitproteine aufweist. Mit diesem Wissen kann das Leitprotein evtl. als Indikator oder Marker dienen, mit denen die pathologische Veränderung diagnostiziert werden kann.

Um neue Medikamente herzustellen, ist die Krankheitsforschung daran interessiert, sogenannte *Target-Moleküle* zu finden, die ein krankheitsspezifisches Muster im Gewebe kennzeichnen. Wenn hier Leitproteine identifiziert werden können, sind Rückschlüsse darauf möglich, ob und wie bestimmte Steuerungsprozesse in der Zelle gestört sind. Aufbauend auf diesem Wissen sind Medikamente denkbar, die gezielt die Leitproteine in einem pathologischen Muster zerstören und somit eine weitere Krankheitsausbreitung aufhalten oder gar die Krankheit heilen können.

2.2 Fluoreszenzmikroskopie

1908 wurde die Fluoreszenzmikroskopie von Köhler entwickelt, später von ihm und Siedentopf vervollkommen.

Sie beruht auf folgendem Prinzip: Bestimmte Stoffe, *Fluorochrome*, senden Licht in einer charakteristischen Wellenlänge aus, wenn diese zuvor mit einer bestimmten Wellenlänge angeregt wurden. Abbildung 2.2 veranschaulicht die Funktionsweise eines Fluoreszenzmikroskops. Licht wird durch einen Anregungsfilter in einer bestimmten Wellenlänge über den Strahlenspiegel auf das im Mikroskop liegende Präparat gegeben. Das *Fluorochrom* setzt diese Energie wieder in Lichtenergie um. Auf physikalischer Ebene hebt sich ein Elektron auf ein höheres Niveau und fällt in Nanosekunden wieder

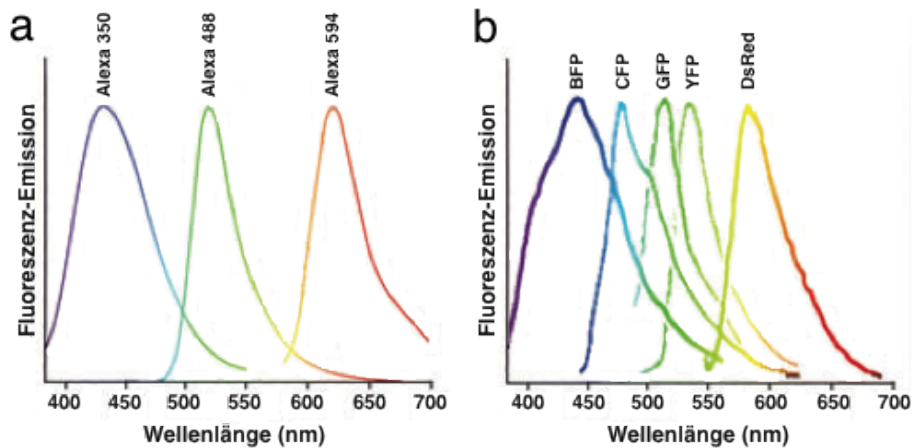


Abbildung 2.3: Emissions-Spektrum ausgewählter Fluoreszenzfarbstoffe. (a) Die dargestellten lichtausstrahlenden Fluoreszenzfarbstoffe sind anhand ihrer spezifischen Wellenlängen gut unterscheidbar. (b) Da die Emissionsspektren sich stark überlagern ist eine eindeutige Zuordnung schwierig [Lichtman und Conchello, 2005].

in seinen Ausgangszustand. Die hier freigesetzte Energie wird als Licht abgegeben. Die Wellenlänge des emittierten Lichts ist jedoch länger als die des zugeführten Lichts, da ein Teil der Energie als Wärme freigesetzt wird. Im Farbspektrum ist das ausgestrahlte Licht in Richtung Rot verschoben. Da das emittierte Licht schwächer als das einstrahlende Licht ist, wird es vom Strahlteiler gefiltert und damit die Lichtemission sichtbar [Lichtman und Conchello, 2005].

Manche Substanzen sind *autofluoreszent*, sie sind in der Lage ohne zuverige Fluoreszenzmarkierung selbst Licht bei einer Anregung durch bestimmte Wellenlängen zu emittieren. Andere Stoffe müssen für eine Detektion durch Fluoreszenzfarbstoffe gekennzeichnet werden, allgemein auch *Fluoreszenzmarkierungsverfahren* genannt. Es werden Markierungsmoleküle an *Fluorochrome* gebunden. Auf dem Gewebe bindet sich der Marker an bestimmte Moleküle (z.B. Proteine). Durch die Fluoreszenzmikroskopie kann nun bestimmt werden, wo der Marker gebunden hat. Für diese Arbeit ist wichtig, dass es Marker gibt, die sich spezifisch an bestimmte Proteine binden. Anhand von Markerbibliotheken kann mit dem Wissen um die Bindungseigenschaften bestimmter Stoffe eine Menge von Markern zusammengestellt werden, die eine gewünschte Anzahl von Proteinen erkennen und sich an sie koppeln.

Auf einen einzelnen Gewebsausschnitt können in der Fluoreszenzmikroskopie maximal fünf verschiedene Marker inkubieren⁷. Das funktioniert jedoch nur, wenn die Wellenlängen der Fluoreszenzemissionen gleichverteilt auf einem Spektrum von 400 nm (violett) bis 700 nm (rot) liegen (Abbildung 2.3.(a)). Bereits bei drei Markern überlagern sich jedoch die Emissionsfrequenzen, was eine genaue Zuordnung der auf-

⁷Inkubation: Einwirken lassen von Enzymen, Antikörpern etc. auf ein Substrat.

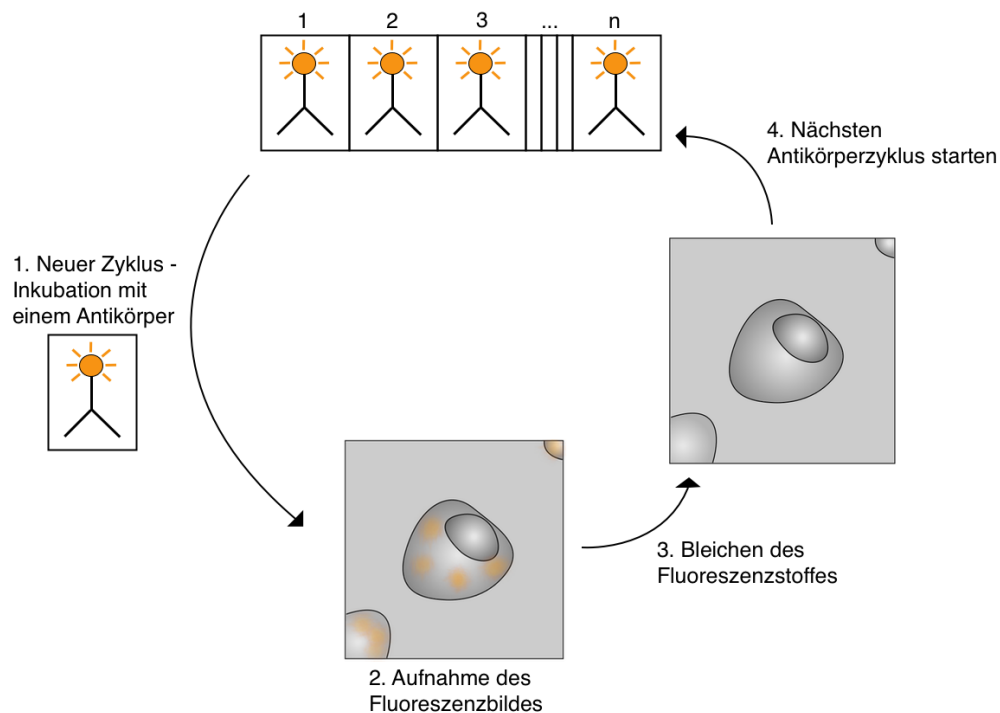


Abbildung 2.4: Schematische Darstellung der Funktionsweise des Multi-Epitope-Ligand-Kartographie Roboters. Zu sehen ist ein Zyklus mit einem Antikörper, der durch einen Fluoreszenzstoff (orange Sonne) markiert ist. Es gibt insgesamt n verschiedene Antikörper, dementsprechend müssen auch n verschiedene Zyklen durchgeführt werden. Ein einziger solcher Zyklus kann 30 - 90 Minuten dauern, abhängig von Parametern wie Inkubationszeit des Antikörpers oder der Bleichzeit des Farbstoffes [Klemm, 2010, S.14].

genommenen Lichtquellen schwierig macht (Abbildung 2.3.(b)) [Lichtman und Conchello, 2005]. Durch spezielle Filter ist es möglich, einen Fluoreszenzfarbstoff in einem 10nm Fenster wahrzunehmen, jedoch zeigen die Ergebnisse nur eine Teilmenge der Marker. Darüber hinaus ist diese Filterung technisch sehr aufwändig.

2.3 Automatisierte Multi-Epitope-Ligand-Kartographie

Diese oben dargestellten Grenzen der Fluoreszenzmikroskopie machen es unmöglich, eine Vielzahl von verschiedenen molekularen Mustern mittels dieser Bildgebung in einem Gewebsausschnitt zu markieren. Mit der auf der Fluoreszenzmikroskopie aufbauenden Technologie der automatisierten Multi-Epitope-Ligand-Kartographie können jedoch mehr als 100 verschiedene Moleküle in einer Gewebeprobe dargestellt werden [Schubert, 1990]. Die bereits in Abschnitt 2.2 genannten Markerbibliotheken werden verwendet, um verschiedene Proteine zu markieren. Um mehr als fünf Fluoreszenz-

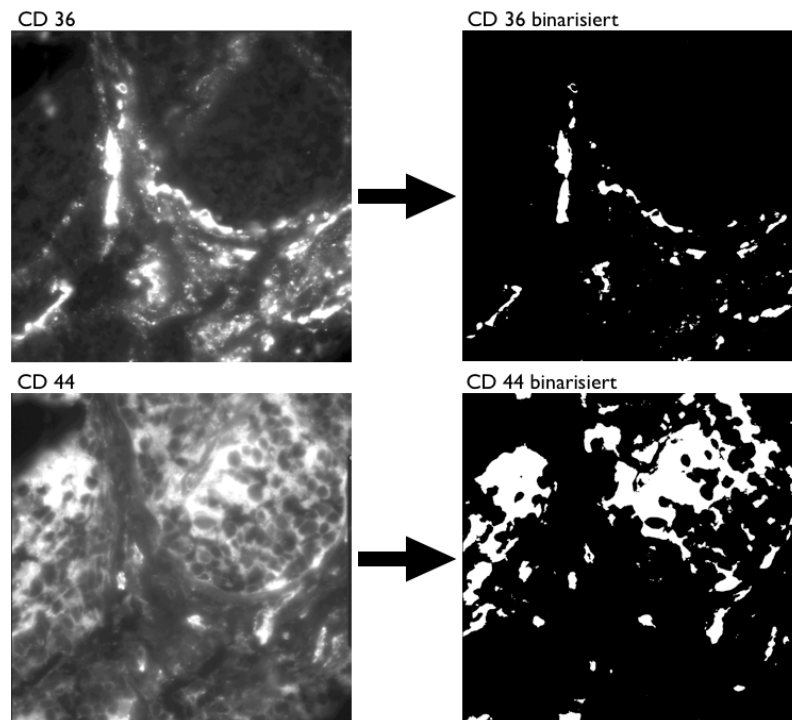


Abbildung 2.5: Vergleich von MELK/TIS-Zyklen mit Antikörpern *CD36* (Zelloberflächenprotein) und *CD44* (Glykoprotein, wichtig für Interaktionen zwischen Zellen) in Neuroblastomgewebe. Das Neuroblastom ist ein bösartiger Tumor, der sich bei Kindern aus Nervenzellen des sympathischen Nervensystems entwickelt. Die von einem Experten binarisierte Version ist rechts neben den Bildern zu sehen [Klemm, 2010, S.15].

farbstoffe messen zu können, wird jeder Marker zyklisch zu dem Gewebe hinzugefügt - Abbildung 2.4 macht einen solchen Zyklus deutlich. Der Marker wird zunächst klassisch zu dem Gewebe hinzugefügt und gemessen. Anschließend wird das Gewebe 20 Minuten lang mit dem Anregungslicht bestrahlt, was den Farbstoff ausbleichen lässt. Dieser kann von nun an kein Licht mehr emittieren. Darauf folgt ein weiterer Zyklus mit dem nächsten Marker, der demselben Ablauf folgt. Es werden solange Zyklen durchgeführt, bis alle Marker auf dem Gewebeausschnitt inkubiert wurden. Ein für diesen Zweck konstruierter Roboter steuert die einzelnen Zyklen und führt diese, ebenso wie die Bildaufnahme durch das Fluoreszenzmikroskop, eigenständig durch.

Für n Marker werden n Zyklen vom MELK/TIS-Roboter durchgeführt. Am Ende des Laufs stehen n Fluoreszenzbilder zur Verfügung. Die Bilder werden anschließend binarisiert, um folgende Berechnungen zu vereinfachen und um den Datenraum zu reduzieren. Die Binarisierung wird mittels eines von einem Biologen durchgeführten

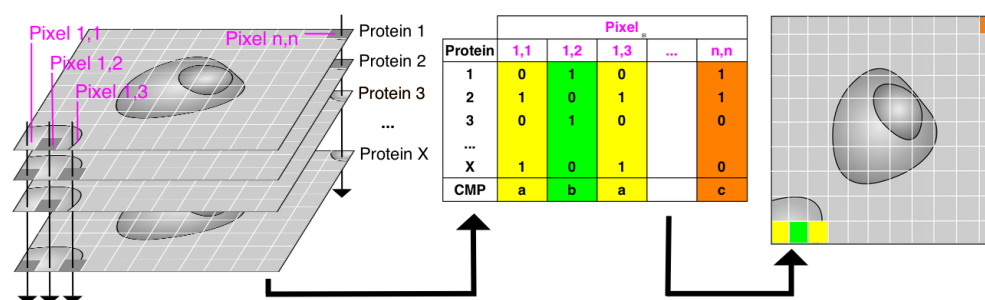


Abbildung 2.6: Beispiel für den Erstellungsvorgang von Combinatorial Molecular Phenotypes. Für die Pixel (1,1), (1,2), (1,3) und (n,n) werden die Vektoren der binären Fluoreszenzsignale bestimmt, von denen vier verschiedene vorliegen (vier Binärbilder). Den Vektoren (als *a*, *b* und *c* bezeichnet) wird eine Farbe zugeordnet (gelb, grün und orange), wobei zwei unterschiedliche CMPs nie dieselbe Farbe aufweisen dürfen. Diese Farbe repräsentiert ein CMP auf dem 2D-Bild. Da die Pixel (1,1) und (1,3) dieselben Proteinvektoren besitzen, gehören sie auch demselben CMP an und teilen sich eine Farbe [Klemm, 2010, S.16].

Schwelldwertverfahren realisiert. Hierbei wird für jedes Pixel⁸ festgelegt, ob es ein Fluoreszenzsignal enthält. Ist der Wert eines Pixel *1* (*anwesend*, *weiß*), enthält es ein Signal. *0* (*abwesend*, *schwarz*) kodiert die Abwesenheit des Signals. Abbildung 2.5 zeigt ein Beispiel für eine Binarisierung.

Da die berechneten Bilder noch nicht zu einem einzelnen Datensatz zusammengefügt wurden, wird ein Format benötigt, das all die hochdimensionalen Informationen zu einzelnen Clustern verknüpft und darstellt.

Die Universitäten Magdeburg, Frankfurt/Main und Warwick (England) wenden diese Technologie bereits erfolgreich an. Analytische Methoden werden an der Universität Bielefeld, Magdeburg und am 'Institute for Computational Biology' in Shanghai im Rahmen des sich im Aufbau befindlichen humanen Toponomprojektes eingesetzt. Die Technologie wird stets weiterentwickelt, sodass in Zukunft noch mit mehr Markerzyklen und höheren Bildauflösungen zu rechnen ist [Schubert, 2003; Schubert et al., 2006].

2.4 Combinatorial Molecular Phenotypes

Das Ergebnis des MELK/TIS-Roboters ist eine Menge von 2D-Bildern, wobei die Anzahl der Bilder von der Anzahl der darzustellenden Proteine abhängt⁹. Dadurch wird bereits für einen einzelnen mikroskopischen Ausschnitt ein Datenraum produziert, der für den Menschen schwer zu analysieren ist. Um alle Markerbilder auf einem einzelnen Bild darstellen zu können, muss der erstellte Datenraum als Vektorfeld aufgefasst werden [Schubert et al., 2006]. Abbildung 2.6 verdeutlicht die Vorgehensweise.

⁸Pixel (Kunstwort, geformt aus den englischen Wörtern *picture* (Bild) und *element* (Element)): Kleinste Einheit einer digitalen Rastergrafik.

⁹Unter Umständen werden verschiedene Marker verwendet, die dasselbe Protein markieren.

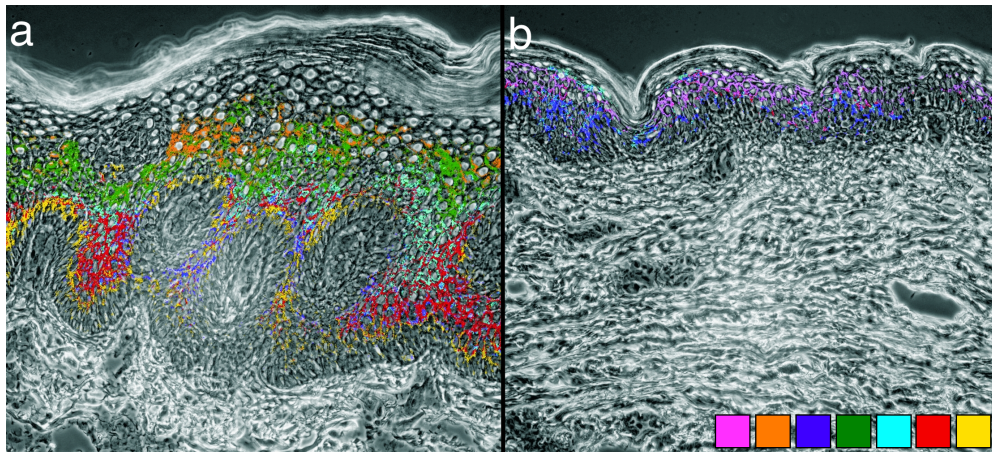


Abbildung 2.7: Darstellung von sieben ausgewählten CMPs in zwei Gewebsschnitten. (a) Probe des Gewebes eines Patienten mit Psoriasis (Schuppenflechte). (b) Ausschnitt von gesundem Hautgewebe der Kontrollgruppe. Das krankhafte Gewebe (a) zeigt eine stark vergrößerte Epidermis sowie eine veränderte Proteinstruktur. Im kranken Gewebe sind CMPs dominant, die durch rot, grün und orange kodiert sind, während in der Kontrollgruppe Strukturen mit den Farben blau und violett vorherrschen [Schubert et al., 2006, S.5, Abb. 4].

Jedes Bild repräsentiert das Vorkommen eines einzelnen Proteins im Gesichtsfeld und liegt binär vor. Es ist also genau entscheidbar, ob Protein X an einer bestimmten Position vorkommt. Der nächste Schritt ist das Erstellen eines Vektors pro Pixel, der für jedes Protein speichert, ob es an angegebener Pixel-Position vorkommt. In Abbildung 2.6 werden solche Vektoren beispielhaft für die Pixel (1,1), (1,2), (1,3) und (n,n) aufgespannt. Die Vektoren werden in eine Tabelle übertragen, in der man erkennen kann, welches Protein an welcher Position vorkommt. Der Vektor für jedes Pixel wird als *Combinatorial Molecular Phenotype* (CMP) bezeichnet. Jedem CMP wird eine Farbe zugeordnet. In Abbildung 2.6 haben die Pixel (1,1) und (1,3) den selben CMP-Vektor, weswegen sie durch dieselbe Farbe repräsentiert werden. Das rechte Bild in Abbildung 2.6 zeigt für die berechneten Pixel das Ergebnisbild.

Abbildung 2.7 zeigt, dass eine Einfärbung der Pixel anhand der CMPs nicht, wie man zunächst annehmen würde, ein Rauschen von Farben ist. Da viele Pixel dasselbe CMP enthalten, bilden sich Muster, die bisher in den komplexen Daten ebenfalls vorhanden sind, jedoch vorher nicht erkennbar waren. Es ist ebenfalls in der Abbildung zu sehen, wie stark sich die molekularen Muster in Kontrollgruppe und pathologischem Gewebe unterscheiden. Es kommen teilweise CMPs im krankhaften Gewebe vor, die in der Kontrollgruppe nicht vorzufinden sind.

Die Zusammenfassung der einzelnen Proteinschichten auf einem einzelnen Bild ist der erste Schritt bei der Visualisierung des Toponomes von Geweben. Für eine ausgewählte Menge von Proteinen können molekulare Muster berechnet und dargestellt werden, wobei funktionelle Abhängigkeiten ableitbar sind. Ein Beispiel hierfür zeigt Abbildung 2.8, in dem fünf CMPs dargestellt sind. In 2.8 (a.1) wurde kein Protein

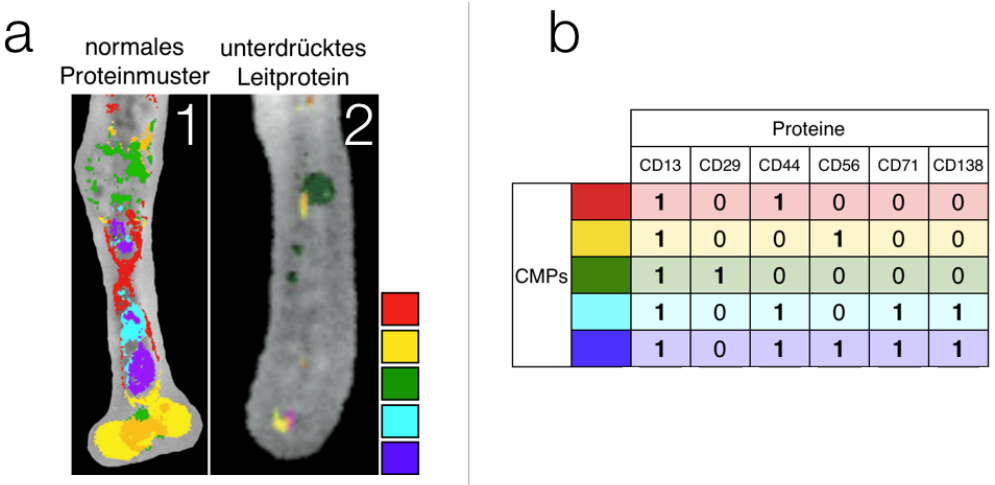


Abbildung 2.8: Darstellung von fünf verschiedenen CMPs in einem Zellausschnitt. (a) Normales, unbeeinflusstes Proteinmuster. In (b) wurde das Leitprotein der Struktur unterdrückt, was die fast vollständige Zerstörung des Proteinmusters zur Folge hat [Schubert et al., 2006]. Die CMP Tabelle zu dem Ausschnitt ist unter (b) zu sehen.

unterdrückt. Die zugehörige CMP-Tabelle ist unter 2.8 (b) zu finden. Dieser kann man entnehmen, dass alle CMPs das Protein *CD13* gemeinsam haben. Wird dieses Protein unterdrückt, zerfällt die Struktur des Netzwerkes, wie in 2.8 (a.2) zu sehen [Schubert et al., 2006]. Im gegebenen Beispiel wird der Zelle die Möglichkeit genommen, das Explorationsstadium voll auszubilden. Sie ist nicht mehr in der Lage, umgebende Zellen abzutasten um ihre eigene Funktion anzupassen.

Mit dem Wissen um das Verhalten solcher molekularen Strukturen wäre der nahe liegende Ansatz, eine maximale mögliche Anzahl von Marker zu testen und zu berechnen, welche Proteine die Netzwerke gemeinsam haben. Es spielen jedoch weitere Faktoren hierbei eine Rolle: Zellen mit unterschiedlichen Funktionen weisen unterschiedliche Proteinmuster auf, direkte Vergleiche von zellulären Zuständen innerhalb unterschiedlicher Zelltypen sind nicht ohne weiteres möglich. So sind für den Vergleich die bereits erwähnten Markerbibliotheken unerlässlich. Mithilfe dieser Wissenssammlung ist es möglich, systematisch eine Vielzahl von Proteinen zu messen und für alle menschlichen Zellen anzuwenden. Hierin liegt die Aufgabe des *humanan Toponomprojektes*. Eine andere Möglichkeit bildet das hypothesenfreie Messen mit Markerbibliotheken gegen nichtcharakterisierte bzw. nicht bekannte Proteine, was ebenfalls zu Erkenntnissen über die räumliche Verschlüsselung von Zellfunktionen und molekularen Netzwerken führen kann. Das Ergebnis der Untersuchung soll ein Funktionsplan von Zellen sein, der aufzeigt, wie die molekularen Strukturen im gesunden und kranken Zustand der Zelle aussehen, bzw. wie die Zelle die einzelnen Funktionen genau umsetzt [Schubert et al., 2006; Schubert, 2003]. Am Beispiel von Prostatakrebs konnten durch diese Me-

thode bereits zwei Proteinkombinationen gefunden werden, die Zellen als pathologisch klassifizieren können [Schubert et al., 2009].

2.5 Zusammenfassung

Ziel dieses Kapitels war es, die biologischen Hintergründe und Grundlagen, auf denen diese Arbeit beruht, aufzuzeigen. Proteine bilden als Grundbausteine die funktionelle Matrix von Zellen und Geweben. Die Verbindung und das Vorkommen der Proteine in ihrer natürlichen Umgebung wird als Toponom bezeichnet. Anhand des Toponomes können krankhafte Veränderungen des Gewebes erkannt und sogar Leitproteine ausgemacht werden - Eiweiße, die eine Schlüsselrolle bei pathologischen Strukturen spielen. Mit der auf der Fluoreszenzmikroskopie aufbauenden Multi-Epitope-Ligand-Kartographie-Technologie (MELK/TIS) ist es erstmals möglich, eine Vielzahl von Proteinen in einem einzelnen Gewebeschnitt zu messen. Die daraus entstehenden Combinatorial Molecular Phenotypes (CMPs) sind individuelle Proteinkombinationen, die selber einen Vektor in dem Datenraum der Protein-Fluoreszenzbilder darstellen. Die Unterdrückung von pathologischen Proteinstrukturen kann zur Auflösung dieser krankhaften Gewebetypen führen.

3 Grundlagen und verwandte Arbeiten

Das Ziel dieses Kapitels ist es, einen Überblick über die in der Arbeit verwendeten Technologien und deren Grundlagen zu vermitteln. Hierbei liegt der Fokus auf dem Verständnis der einzelnen Methoden.

Zunächst wird die direkte Volumenvisualisierung als Methode, eine Menge von 3D-Datenpunkten darzustellen, beleuchtet. Diese in der medizinischen Visualisierung etablierte Technik bildet die Grundlage für die spätere Darstellung von mehrschichtigen Toponomdaten.

Durch die Projektion eines 3D-Volumens auf den Ausgabebildschirm als 2D-Ebene entstehen eine Vielzahl von Problemen bei der Erfassung des Datenraumes durch den Menschen. Diese reichen von der zunächst trivial erscheinenden Aufgaben, wie das Einschätzen der Entfernung und der Distanz von Punkten, über Verdeckungsprobleme bis hin zu verschiedenen Ansatzpunkten, wie sich ein solcher Datensatz effizient mit der Kombination von Tastatur und Maus navigieren lässt.

Da ein Teil der Arbeit die Darstellung der Toponomdaten als Graph-Struktur vorsieht, werden in diesem Kapitel grundlegende Begriffe im Zusammenhang mit Graphen geklärt.

Wie in Kapitel 2 beschrieben, werden CMPs über Farben kodiert. Da in einem einzelnen Datensatz leicht mehrere tausend solcher molekularen Muster gefunden werden können, ist die Frage, wie möglichst viele vom Menschen unterscheidbare Farben gleichzeitig verwendet werden können, essentiell. Die Diskussion der Grundlagen sowie der bisherigen Forschung zu diesem Thema bildet den Abschluss dieses Kapitels.

3.1 Direkte Volumenvisualisierung

Die direkte Volumendarstellung ist bei Toponomdaten der indirekten Visualisierung vorzuziehen, da Algorithmen der indirekten Volumenvisualisierung, wie z.B. Marching Cubes die Segmentierung eines Objektes mit einem bestimmten Isowert¹ erfordern. Da für Toponomdaten zu Beginn einer Untersuchung noch nicht festgelegt werden kann, welche Strukturen mit welchen Isowerten zu einer semantischen Einheit gehören, sind solche Unterteilungen des Datensatzes schwer umsetzbar. Es kann nicht ausgeschlossen werden, dass einzelne Datenpunkte eines bestimmten molekularen Musters signifikant für ein Krankheitsbild sein können, weswegen die Vernachlässigung solcher Informationen im Rahmen einer Segmentierung nicht akzeptabel ist. Es ist ebenfalls schwer eine algorithmische Wichtung der Signifikanz bestimmter Muster vorzunehmen, da diese meist erst durch die Exploration von einem erfahrenen Biologen bestimmt wird. Ein

¹Isolinie: Linie, auf dem an jedem Punkt der gleiche Wert auftritt. Dieser wird als *Isowert* bezeichnet.

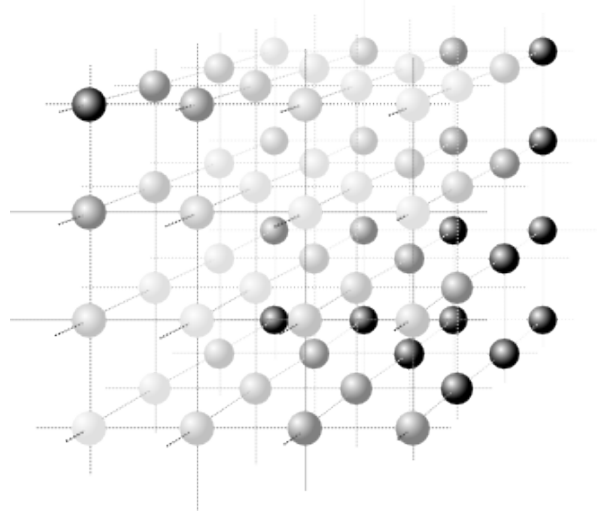


Abbildung 3.1: Veranschaulichung eines Gitters von Datenpunkten. Jeder Datenpunkt (ein sog. *Voxel*) wird durch eine bestimmte Farbe und Opazität² repräsentiert - in der Abbildung abstrakt als unterschiedlich graue *Voxel* dargestellt [Vox, 2011].

derart komplexes Wissen zu algorithmisieren um ähnliche Schlüsse maschinell ziehen zu können, ist aufgrund des Aufwandes nicht zielführend. Aus diesem Grund wird in dieser Arbeit die direkte Volumenvisualisierung verwendet.

3.1.1 Daten

Als Volumendatensatz wird im Allgemeinen eine Menge von 2D-Schichtdaten verstanden, die entlang einer Achse in einem bestimmten Abstand zueinander angeordnet sind. Üblicherweise werden die Schichten entlang der z-Achse übereinander "gestapelt". Die Gitterstruktur von Datenpunkten, die hierbei entsteht, ist im Normalfall regulär, d.h. die Punkte haben jeweils immer in der jeweiligen Dimension einen gleichbleibenden Abstand zu ihren Nachbarn. Es wird davon ausgegangen, skalare Beobachtungspunkte entlang eines Rasters zu haben (Abbildung 3.1). Die Beobachtungspunkte werden als *Voxel*³ aufgefasst [Schumann und Müller, 1999, S.252]. Beim Rendern⁴ eines Volumendatensatz geht es nun darum, abhängig vom aktuellen Blickpunkt auf das Volumen, jedem Pixel einen Rot-Grün-Blau-Alpha⁵ Wert zuzuordnen, um so ein 2D-Bild der Szene zu erzeugen.

²Opazität: Maß für die Lichtundurchlässigkeit. Mit steigender Opazität tritt weniger Licht durch einen Körper hindurch.

³Voxel: Kunstwort, das sich aus den Wörtern **V**olumen und **P**ixel zusammensetzt.

⁴Rendern: Konvertierung einer Vektorgrafik in eine Pixeldarstellung.

⁵Rot-Grün-Blau-Alpha (kurz: RGBA): Farbraum, der Farben durch additiver Mischung der Grundfarben Rot, Grün und Blau erstellt. Der Alpha-Kanal legt die Transparenz der Farbe fest.

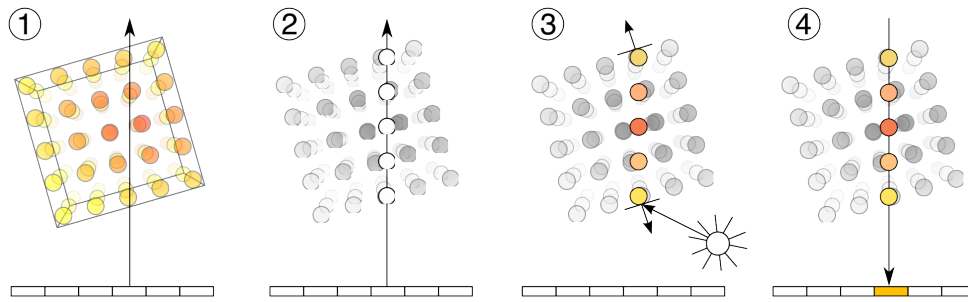


Abbildung 3.2: Arbeitsweise des “Volume ray casting“-Algorithmus. (1) Für jedes Pixel wird ein Strahl in die Szene geschickt. (2) Abtastpunkte werden ausgewählt. Diese befinden sich meist zwischen den Voxeln. (3) Einfärbung und Beleuchtung der Abtastpunkte anhand der Gradienten. (4) Zusammensetzung der Datenpunkte entlang des Strahls (*back-to-front*) [Preim und Bartz, 2007, S.197-199] [Ray, 2011].

3.1.2 “Volume ray casting“-Algorithmus

Betrachtet man ein $64 \times 64 \times 64$ Einheiten großes Datengitter, entstehen über eine viertel Million Speicherplätze. Aus einem Datenumfang von 4 Byte pro Gitterpunkt resultiert eine Datenmenge von 1 Megabyte für die gesamte Struktur. Unter der Berücksichtigung, dass allein eine einzelne Schicht eines hoch aufgelösten CT-Datensatzes, mehr als 40 Megabyte Speicher beanspruchen kann, ist leicht vorstellbar, wie groß die Datenmengen und damit einhergehend Zugriffszeiten werden können [Schumann und Müller, 1999, S.254]. Eine weit verbreitete Methode, die ausgehend vom aktuellen Blickpunkt die Volumendaten in ein 2D-Bild umwandelt, ist der “Volume ray casting“-Algorithmus, dessen Funktionsweise in Abbildung 3.2 veranschaulicht wird.

Zunächst wird für jedes Pixel des zu berechnenden 2D-Bildes ein Strahl in Richtung des Volumens geschickt. Da in den seltensten Fällen das Volumen anhand der Kameraposition ausgerichtet ist, treffen die Strahlen meist nicht direkt die Voxelzentren. Es ergeben sich Abtastpunkte, deren skalarer Wert durch Interpolation (üblicherweise trilinear) der umliegenden Voxel berechnet wird. Anschließend wird das Shading⁶ für jeden Datenpunkt berechnet - also ihm abhängig vom Normalenvektor und der Lichtquellen Farbe und Helligkeit zugewiesen. Welcher Farbwert dem skalaren Abtastpunkt zugewiesen wird, berechnet eine sog. Transferfunktion oder in diskretisierter Form eine Lookup-Tabelle. Abschließend werden für jeden Strahl beginnend vom letzten Datenpunkt bis hin zum ersten die Shading-Werte der Abtastpunkte zusammengesetzt, um die Farbe für das zugehörige Pixel zu berechnen (sog. *back-to-front-compositing*, damit eigentlich verdeckte Punkte nicht das Ergebnispixel beeinflussen) [Preim und Bartz, 2007, S.197-199].

Es gibt mehrere Möglichkeiten diesen Algorithmus zu beschleunigen. Die Schrittweite, mit der Beobachtungspunkte entlang des Strahls eingefügt werden, kann abhängig

⁶Shading (Schattierung): Simulation der Oberflächeneigenschaften von Objekten.

von dem Informationsgehalt der bisherigen Punkte dynamisch angepasst werden - bei transparenten Bereichen wird die Schrittweite erhöht, analog dazu bei opaken Bereichen verringert. Eine weitere Option besteht darin, den Strahl nach Erreichen einer bestimmten Opazität zu stoppen (*early ray termination*).

Eine häufig verwendete Optimierungsmethode ist die Verwendung des Grafikprozessors (GPU⁷) für die Berechnung der Strahlen. Hierbei wird der Datensatz so aufbereitet, dass Berechnungen in kleine unabhängige Cluster unterteilt und diese anschließend von GPU-Threads bearbeitet werden, was aufgrund der hochparallelisierten Architektur von Grafikprozessoren einen großen Geschwindigkeitszuwachs bedeutet [Krüger und Westermann, 2003] [Preim und Bartz, 2007, S.202]. Weitere Algorithmen für die direkte Volumenvisualisierung unter Nutzung der GPU sind in [Rezk-Salama, 2004] zu finden.

3.2 3D-Interaktion

Mit dem Gewinn an Informationen durch die dritte Dimension erreicht auch das Problem der Navigation dieser Daten eine neue Komplexität. Allgemein kann die 3D-Interaktion in vier Teilaspekte untergliedert werden:

- Navigation durch die Szene,
- Selektion von Objekten in der Szene,
- Manipulation von Objekten in der Szene und
- Kontrolle von Systemobjekten - Aktionen, die direkt die Anwendung steuern

Bevor die einzelnen Aspekte näher beleuchtet werden, muss jedoch kurz aufgezeigt werden, welche Probleme der Sprung von der zweiten in die dritte Dimension mit sich bringt.

3.2.1 Wahrnehmungsprobleme in 3D

Bei der Projektion einer 3D-Szene auf ein 2D Ausgabegerät findet stets ein Informationsverlust statt - die Distanz zwischen den Punkten oder die genaue Positionierung von Objekten ist nicht mehr eindeutig erkennbar. Opake Elemente verdecken die Sicht auf dahinter liegende Strukturen. Eine Abhilfe kann durch die durchsichtige Darstellung einzelner Strukturen geschaffen werden. Das führt wiederum zu einer Verfälschung der Farbe der im Hintergrund liegenden Objekte, da sich deren Farbe mit den Elementen mischt, die sich auf deren Sichtlinie befinden. Da für diese Arbeit eine unverfälschte Farbdarstellung essentiell ist, werden in Abschnitt 4.4.3 Techniken entwickelt, die dieses Problem umgehen.

Bei der Navigation des Volumens kann es dazu leicht vorkommen, dass der Nutzer nicht mehr einschätzen kann, wo er sich gerade in der Szene befindet. Eine einfache

⁷Graphics Processing Unit (GPU) Verantwortlich für die Berechnung der Bildschirmausgabe auf Computern.

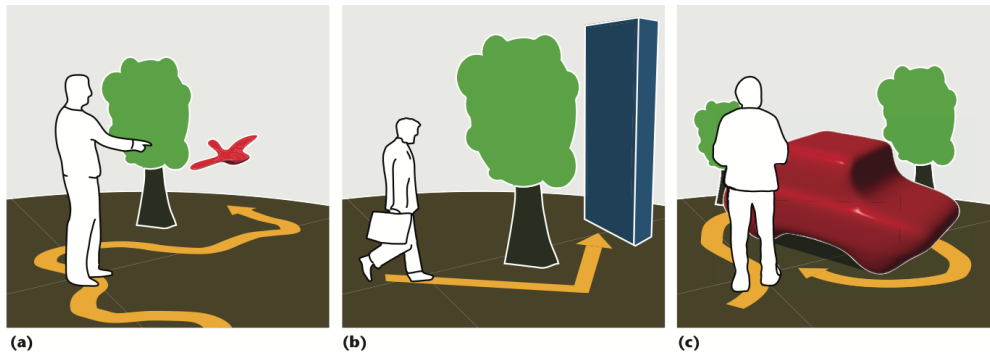


Abbildung 3.3: Die drei Navigationstypen. (a) Explorieren, (b) Suchen, (c) Manövrieren [Kulik, 2009, S.24, Abb. 2].

Möglichkeit, die aktuelle Ausrichtung der Kamera deutlich zu machen, ist es, die Koordinatenachsen anzuzeigen. Dieses unterstützende System verhindert jedoch nicht, dass der Nutzer die Orientierung verlieren kann.

Es existieren noch eine Reihe weiterer Probleme im Zusammenhang mit 3D-Darstellungen, die allerdings nicht essentiell für das Verstehen der Arbeit sind und nicht intensiver diskutiert werden. Weitere Ansätze diesen Problemen zu begegnen, sind in [Houde, 1992] und [Hand, 1997] zu finden.

3.2.2 Navigation und Selektion von Objekten

Im Rahmen der Arbeit besonders interessant sind Techniken zum Navigieren einer Szene zu Explorationszwecken und zum Selektieren einzelner Objekte. Um diesen Problemen zu begegnen, gibt es verschiedene Techniken. Die wesentlichen davon werden in [Hand, 1997] überblicksweise zusammengefasst. Das Prinzip der Erkundung unterteilt man üblicherweise in:

- Wegfindung (wie komme ich von Punkt *A* zu Punkt *B*) und
- Navigation (der Bewegung von *A* nach *B*)

Gleichzeitig wird die Navigation nach [Kulik, 2009] (Abbildung 3.3) unterteilt in:

- Explorieren (Erkundung ohne Wissen um klaren Zielort),
- Suchen und
- Manövrieren (Umgehen eines Objektes)

Es ist schwierig diese unterschiedlichen Aspekte in einer einzelnen Technik zu gleichen Teilen zu berücksichtigen. Häufig wird die Flugmetapher verwendet, der Nutzer kann sich frei im Raum bewegen. Da diese Art der Fortbewegung jedoch für die menschliche

Wahrnehmung ungewohnt ist, kommt es schnell zu den bereits erwähnten Orientierungsproblemen. Da für die Toponomanalyse das Erkunden des Datenraumes im Vordergrund steht, ist das Ziel, ein möglichst gutes räumliches Verständnis zu schaffen. Hierfür ist es erforderlich, dass der Nutzer die aktuelle Kameraposition möglichst frei in festgelegten Intervallen bestimmen kann [Kulik, 2009].

Bei der Selektion ist dem Nutzer bereits bewusst, welches Objekt er auswählen möchte. Ein zu selektierendes Objekt muss in Reichweite, nicht zu klein und nicht verdeckt sein, andernfalls muss zunächst eine Navigation erfolgen. Ist das Ziel des Nutzers dem System bekannt, kann die Navigation automatisch erfolgen. Die Selektion in der direkten Manipulation wird durch *picking* erreicht, d.h. durch das Zeigen eines Cursors wird das Objekt ausgewählt. Ähnlich dem Raycasting-Algorithmus wird das erste Objekt selektiert, das ein Strahl trifft, der an der Zeigeposition losgeschickt wurde. Schwierigkeiten bereiten Szenen mit vielen Objekten, transparente Elemente, besonders kleine Objekte und große Entfernungen. Um Fehlselektionen zu vermeiden und den Nutzer in seiner Aufgabe zu unterstützen, sind *lokale*, *regionale* und *globale* Hervorhebungstechniken sinnvoll [Preim und Ritter, 2002].

- *lokale* Hervorhebung: Lediglich das Objekt, auf dem der Fokus liegen soll, wird beeinflusst. Die Struktur kann z.B. durch Farbe, Rahmungen, Pfeile, Bounding-Box oder Fadenkreuze hervorgehoben werden.
- *regionale* Hervorhebung: Strukturen, die das Fokusobjekt verdecken, werden beeinflusst. Beispielsweise wird bei einem sog. *Cutaway-View* ein zylinderförmiger Bereich aus den verdeckenden Elementen ausgeschnitten, um die Sicht freizugeben.
- *globale* Hervorhebung: Die Kameraposition passt sich so an, dass das gewünschte Objekt optimal sichtbar ist.

Nach [Mine et al., 1997] ist die Selektion die Basis für die Interaktion mit Objekten, sie bildet damit die Voraussetzung für die Manipulation von Strukturen.

Auf die Manipulation von Objekten und der Kontrolle von Systemobjekten wird in dieser Arbeit nicht weiter eingegangen. Für weitere Informationen zur Manipulation, die meist über sog. *Widgets* gelöst wird, wird auf [Conner et al., 1992] verwiesen. Für weiteres Studium der 3D-Interaktion sind [Houde, 1992; Hand, 1997; Kulik, 2009] lesenswert.

3.2.3 Eingabegeräte für 3D-Interaktion

Es wurden zahlreiche verschiedene Eingabegeräte entwickelt, um den Anforderungen für 3D-Interaktion besser gerecht zu werden. Diese sind meist kostspielig und wenig

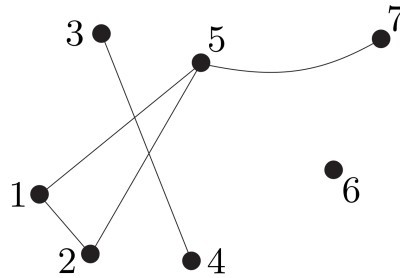


Abbildung 3.4: Beispiel für einen Graph. $V = \{1, \dots, 7\}$ mit Kantenmenge $E = \{\{1, 2\}, \{1, 5\}, \{2, 5\}, \{3, 4\}, \{3, 5\}, \{4, 5\}, \{5, 7\}\}$ [Diestel, 2010, S.2, Abb. 0.1.1].

verbreitet, bieten allerdings mehr Freiheitsgrade⁸, als die klassische Computermaus. Darüber hinaus wird intensiv an der Navigation mittels Multi-Touch-Interfaces geforscht [Martinet et al., 2010].

Nach wie vor ist die Computermaus, meist in Verbindung mit der Tastatur, eine beliebte Methode für die Fortbewegung im 3D-Raum etwa bei Modellierungsprogrammen (z.B. 3ds Max) oder Computerspielen. Die Maus wird entweder verwendet, um den Blickwinkel der Kamera zu ändern oder dargestellte Objekte zu rotieren. In einer Evaluation von [Bade et al., 2005] über verschiedene Rotationstechniken hat sich besonders die *Two-Axis Valuator Trackball*-Technik hervorgehoben, bei der die horizontale und vertikale Mausbewegung das Objekt um bzw. normal zu dem Up-Vektor der Kamera rotieren lassen. In vielen Anwendungen ist der Rotationsmodus auf eine Maustaste gelegt, während die andere Maustaste die Kameraposition auf einer 2D-Ebene verändert. Über ein Mousrad kann ebenfalls die Kameraposition manipuliert werden, z.B. durch eine Fortbewegung entlang des Blickrichtungs-Vektors.

Wie oben bereits erwähnt, existieren eine Reihe Eingabegeräte mit drei und mehr Freiheitsgraden, wie Flysticks (ähnlich der Nintendo-Wii-Remote), Trackballs und Touchscreens, deren Nutzung aber nicht Thema dieser Arbeit ist. Hier sei weiterführend auf [Zhai, 1998] und [Martinet et al., 2010] verwiesen.

3.3 Graphvisualisierung von Toponomdaten

An der Universität Magdeburg wird seit 2010 erforscht, wie Toponomdaten mithilfe von Graphen visualisiert werden können. Entstanden ist eine Methode, mit der einfach erkennbar ist, welche Proteine in einem bestimmten Gewebssausschnitt miteinander vorkommen bzw. welche nicht - eine Information, die bisher nur über Umwege zu erlangen war [Oeltze et al., 2011]. Im Rahmen dieser Arbeit wurde eine Visualisierung entwickelt, die auf diesen Ergebnissen aufbaut aber andere interaktive und statistische

⁸Freiheitsgrade (engl. *Degrees of freedom (DOF)*): Unabhängige Verschiebungen und Rotationen, die komplett die verschobene Position oder verformte Orientierung eines Körpers oder Systems beschreiben. Bsp.: Die Computermaus hat zwei Freiheitsgrade, in Form ihrer Position durch x- und y-Koordinate bestimmt.

Komponenten aufweist. Sie wurde in die im Laboralltag verwendete Software integriert. Um die grundlegenden graphentheoretischen Aspekte und Termini zu klären, befasst sich der folgende Abschnitt mit einer knappen Einführung in das Thema. Anschließend wird der aktuelle Stand der Forschung zu diesem Gebiet dargelegt.

3.3.1 Graphentheoretische Grundlagen

Ein Graph G ist klassisch definiert durch $G = (V, E)$. V ist eine Menge von *Knoten* (im Englischen als *Vertex* bezeichnet). Unter E versteht man *Kanten* (engl. *Edge*). Wird ein Graph visualisiert, stellen die Knoten üblicherweise Punkte dar, die durch Kanten, repräsentiert durch Linien, miteinander verbunden sind. Zwei Knoten dürfen nur durch eine einzige Kante verbunden sein. Abbildung 3.4 zeigt einen *ungerichteten* Graph, d.h. Kanten besitzen keine festgelegte Richtung. Gerichtete Graphen sind für diese Arbeit nicht relevant.

3.3.2 Graphlayout

Während die Definition von Graphen eindeutig ist, unterliegt deren Darstellung großen Freiheitsgraden. Neben ästhetischen Unterschieden existieren auch praktische Anforderungen an eine Graphvisualisierung. Die Anzahl der Kanten, die sich überschneiden, sollte minimal sein, um Verwechslungen zu vermeiden. Die Abstände zwischen den Knoten selber sollte auch gering sein, aber groß genug um Kanten ausmachen zu können. Um Kanten besser verfolgen zu können, sollten sie wenig geschwungen sein. Da die Graphvisualisierung bereits seit langem Gegenstand wissenschaftlicher Forschung ist, existieren eine Reihe von Layoutstrategien, von denen hier zwei ausgewählte vorgestellt werden sollen.

Force-directed Layout

Die kräftebasierte Layout von Graphen basiert darauf, dass möglichst alle Kanten dieselbe Länge besitzen und sich möglichst wenig überschneiden. Der komplette Graph wird als physikalisches System aufgefasst, in dem Knoten wie elektrisch geladene Partikel und Kanten als Federn betrachtet werden. Die Kräfte werden solange simuliert, bis das System einen Ruhezustand erreicht hat. Neben der Einfachheit und Intuitivität des Algorithmus erreicht man gute Resultate bezüglich der Kantenlänge und der uniformen Verteilung der Knoten. Die Ergebnisgraphen sind ebenfalls ästhetisch ansprechend. Die Nachteile dieser Methode liegen in der aufwändigen Berechnung des Ruhezustandes [Di Battista et al., 1998, S.29].

Circular Layout

Die Knoten des Graphen werden bei diesem Layout in gleichen Abständen zueinander entlang eines Kreises ausgerichtet. So sind sie schnell auszumachen, während die Kanten innerhalb des Kreises - in besonderen kreisförmigen Layouts auch außerhalb - verlaufen. Knoten können nicht durch andere Knoten verdeckt werden. Da auch zwischen zwei Knoten nur eine einzige Kante verlaufen darf, ist eine Kantenüberdeckung

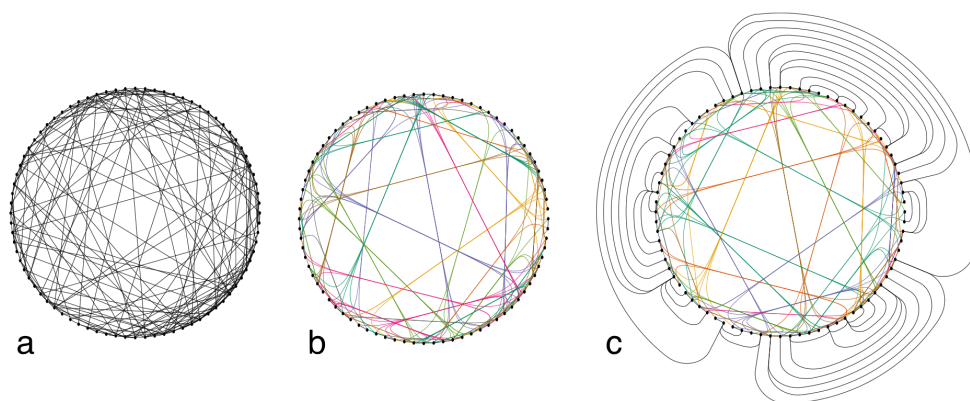


Abbildung 3.5: Vergleich von verschiedenen Circular Layouts. (a) Normales kreisförmiges Graphlayout. (b) Bündelung der Kanten (*edge bundling*) und Verbesserung der Lesbarkeit durch Farbe. (c) Kanten können nun auch außerhalb des Kreises gezeichnet werden, was die Kantendichte reduziert [Gansner und Koren, 2007, Abb. 1].

ausgeschlossen. Für kreisförmige Graphen wird häufig das *edge bundling* angewandt, bei dem Kanten, die in dieselbe Richtung verlaufen, zusammengezogen werden, um eine bessere Übersicht zu schaffen [Gansner und Koren, 2007]. Abbildung 3.5 zeigt verschiedene Strategien beim Zeichnen von kreisförmigen Graphlayouts.

3.3.3 Interaktive Graphvisualisierung von Toponomdaten

Um das Toponom mithilfe von Graphen zu visualisieren, muss zunächst entschieden werden, welche Informationen Knoten und Kanten repräsentieren sollen. [Oeltze et al., 2011] haben den einzelnen Markern, die gemessen wurden, einen Knoten zugeordnet. Zwei Knoten werden durch eine Kante verbunden, wenn in einem ausgewählten CMP beide von den Knoten repräsentierten Markern anwesend sind (Wert 1 haben). Abbildung 3.6 zeigt eine solche Graphstruktur. Sind mehrere CMPs ausgewählt, werden alle Verbindungen im Graph gezeichnet, die in den einzelnen CMPs vorkommen. So kann anhand des Graph sofort erkannt werden, welche Marker in einer Menge von Proteinstrukturen zusammen vorkommen. Der Graph zeigt alle Verbindungen von Markern für eine ausgewählte Menge von molekularen Mustern, losgelöst von ihrer Topologie.

Bezüglich der Anordnung der Knoten haben sich Oeltze et al. für ein kreisförmiges Layout entschieden (Abbildung 3.6 (a)). Ein Vorteil gegenüber anderen Anordnungen ist neben der alphabetischen Anordnung der Knoten nach Markernamen die Vermeidung von sich überlappenden Kanten (Abbildung 3.6 (b)). Dagegen ist es von Nachteil, wenn sich nach jeder erneuten Selektion von CMPs das Graphlayout bzgl. der Anordnung der Kanten ändert. Es müssen jedes Mal erneut die interessanten Knoten vom Nutzer lokalisiert werden.

In dem Graph wurden darüber hinaus weitere Informationen kodiert. In der Kantestärke ist die Häufigkeit der Verbindung abgebildet. Während bei einfachem Vor-

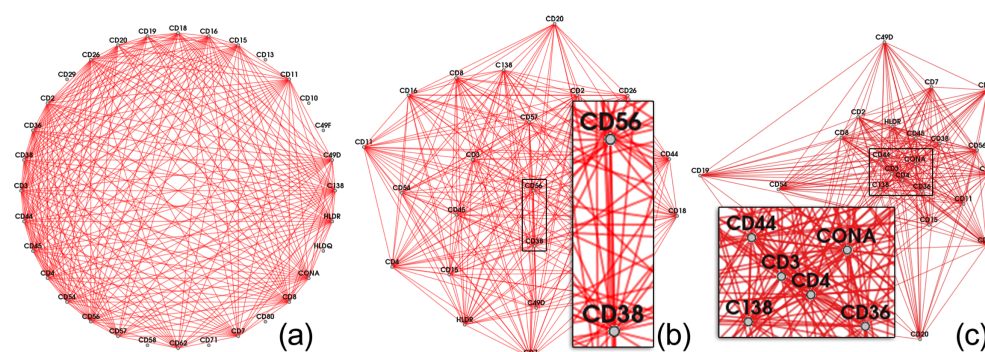


Abbildung 3.6: Vergleich von verschiedenen Graphlayoutstrategien. (a) Circular Layout. (b) Force-directed Layout. (c) Force-directed Layout unter Berücksichtigung der Kantenhäufigkeit. In der Hervorhebung in (b) ist zu sehen, dass sich einige Kanten stark überschneiden, sodass eine eindeutige Zuordnung schwierig wird. In (c) sind Knoten mit vielen Kanten nahe beieinander, was eine Verfolgung schwierig macht. Der Graph bildet ein Extrembeispiel, um das Kantenverdeckungsproblem hervorzuheben, üblicherweise sind in Proteinclustern weniger Verknüpfungen vorzufinden [Oeltze et al., 2011, S.4, Abb. 1].

kommen von zwei Markern in einem CMP die Kante durch eine dünne Linie repräsentiert wird, gewinnt sie bei häufigeren Vorkommen an Stärke. So sind prominente Markerverbindungen leichter auszumachen. Für die Erforschung des Toponomos ist die Häufigkeit eines bestimmten Proteins interessant. So können Fragestellungen, ob das Protein, das mit allen anderen Knoten im Graphen verbunden ist, sehr häufig im Gewebe vorkommt oder nur sehr selten zu beobachten ist, beantwortet werden. Um die Häufigkeit zu visualisieren, wurde neben jedem Knoten ein Glyph⁹ in Form eines Kreisdiagramms gezeichnet, das anhand der Füllung das prozentuale Vorkommen des Markers im Gewebe darstellt.

[Oeltze et al., 2011] verwendeten Graphen, um Unterschiede zwischen T8-Zellen¹⁰ in einem Lymphozyten¹¹-Datensatz festzustellen. In dieser Arbeit wird bzgl. der Graphvisualisierung im Wesentlichen auf diesem Ansatz aufgebaut und ausgehend von den präsentierten Ergebnissen einige Anpassungen vorgenommen.

3.4 Perzeptuell unterscheidbare Farben

Die Wahl der Farben, mit denen die einzelnen molekularen Muster als CMP visualisiert werden (siehe Abschnitt 2.4), hängt von der Anzahl der unterschiedlichen Muster ab, mit denen der Nutzer arbeitet. Dieser Abschnitt stellt verwandte Arbeiten in Bezug auf die Generierung perzeptuell unterscheidbarer Farben vor. In der Literatur ist

⁹Glyph: Visuelle Repräsentation von Daten, wobei die grafischen Attribute durch Parameter der Daten bestimmt werden.

¹⁰T8-Zelle (auch “Suppressor-T-Zelle“): Immunzelle, die Immunreaktion nach der Zerstörung eines eindringenden Organismus zurückfahren.

¹¹Lymphozyten: Als Teil der “weißen Blutkörperchen“ zelluläre Bestandteile des Blutes.

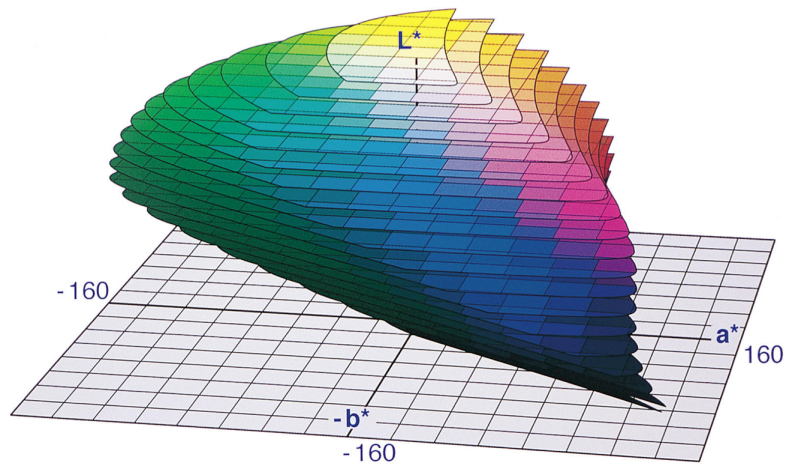


Abbildung 3.7: CIELab Farbraum in 3D-Darstellung. Eine Farbe wird durch die Koordinaten L^* , a^* und b^* definiert, wobei L^* die Helligkeit festlegt, während a^* und b^* die Grün/Rot- bzw. Blau/Gelb-Achse definieren.

erläutert, dass der Mensch sehr gut zwischen fünf und sieben gleichzeitig dargestellten Farben unterscheiden kann [Healey, 1996]. Die Unterscheidbarkeit hängt ebenfalls davon ab, wie stark der Kontrast (Unterschied von Farbe, Sättigung und Helligkeit) zu den umliegenden Farben ist [Ware, 2004].

Verschiedene Farb-Schemata wurden basierend auf den Ergebnissen von [Itten, 1961, S.35] zur menschlichen Wahrnehmung erstellt [Cohen-Or et al., 2006]. [Wang et al., 2008] stellen mit ihrem Color-Selection Framework ein System vor, dass durch eine vom Nutzer gewählte Farbe durch a-priori-Wissen (wie wichtig ein Objekt Element ist, ist bereits bekannt) Farben erstellt, die sich nicht nur gut unterscheiden lassen, sondern auch die Verfälschung durch Überdeckungen minimieren. Dies wird erreicht, indem die Wichtigkeit der Objekte, die sich überlappen, berücksichtigt wird und diesen die Komplementfarben zugeordnet werden, womit die überlappenden Schichten keinen neuen Farbton erzeugen. Ist das nicht möglich, da sich mehrere Objekte gleichzeitig überlappen, wird die Transparenz der Objekte angepasst und die Farben so gewählt, dass der neu entstandene Farbton keinem der bereits vergebenen Farbton ähnelt. Die Sättigung und Helligkeit der Farbe wird in Abhängigkeit von der Wichtigkeit der Objekte bestimmt.

Farbunterschiede können im 1976 festgelegten CIELab-Farbraum [CIE, 2011] berechnet werden. Er enthält alle vom Menschen wahrnehmbaren Farben und ist DIN genormt (DIN 6174). Der euklidische Abstand zwischen zwei Farbpunkten entspricht dem vom Menschen empfundenen Unterschied zwischen den Farben. Wie in Abbildung 3.7 zu sehen, wird der Farbraum von drei Parametern aufgespannt:

- L^* (Luminance): Helligkeitsparameter, von 0 (schwarz) bis 100 (weiß)
- a^* : Die Rot-Grün Achse geht für positive Werte in den roten Bereich, für negative

in das grüne Spektrum. Der Wertebereich umfasst -160 bis 160, wobei nicht für jede Koordinate im 3D-Raum eine Farbe definiert ist.

- **b***: Gelb-Blau Achse - positive Werte gehen in den gelben Bereich, negative in das blaue Spektrum. Der Wertebereich entspricht dem von **a***.

[Sharma et al., 2005] zeigen, wie die einzelnen vom CIE-Konsortium über die Jahre weiterentwickelten Farbunterschiedsberechnungen, die weit über den euklidischen Abstand hinausgehen, implementiert werden.

Ein weiteres System für die Selektion von Farbschemata ist das Tool ColorBrewer [Col, 2011], womit für individuelle Aufgaben angepasste Farben zur Verfügung gestellt werden können [Harrower und Brewer, 2003]. Die Anzahl der Farben wird durch die Anzahl der benötigten Klassen vom Nutzer festgelegt, wobei maximal 12 Klassen unterstützt werden. Abhängig vom Ausgabemedium (wie z.B. Beamer, Bildschirm oder Drucker) schlägt das Tool verschiedene Farbschemata vor. Geordnete Daten, Daten mit einer besonders wichtigen Klasse und ungeordnete Daten werden vom System unterstützt. Letztere sind für diese Arbeit interessant. Die Autoren haben sich hier für Farben entschieden, die eine ähnliche Sättigung und Helligkeit besitzen, jedoch sich in der Wahrnehmung gut unterscheiden. Gleichzeitig existieren zusätzlich Farbschemata, die eine bestimmte Klasse durch eine helle Farbe hervorheben. Leider geht das Paper nicht darauf ein, nach welchem Algorithmus die Farben erstellt wurden.

3.5 Zusammenfassung

In diesem Kapitel wurde beleuchtet, wie Toponomdaten durch direkte Volumenvisualisierung dargestellt werden können und wie der Raycast-Algorithmus diese Aufgabe löst. Für die Visualisierung der Toponomdaten kommt diese Technik zum Einsatz, weil sie die Daten unverfälscht abbildet. Die Möglichkeiten, die zur Verfügung stehen, um einen solchen Datensatz interaktiv zu explorieren, wurden danach im Zusammenhang mit der Interaktionspipeline ebenso diskutiert, wie Wahrnehmungsprobleme, die durch die Projektion eines 3D-Raumes auf ein 2D-Ausgabegerät entstehen können. Oeltze et al. stellen bei der Visualisierung des Toponoms als Graph die einzelnen Proteine als Knoten und deren Verknüpfung in einem ausgewählten Proteincluster als Kanten dar. Diese Informationsabbildung wird im Entwurfskapitel aufgegriffen. Hierbei wurden verschiedenen Graphlayouts beleuchtet, wobei die kreisförmige Anordnung sinnvoll erscheint. Verschiedene Arbeiten im Zusammenhang mit perzeptuell unterscheidbaren Farben bildeten zusammen mit dem CIELab Farbraum den Abschluss des Kapitels.

4 Entwurf der 3D- und Graphvisualisierung

In diesem Kapitel wird das Konzept für Visualisierung von 3D-Toponomdaten durch direktes Volumenrendering und Graphenvisualisierung erstellt. Die Abläufe in der täglichen Forschung werden analysiert, bevor die Anforderungsanalyse in den Fokus rückt. Gegenstand der darauf folgenden Diskussion sind die Methoden mit denen 3D-Toponomdaten visualisiert und explorierbar gemacht werden können. Anschließend wird eine Lösung vorgestellt, wie die Kombination von Epitopen durch einen Graph dargestellt werden kann, sodass diese Visualisierung eine hilfreiche Ergänzung zur 3D-Ansicht ist. Da es eine große Anzahl unterschiedlicher Klassen von molekularen Netzwerken (CMPs) zu handhaben gilt und diese farbkodiert werden, ist es notwendig, einen Algorithmus zu entwickeln, der eine Vielzahl perzeptuell unterscheidbarer Farben generieren kann. Eine Lösung für dieses Problem wird im letzten Abschnitt vorgeschlagen.

4.1 Analyse der Forschungsabläufe

Das große Ziel in der Toponomforschung ist das Extrahieren von Regeln der Funktionen der Zelle und das Treffen von Vorhersagen. Die in der Arbeitsgruppe “Molekulare Mustererkennung“ in Magdeburg angewendeten Arbeitsschritte werden in zwei Abschnitte unterteilt, die nachfolgend beschrieben werden.

Hypothesenfreie Analyse der Zelle

Da kaum Vorhersagen über die Funktion der einzelnen molekularen Bestandteile der Zelle getroffen werden können, wird ein gewählter mikroskopischer Ausschnitt hypothesenfrei analysiert. Das Ziel ist die Analyse der biologischen Ordnung von Zellen und der Ableitung von funktionellen Regeln sowie das Erkennen von Wechselmechanismen zwischen den funktionellen Körpern einer Zelle (der Austausch an den Membranen).

Es existieren etwa 8.000 verschiedene Oberflächenmoleküle, die mit den verschiedenen Markern in einem Datensatz gleichzusetzen sind [Boyd et al., 1998]. Es wird aus dieser Menge eine Untermenge von Molekülen ausgewählt und gemessen, obwohl die Ordnung nicht bekannt ist. Diese werden entweder zufällig oder aufgrund der biologischen Eigenschaften einzelner Moleküle ausgewählt.

Explorieren des Datensatzes und Ableitung von Heuristiken

Nachdem mithilfe der MELK/TIS-Technologie (siehe Abschnitt 2.3) die ausgewählten Marker gemessen wurden, gilt es nun die gewonnenen Daten auszuwerten und Heuristiken sowie Wechselwirkungen zwischen verschiedenen Epitopen abzuleiten. Es werden Interaktionsschemata ausgearbeitet, um zu verstehen, wie Zellen biologische Funktionen verpacken. Hierfür sind verschiedene Fragen zu beantworten:

- Hat jede Zelle/Zellkörper individuelle Strukturen/Bestandteile oder sind Strukturen/Bestandteile ähnlich und nur anders kombiniert? Wahrscheinlich, jedoch nicht wissenschaftlich belegt, ist letzteres.
- Wann kommen Moleküle miteinander vor, wann nicht?
 - Was sind die Regeln hierfür, bzw. welche biologische Komponenten sind für sie verantwortlich?
- Welche Molekülkombinationen unterscheiden sich beim Vergleich von gesundem und krankhaftem Gewebe?
 - Gibt es krankheitsspezifische Marker?
 - Sind die Moleküle in den Geweben gleich, lediglich ihre räumliche Kombination ist anders?

Die Analyse der einzelnen Cluster erfolgte durch An- und Abwahl einzelner molekulare Strukturen (CMPs), um zu überprüfen, an welchen Stellen sie vorkommen und ob aus der Topologie Schlüsse auf die Funktion gezogen werden können. Es ist nicht zielführend stets alle Muster gleichzeitig darzustellen, da ein Datensatz meist mehrere tausend CMPs aufweist.

Um die aufgeführten Fragen beantworten zu können, muss nicht nur ein effizientes Tool zum Analysieren des Datensatzes zur Verfügung stehen, sondern es ist ebenfalls zu berücksichtigen, dass die komplexen Informationen des Datensatzes so aufbereitet werden, dass sie überhaupt noch kognitiv verarbeitet werden können. Im nachfolgenden Abschnitt werden die einzelnen Anforderungen an das Programm hergeleitet.

4.2 Anforderungsanalyse

Die im Abschnitt 4.1 vorgestellten Arbeitsschritte wurden bisher auf 2D-Toponomatensätzen angewandt. Eine Zelle weist jedoch eine 3D-Struktur auf, funktionelle Abläufe sind durch den hinzukommenden Zeitparameter vierdimensional. Dieser Parameter wird jedoch aktuell nicht beachtet. “Snapshots“ von verschiedenen Zeiten eines Ablaufs sind Ausdruck der Zellfunktion.

In 2D-Datensätzen sind es zum großen Teil die Zelloberflächen (Membranen), die analysiert werden. Die 3D-Analyse hat das Ziel, den Austausch zwischen großen molekularen Systemen mit der Zelloberfläche zu verstehen. In 3D sind darüber hinaus Struk-

turen besser erkennbar - die Kombinatorik der Moleküle könnte gleich sein, während sie eine unterschiedliche Anordnung aufweist. Eine fehlende echtzeitfähige¹ Visualisierung dieser Daten verhinderte bislang eine solche Analyse. Die Exploration der molekularen Muster im dreidimensionalen Raum bildet die nächste Stufe in der Toponomforschung.

4.2.1 Aktueller Stand der Arbeiten

Die Software, die in der Arbeitsgruppe “Molekulare Mustererkennung” verwendet wird um Toponomdaten darzustellen, ist als Eigenentwicklung bisher für die Analyse von 2D-Datensätzen im Einsatz. Nachdem die vom MELK/TIS-Roboter aufgenommen Bilder entsprechend aufbereitet wurden (wird in Abschnitt 4.4 vertieft), wird die Software dazu eingesetzt, verschiedene Messungen zu laden und die CMPs darzustellen. Unterschiedliche Funktionen, wie die Möglichkeit, die Menge von CMPs, die angezeigt werden, dynamisch anzupassen, sind für die Exploration vorhanden. Es ist ebenso möglich, verschiedene Filter zu erstellen, die nur molekulare Muster mit einer bestimmten Struktur selektieren sowie verschiedene molekulare Muster zu gleichfarbigen Clustern zusammenzufügen.

Im Rahmen der Arbeit [Klemm, 2010] wurde die Geschwindigkeit der Software signifikant verbessert sowie neue Interaktionsmöglichkeiten hinzugefügt. Es ist seitdem möglich, durch Point & Click CMPs in selektierten Pixeln hervorzuheben. Darüber hinaus können durch eine Lasso-Funktion Muster in einem bestimmten Bereich hervorgehoben werden, was vor allem bei Datensätzen interessant ist, die unterschiedliche Instanzen einer Struktur besitzen und die molekularen Muster dieser verglichen werden sollen (z.B. zwischen Lymphozyten).

Eine Möglichkeit zur Visualisierung von 3D-Datensätzen bot die Software bisher nicht. Die in [Schubert et al., 2006] dargestellten 3D-Visualisierungen erfolgten unter Nutzung des Programms Imaris [Ima, 2011]. Die extrem aufwendige Anpassung der Datensätze in ein vom Programm akzeptiertes Format ist jedoch nicht praktikabel (1-2 Werktagen für 30 CMPs). Durch programmbedingte Beschränkungen können auch nur rund 30 Kanäle (= CMPs) in akzeptabler Form verwendet werden. Darüber hinaus sind auch nicht die von der 2D-Analyse gewohnten Werkzeuge zur Analyse verfügbar. Die Darstellung der Daten erfolgte analog zur 2D-Visualisierung über die Farbkodierung der Proteinnetzwerke. Weitere Details hierzu sind auch im Abschnitt 6.2 dargestellt.

4.2.2 Anforderungen und Zielstellungen

Zusammen mit der Arbeitsgruppe “Molekulare Mustererkennung” wurden genaue Anforderungen ausgearbeitet:

- 3D Visualisierung von Toponomdaten (CMPs) und Implementierung einer interaktiven Point-And-Click Lösung (analog der „Lasso“-Funktion in 2D)

¹Echtzeit: Bezieht sich auf einen vorgegeben Zeitrahmen, innerhalb dessen ein System auf ein Ereignis reagieren muss [Liu und Layland, 1973]. Im Allgemeinen wird unter *echtzeitnah* ein Berechnungszeitrahmen verstanden, der dem sensorischen Kurzzeitgedächtnis von ca. 15 ms entspricht [Klemm, 2010, S.8].

- Schnittstellenprogrammierung zwischen Toponomdaten und Visualisierungswerkzeug
- Übersichtliche und einfache Visualisierung des Protein-Netzwerks als Graph
- Integration von statistischen Merkmalen aus den Toponomdaten in den Graphen
- Suche eines geeigneten OpenSource-Frameworks zur Graphvisualisierung

Im Laufe der Arbeit wurden neue Anforderungen ergänzt. Die für die Toponomvisualisierung verantwortliche Software, die Grundlage für die Erweiterungen der Arbeit ist, wies vorher schon eine Vielzahl von unterschiedlichen Werkzeugen und damit verbundenen Fenstern auf. Mit dem 3D-View und der Graphansicht kommen zwei Programmfenster hinzu, was eine Überladung der Bildschirmdarstellung zur Folge hat. Um zu gewährleisten, dass der Nutzer die Übersicht wahren kann, müssen bzgl. der Anordnung der Nutzeroberfläche neue Strategien gefunden werden. Darüber hinaus ist es wichtig, dass die einzelnen Werkzeuge untereinander kommunizieren können, um auf verschiedene Nutzereingaben in angemessener Weise zu reagieren. Dies gilt vor allem dann, wenn unterschiedliche Werkzeuge sich überschneidende Funktionen aufweisen.

Die farbkodierte Darstellung von CMPs warf im Laufe der Arbeit die Frage nach einem effizienten Algorithmus zur Erstellung von unterscheidbaren Farben. Bislang wurden nur Farben zwischen den Übergängen von Rot zur Grün, Grün zu Blau und Blau zu Rot verwendet.

Hieraus ergeben sich weitere Anforderungen, die zusätzlich zu den oben Genannten gestellt werden:

- Redundante Abbildung von Proteinclustereigenschaften auf visuelle Parameter von verschiedenen Visualisierungen zur Verringerung des kognitiven Workload des Nutzers
- Erstellung einer Lösung für zwei Monitore für die Anordnung der einzelnen Werkzeuge
- Entwicklung eines Algorithmus zum Erstellen perzeptuell möglichst gut unterscheidbarer Farben für bessere Diskriminierung von CMP-Datenpunkten

4.3 Vorhandene Visualisierung

Mittels der “MultiCompare“ Software wurden bisher 2D-Toponomdatensätze visualisiert, indem die einzelnen CMPs farbkodiert dargestellt wurden. Die Menge aller im Datensatz vorkommenden CMPs wird in einer *CMP-Liste* dargestellt, in der jede Spalte ein gemessenes Epitop repräsentiert und ein CMP-Eintrag den Wert *1* (Epitop *anwesend*) bzw. *0* (Epitop *abwesend*) annehmen kann. Diese Liste ist in Abbildung 4.1 (a) dargestellt. Die Pixel, die zu den ausgewählten CMPs gehören, werden in Überlagerung auf einem nebenstehenden Übersichtsbild dargestellt. Als Hintergrundbild werden für weitere Kontextinformationen nichtbinarisierte Fluoreszenzbilder aus

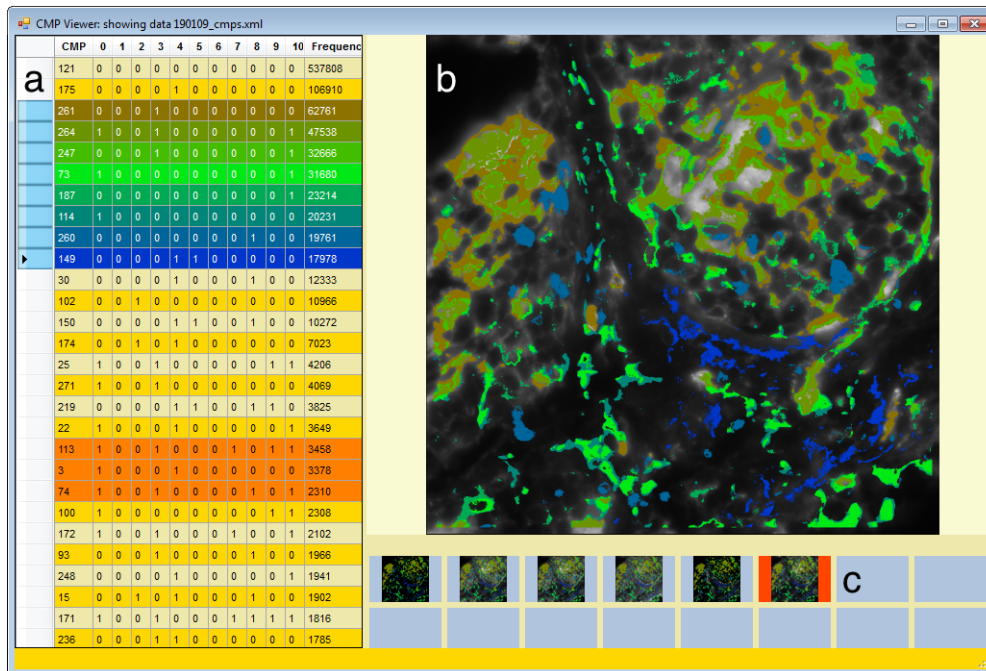


Abbildung 4.1: CMP-Viewer mit Neuroblastomdatensatz auf CD44-(Oberflächenprotein) Fluoreszenzbild. (a) CMP-Tabelle, die alle CMPs des Datensatzes enthält. Spalte 'CMP' speichert die CMP-Nummer. '0' bis 'n' stellt die Zugehörigkeit zum jeweiligen Fluoreszenzmarker dar. Die Spalte 'Frequency' zeigt, wie viele Pixel zum CMP gehören. Durch Auswählen von CMPs in der Tabelle werden sie in (b) dargestellt. (c) legt fest, auf welchem Bild die CMPs gezeichnet werden [Klemm, 2010, S.35, Abb. 4.3].

dem Datensatz verwendet (Abbildung 4.1 (b)). Die Messbilder werden verwendet, um einen topologischen Zusammenhang zwischen den CMPs zu verstehen. Da hier häufig mehrere Fluoreszenzaufnahmen relevant sein können, werden in (c) Platzhalter angezeigt, die jeweils das aktuelle Hintergrundbild, auf dem die CMPs gezeichnet werden, auswechseln.

Zusätzlich zu dieser Visualisierung stehen weitere Tools zur Verfügung, die die Exploration des Datensatzes unterstützen. Durch den *Filter-View* können die einzelnen molekularen Muster nach Epitopen gefiltert werden, indem ein bestimmtes Muster für die einzelnen gemessenen Antigene angegeben wird. So können nur solche CMPs angezeigt werden, die einer bestimmten Vorlage entsprechen. Durch den *Fullscreen-View* kann beliebig nah an CMP-Daten heran gezoomt werden, sodass ein besserer Einblick in Gebiete möglich ist, in denen viele unterschiedliche CMPs vorkommen. Darüber hinaus bietet dieser View zwei weitere Werkzeuge.

- *Picking-Tool:* Durch einen Mausklick auf ein Bildpixel kann das zugehörige CMP

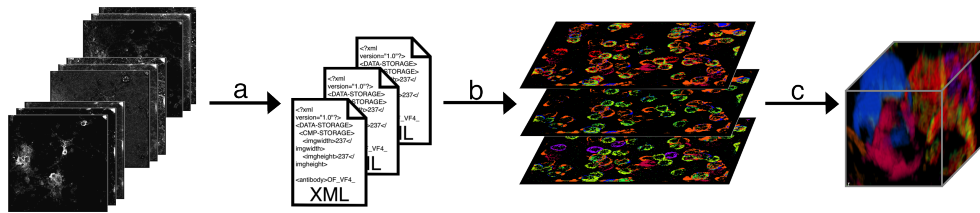


Abbildung 4.2: Pipeline von Fluoreszenzbildern zu einem 3D-Volumen, ausgehend von drei gemessenen Schichten. (a) Gemäß Abschnitt 2.4, Abbildung 2.6 werden die binarisierten Bilder in CMPs umgewandelt. Ihr Aufbau und zugehörige Pixelpositionen werden dann im XML-Format gespeichert. (b) Die einzelnen Schichten werden anschließend in einem vorher definierten Abstand entlang der z-Achse übereinandergelegt. (c) Die einzelnen Farbwerte der Pixel werden als *Voxel* aufgefasst, Zwischenwerte mittels trilineare Interpolation angenähert und dargestellt.

angezeigt bzw. ausgeblendet werden. Diese Technik wird auch dazu verwendet, festzustellen, welches CMP an welcher Position im Gewebe vorliegt.

- *Lassotool*: arbeitet nach dem Prinzip des *Picking-Tools*, jedoch ist es hiermit möglich, durch das Aufspannen eines Rechtecks einen Bereich festzulegen, innerhalb dessen alle CMPs hervorgehoben werden.

Darüber hinaus ist es möglich, bestimmte CMPs zu Gruppen zusammenzufügen und diese mit einer nutzerspezifizierten Farbe darzustellen. So können funktionell zusammenhängende Muster markiert werden, was dem Nutzer eine bessere Übersicht vermittelt.

4.4 Entwicklung der 3D-Visualisierungstechnik

Ziel dieses Abschnittes ist es, eine Visualisierung zu entwickeln, die den im Abschnitt 4.2 aufgestellten Anforderungen an eine 3D-Ansicht von Toponomdaten gerecht wird. Essentiell hierbei ist es nicht nur, die einzelnen Visualisierungsparameter geeignet zu bestimmen, sondern auch sich mit der Explorierbarkeit der Daten auseinanderzusetzen. Hierbei wird sich an der bisherigen Funktionsweise der vorliegenden Software orientiert, um eine möglichst konsistente Interaktion zu gewährleisten. Wie die Daten für eine effiziente Darstellung im 3D-Raum aufbereitet werden können, bildet die Grundlage der Visualisierungstechnik.

4.4.1 Generierung des 3D-Volumens

In Abschnitt 2.4 wurde beschrieben, wie sich aus den vom MELK/TIS-Roboter gemessenen Fluoreszenzaufnahmen die CMP-Daten zusammensetzen. Da per Definition

nur ein CMP pro Pixel existieren kann, sind Probleme wie Verdeckungen und Farbverfälschungen ausgeschlossen [Klemm, 2010, S.31]. Durch die neu hinzugekommene dritte Dimension liegt eine Menge von 2D-Messungen vor, die im bestimmten Abstand zueinander durchgeführt wurden. Der einfache Ansatz ist nun, die Daten entlang der z-Ebene, mit Rücksicht auf den Abstand zwischen den einzelnen gemessenen Schichten übereinander gesetzt, darzustellen (Abbildung 4.2). Hierbei werden die Datenpunkte, die bisher von Pixeln repräsentiert werden, nun als *Voxel* betrachtet.

Die Abstände zwischen den skalaren Voxelwerten werden durch trilineare Interpolation angenähert. Um zu gewährleisten, dass durch die Interpolation keine falschen Informationen produziert werden, ist es essentiell, dass die Abstände zwischen den gemessenen Gewebeproben möglichst gering sind.

Für die Darstellung des Volumens kommt der “Volume ray casting“-Algorithmus zum Einsatz (Abschnitt 3.1.2). Seine Parametrisierung wird nachfolgend diskutiert.

4.4.2 Visualisierungsparameter und -variablen

Schubert et al. haben bereits eine Vielzahl von wissenschaftlichen Arbeiten im Bezug auf die Visualisierung der Toponomdaten im zweidimensionalen Raum veröffentlicht [Schubert, 2003; Schubert et al., 2006, 2009; Schubert, 2010]. Die hier angewandte und bewährte Visualisierung für 2D-Datensätze beruht im wesentlichen auf der direkten Darstellung der Pixel, die ein selektiertes CMP enthalten. Da es selten vorkommt, dass bei einer ausgewählten Menge von CMPs alle Pixel des Bildes einen CMP-Wert besitzen (vgl. Abbildung 2.7), werden die Daten oft auf einem Hintergrundbild dargestellt, das in den meisten Fällen ein originales, nichtbinarisierendes Fluoreszenzbild ist.

Prinzipiell ist der Schritt von einer 2D-Darstellung hin zu 3D einfach. Es ist leicht vorzustellen, aus einer Menge von 2D-Messungen ein 3D-Objekt zusammenzusetzen. Die indirekte Volumenvisualisierung scheidet aus, da es nicht akzeptabel ist, dass womöglich pathologische Muster durch einen Segmentierungsalgorithmus als Artefakte² behandelt werden (vgl. Abschnitt 3.1). Die Schlussfolgerung ist die Verwendung von direkter Volumenvisualisierung. Wie der hierfür verwendete Algorithmus die Messdaten in Bildpunkte umwandelt, wird nun anhand der Transferfunktion³ beschrieben.

Transferfunktion

Der “Volume ray casting“-Algorithmus zeichnet sich durch eine gute Bildqualität gekoppelt an eine lange Laufzeit aus. Üblicherweise benötigt ein solcher Algorithmus eine Transferfunktion, die den einzelnen Voxeln ihre Farbeigenschaft zuweist. Da das 3D-Volumen aus einer Menge von 2D-Schichten besteht, die von CMP-Pixeln generiert werden, ist für die einzelnen skalaren Punkte klar, welche Farbe sie aufweisen. Sie sind selber an die CMPs gekoppelt. Jeder Datenpunkt besteht aus einem RGBA-Wert, von denen jeder einzelne Farbkanal 8-Bit Speicher belegt, er kann 256 verschiedene Werte annehmen. Somit wird keine explizite Transferfunktion benötigt, die Farbe kann

²Artefakt: Sichtbare, unerwünschte Anzeige in digitalen Bildern, die nicht von den Ausgangsdaten herrührt.

³Transferfunktion: Bildet Intensitätswerte eines Volumens auf Farbwerte ab.

direkt an den einzelnen Datenpunkten abgelesen werden. Das getroffene Voxel muss nicht erst klassifiziert werden, sondern kann direkt anhand der ihm eigenen Informationen dargestellt werden.

GPU-Raycasting

Aufgrund der Funktionsweise des Raycasting-Algorithmus eignet er sich für die Ausführung auf dem Grafikprozessor⁴, der durch seine hochparallele Arbeitsweise bei der Abarbeitung von voneinander unabhängigen Berechnungen wesentlich schneller arbeiten kann als eine Implementation auf dem Hauptprozessor. Da die beim Raycasting berechneten Sichtstrahlen voneinander unabhängig ermittelt werden können, sind diese Berechnungen auf die GPU auslagerbar. Eine Anforderung hierfür ist, dass das 3D-Volumen auf den Speicher der Grafikkarte übertragen werden kann, der somit entsprechend groß und für die Datenstruktur geeignet sein muss. Dieser Schritt der Konvertierung der Daten ist häufig die größte Hürde bei der Portierung von Algorithmen auf die GPU. Der in dieser Arbeit verwendete Algorithmus bildet die Datenstruktur im 3D-Texturspeicher der Grafikkarte ab. Für jedes Pixel wird anschließend unabhängig voneinander ein Strahl durch das Volumen berechnet. Neben der parallelen Arbeitsweise sind die geringeren Zugriffszeiten auf das Datenobjekt ausschlaggebend für den Geschwindigkeitsvorteil [Bertel, 2008].

Abstand der Schichten

Wendet man den gegebenen Maßstab auf die virtuelle Repräsentation an, wird die Informationen in der z-Achse sehr komprimiert dargestellt. Es ist die oberste und die unterste Schicht zu erkennen. Wie die molekularen Strukturen dazwischen aussehen, ist jedoch nur bei einer sehr hohen Zoomstufe zu erkennen. Ein Lösungsansatz ist die Streckung der Daten entlang der z-Achse. Hier wird die originalgetreue Darstellung der Daten gegen die Explorierbarkeit aufgewogen. Da an den Daten selber keine Änderung vorgenommen wird, ist dieser Schritt bedenkenlos umzusetzen. Bislang wird als Standardskalierung der Achsen der Maßstab 1:1:2 (x:y:z) verwendet. Welche Werte für eine optimale Darstellung angewandt werden sollen, ist noch Gegenstand von weiteren Untersuchungen. Es steht dem Nutzer über das Interface frei, die Skalierung jeder einzelnen Koordinatenachse selber zu bestimmen.

Damit einzelne Schichten unabhängig voneinander betrachtet werden können, wurde hierfür eine Option in das User-Interface integriert. Wird der *Slicing Mode* aktiviert, ist es möglich, durch jede einzelne Schicht des 3D-Datensatzes durchzuschalten. So wird in 3D auch die Funktionalität des bereits integrierten Fullscreen-View nutzbar gemacht.

⁴Grafikprozessor (GPU - **G**raphics **P**rocessing **U**nit): Berechnet die Bildschirmausgabe eines Computers.

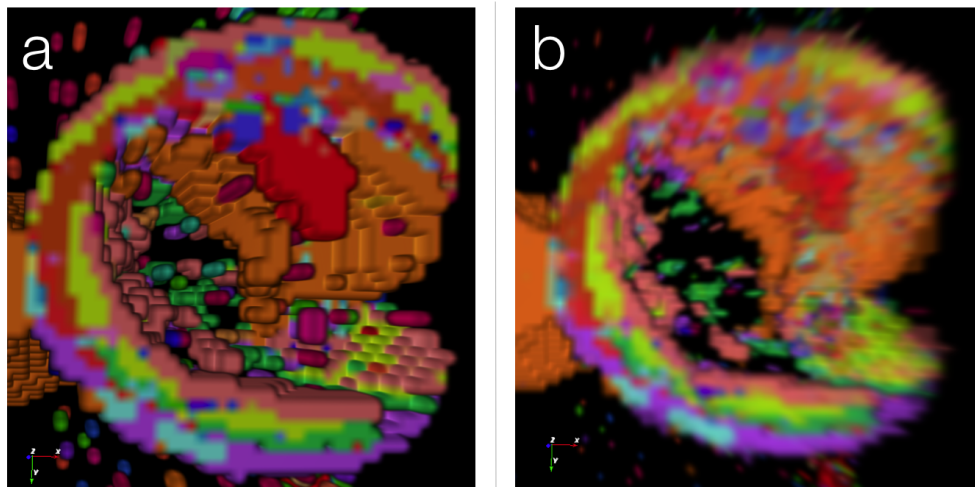


Abbildung 4.3: Farbvergleich bei unterschiedlichen Transparenzstufen. (a) CMPs in einem Lymphozyten dargestellt mit voller Opazität. (b) Derselbe Ausschnitt des Datensatz, jedoch mit erhöhter Transparenz. Während bei CMPs, die durch viele benachbarte Voxel repräsentiert werden, die Farbtöne erhalten bleiben, ist bei kleineren Mustern die Farbe schwerer erkennbar. Single-CMPs sind fast nicht mehr zu erkennen. Die Farben von überlagerten CMPs vermischen sich - ein Effekt, der bei der Echtzeit-Navigation im Datensatz noch offensichtlicher wird, als es das statische Beispielbild zum Ausdruck bringt.

Shading, Beleuchtung und Transparenz

Da keine besonderen Oberflächeneigenschaften simuliert werden müssen, wie es z.B. bei der Volumenvisualisierung von MRT-Daten der Fall ist, wurde auf ein Shading⁵ des Volumens verzichtet. Die Einführung bestimmter Oberflächensimulationen hätte Irritationen durch falsche Reflektionen zur Folge. Da keine bestimmte natürliche Beleuchtung simuliert werden muss, wurden keine weiteren Lichtquellen zu der 3D-Szene hinzugefügt. Das Volumen wird durch eine globale farbneutrale Lichtquelle erhellt.

Die Nutzung von Transparenz ist bei der 3D-Toponomvisualisierung eine große Herausforderung. Während in 2D eine Überlagerung von CMPs per Definition ausgeschlossen ist, kann im schlechtesten Fall an einer festgelegten Pixelposition in jeder Schicht ein anderes CMP, vielleicht sogar ein Single-CMP (CMP, das nur von einem einzelnen Pixel präsentiert wird), aufhalten. Würde man entlang der z-Achse - also von "oben" - auf das Volumen blicken, wäre nur das CMP der obersten Schicht zu sehen. Beim trivialen Ansatz wird nun die globale Transparenz des Volumens angehoben, also die Alpha-Werte jedes Voxels abgesenkt, sodass beim Raytracing nicht nur das erste getroffene Voxel die Information des Pixels definiert, zu dem der Strahl gehört. Wie in Abbildung 4.3 (a) zu erkennen, besteht das dargestellte Volumen meist aus einer Vielzahl ver-

⁵Shading: Simulation von Oberflächeneigenschaften.

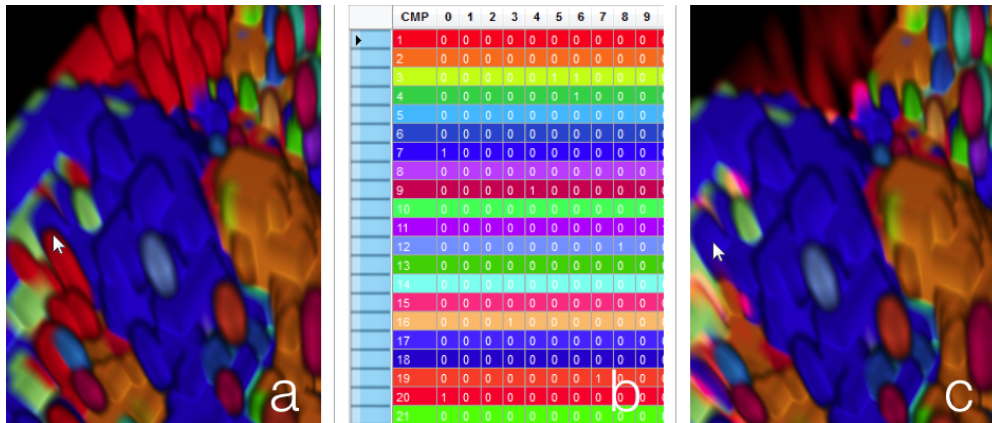


Abbildung 4.4: Beispiel für die Hervorhebung von einzelnen CMPs. (a) Der Mauszeiger markiert ein CMP, das parallel als oberstes Element in der CMP-Liste angezeigt wird (b). (c) Mit der selektiven Transparenz kann das aktuell durch des Pointer hervorgehobene CMP transparent geschaltet werden, um zu sehen, welche Strukturen von diesen verborgen werden. Im Beispiel wird das in (a) selektierte CMP transparent dargestellt.

schiedener CMPs, die jeweils durch unterschiedliche Farben repräsentiert werden. Eine globale Änderung der Transparenz hat eine Farbverfälschung jedes einzelnen Datenpunktes zur Folge, da eine veränderte Opazität die Veränderung des wahrgenommenen Farbwertes zu Folge hat (Abbildung 4.3 (b)).

Darüber hinaus ist es möglich, dass durch die Mischung der Farben von verschiedenen, auf dem Sichtstrahl liegenden Datenpunkten fälschlicherweise neue Farbtöne entstehen. Da jedoch, wie in Abschnitt 4.3 bereits beschrieben, der 3D-View streng genommen nur eine unterstützende Visualisierung unter Verwendung der Listenansicht der CMPs ist, ist eine solche Verfälschung der Farbwahrnehmung ohne weitere Hilfsmittel nicht akzeptabel. Die Farbe bildet die Verknüpfung der Informationen zwischen den beiden Ansichten. Eine in Abschnitt 3.4 diskutierte Lösung von [Wang et al., 2008], sich überdeckende Farben durch Komplementfarben zu ersetzen bzw. die Farben so zu wählen, dass keine bereits vergebenen Farbtöne durch Überlagerung entstehen können, ist aufgrund der hohen Anzahl unterschiedlicher CMPs nicht realisierbar. Da der Nutzer den Blickwinkel auf das Volumen frei wählen kann, sind abhängig von der aktuellen Kameraposition stets andere Voxel überlagert. Warum eine dynamische Wahl von Farben nicht erstrebenswert ist, wird in Abschnitt 4.6 erläutert.

Um dem Problem der Verdeckung zu begegnen, wurde über die Nutzerinteraktion eine Möglichkeit geschaffen, sodass selbst eine globale Transparenz bei der Toponomvisualisierung ein nützliches Werkzeug sein kann. Die selektive Transparenz wird ebenfalls im folgenden Abschnitt beschrieben.

4.4.3 Explorationswerkzeuge und Nutzung des Zeigers

Der Mauszeiger bildet ein nützliches Werkzeug bei der Explorierung des Volumens. Schon in der bereits implementierten 2D-Ansicht wurde er verwendet, um durch Point-

and-Click einzelne CMPs hervorzuheben oder zu deselektieren. Eine ähnliche Funktion findet auch im 3D-View seine Anwendung.

Hervorheben von einzelnen CMPs

Da z.B. bei Single-CMPs nicht immer sofort ersichtlich ist, welche Farbe das zugehörige Voxel hat, kann durch Bewegen des Zeigers auf das Voxel die Farbe im zugehörigen Farbfenster (angezeigt in der linken oberen Ecke des 3D-Views) hervorgehoben und mit dem Eintrag in der CMP-Tabelle abgeglichen werden.

Bei Datensätzen mit mehreren tausend CMPs ist es nicht möglich, die Farben aller CMPs eindeutig voneinander zu unterscheiden. Stellenweise werden dann für unterschiedliche Muster dieselben Farben verwendet, was eine eben beschriebene Hervorhebung nicht nützlich macht. Um hier Abhilfe zu schaffen, wird das CMP, auf das der Zeiger verweist, im sichtbaren Teil der CMP-Liste an die oberste Stelle gerückt. Diese Funktion wird in Abbildung 4.4 (a) und (b) demonstriert. Kann das Sichtfenster der CMP-Liste 20 CMPs anzeigen und der Zeiger hebt gerade das CMP mit der ID 32 hervor, zeigt das Sichtfenster der CMP-Liste die CMPs von ID 32 bis 52 an. Diese Funktion erspart das langwierige Suchen in der CMP-Liste nach dem korrekten Eintrag und lässt den Nutzer effizient die sichtbaren Voxel den einzelnen CMPs zuordnen. Um dem Problem der gleichgefärbten CMPs zu begegnen, wird im Abschnitt 4.6 eine Methode zur Farbspreizung vorgestellt.

Wie bereits im vorhergehenden Abschnitt ausgeführt, kann diese Zeige-Technik auch bei semitransparent dargestellten Voxeln dazu verwendet werden, einzelne Datenpunkte genau CMPs zuzuordnen. Zwar wird hier eine Farbverfälschung in Kauf genommen, jedoch kann durch die transparenten Objekte ein besserer Einblick in komplexe Strukturen wie einzelne Zellapparate gewährt werden, die von anderen Strukturen ummantelt sind. Mittels eines Sliders lässt sich die globale Opazität des Volumens zwischen 100% und 0% einstellen.

Selektive Transparenz von CMPs

Um Transparenz nutzen zu können, muss nicht unbedingt der Alpha-Wert jedes einzelnen Voxels herabgesetzt werden. Sinnvoller ist der Einsatz von Techniken, die nur bestimmte molekulare Muster durchsichtig erscheinen lassen. Um ein solches Resultat zu erreichen, werden hier zwei Ansätze verfolgt.

Die erste Möglichkeit besteht in der selektiven Transparenz, die dem Mauszeiger folgt. Es wurde die Beobachtung gemacht, dass häufig CMPs, die von sehr vielen Voxeln repräsentiert werden, die Sicht auf dahinterliegende Objekte verdecken. Zwar löst die globale Transparenz dieses Problem zum Teil, jedoch sind die verdeckten Elemente hier sehr schwer zu erkennen, da diese ebenfalls durchsichtig dargestellt werden. Aus diesem Grund wurde in die 3D-Ansicht eine Option implementiert, die das CMP, auf das der Zeiger gerade verweist, transparent darstellt und so die dahinter liegende Objekte angezeigt werden. Dies hat den zunächst unbeabsichtigten aber nützlichen Nebeneffekt,

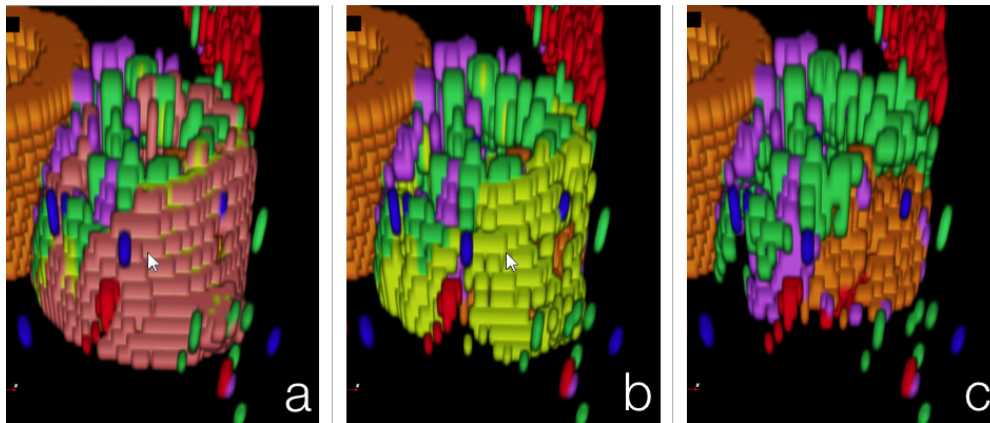


Abbildung 4.5: Abschälen von CMPs mittels Point-And-Click. (a) Lymphozyt mit verdeckter Struktur. Durch einen Klick auf das rosa CMP wird es entfernt. (b) ist das Ergebnisbild, auf dem nun ein gelb dargestelltes CMP die Sicht verdeckt. Ein Klick auf das CMP führt zu (c). So kann man sich Stück für Stück zu dem Kern einer Struktur vorarbeiten.

dass die Transparenzschaltung durch den Pop-out-Effekt präattentiv⁶ alle Voxel eines CMPs erkennbar macht. Das Muster hat eine geringere Opazität, aber die Umrisse sind immer noch eindeutig zu erkennen. Die Funktionalität ist in Abbildung 4.4 (c) aufgezeigt.

Diese Technik ist allerdings auf das jeweils oberste Muster limitiert, auf das der Cursor trifft. Befindet sich unter der transparenten Struktur ein weiteres CMP, das den Blick in den Zellapparat verdeckt, sind die Möglichkeiten dieser Technik erschöpft. Das Problem ist gleich dem Picking-Problem, welches bei der *regionalen* Hervorhebung auftritt, wie bereits in Abschnitt 3.2.2 beschrieben. Wie in [Preim und Ritter, 2002] vorgeschlagen, könnte das Problem mit Cutaway-Views oder ähnlichen Techniken umgangen werden, doch muss hierfür bekannt sein, welches Objekt der Nutzer genau untersuchen möchte - was bei der hypothesenfreien Exploration eines Toponomdatensatzes nur selten der Fall ist.

Die zweite Möglichkeit ist die Festlegung der Transparenz der einzelnen molekularen Muster. Es erscheint durch das Klicken auf ein CMP ein Slider, durch den die Opazität analog zur globalen Transparenz zwischen 100% und 0% eingestellt werden kann. Auf diesem Weg kann aufeinanderfolgend die Transparenz für die einzelnen sichtverdeckenden Objekte festgelegt werden, sodass verdeckte Objekte besser zur erkennen sind. Dies funktioniert jedoch nur solange ein CMP, wie es bei Oberflächenmustern möglich ist, eine Struktur nicht gänzlich umschließt. In diesen seltenen Fällen sind die vorgestellten Techniken erfolglos.

Eine Abhilfe hierzu wurde mit einer weiteren Technik geschaffen. Mit Hilfe der Des-

⁶Präattentive Wahrnehmung: Wenn eine Wahrnehmungsentscheidung eine feste Dauer unabhängig von der Anzahl der ablenkenden Faktoren aufweist, wird sie als präattentiv bezeichnet.

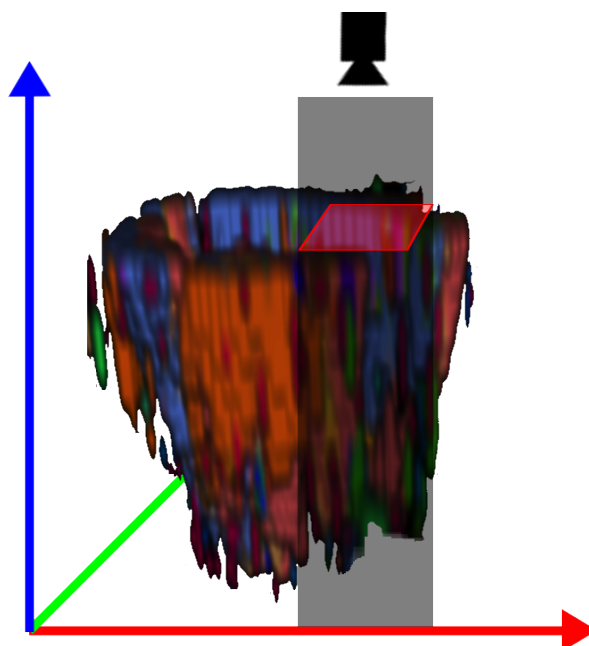


Abbildung 4.6: Lassotool zum Hervorheben von CMPs. Beispielhaft dargestellt ist ein Lymphozyt, wobei die Kamera den Blickwinkel des Nutzers repräsentiert. Das rot dargestellte Rechteck stellt die Selektion des Nutzers dar. Es werden nicht nur die CMPs hervorgehoben, die direkt im Rechteck enthalten sind, sondern ebenfalls jene, die Teil des grau dargestellten Selektionsblocks sind, da sonst verdeckte Strukturen nicht berücksichtigt werden. Das Volumen entspricht dem Ergebnis der Selektion.

elektion von CMPs durch Point-And-Click auf das zugehörige Voxel können Stück für Stück die verdeckenden CMPs “abgeschält” werden, um den Blick freizulegen (Abbildung 4.5).

Hervorheben von CMP-Komplexen

Bei Arbeiten in der Arbeitsgruppe “Molekulare Mustererkennung” zur Analyse von 2D-Toponomdaten konnte das implementierte Lassotool seinen großem Nutzen beweisen. Es wird zum einen verwendet, um heraus zu finden, was für CMPs sich in einem bestimmten Bereich befinden, und damit Aussagen über die Zusammensetzung von Strukturen zu treffen. Zum anderen wird es genutzt, um verschiedene Bereiche zu vergleichen, und so Strukturen zu finden, die in ihrem topologischen Muster von artverwandten Systemen abweichen. Hierbei ist vor allem von Interesse, ob bestimmte CMPs in einer pathologischen Struktur anwesend oder abwesend sind. Wird eine solche Beobachtung gemacht, liegt der Schluss nahe, dass die krankhafte Veränderung mit dem auffälligen molekularen Muster zusammenhängt.

Eine ähnliche Funktionalität sind durch die oben genannten Anforderungen auch für

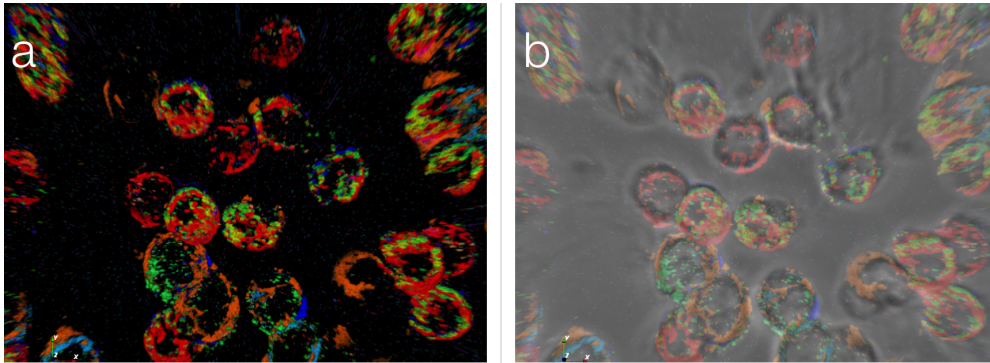


Abbildung 4.7: Unterstützung der Orientierung durch Einblenden eines Referenzvolumen. (a) Lymphozyten-Datensatz mit fünf visualisierten CMPs. (b) Referenzvolumen erstellt aus Phasenkontrastbildern des Markers C49F.

die 3D-Visualisierung nötig. Um das mentale Modell des Nutzers bezüglich der 2D-Lasso-Funktion beizubehalten, entspricht die Eingabe in 3D der in der 2D-Version. Der Nutzer wählt einen interessanten Bereich aus, indem er bei aktiviertem Lassotool mit gedrückter linker Maustaste einen Rahmen über die Struktur zieht, vergleichbar dem Selektieren von mehreren Dateien mit der Maus in WIMP-Benutzerschnittstellen⁷. Mit dem Loslassen der Maustaste beginnt die Berechnung der sich im Rahmen befindlichen CMPs. Da meist alle molekularen Muster eines Bereiches interessant sind, wie etwa bei dem Vergleich von zwei Instanzen einer Struktur, wurde der Selektionsalgorithmus so angepasst, dass die 2D-Eingabe des Lassotools einen Quader aus dem Volumen selektiert, also jede einzelne z-Ebene bei der Selektion berücksichtigt wird. Das Verhalten des Algorithmus wird in Abbildung 4.6 erklärt.

Der Algorithmus berechnet die Voxel, die vom linken oberen und rechten unteren Punkt des Auswahlrahmens begrenzt werden und ermittelt für diese die Pixelkoordinaten auf den 2D-Schichten. Anschließend wird für jedes Pixel innerhalb des aufgespannten Vierecks je Schicht berechnet, welches CMP ihm zugeordnet ist. Diese CMPs werden anschließend in der CMP-Liste selektiert, während alle weiteren CMPs, deselektiert werden.

Die in diesem Abschnitt vorgestellten Techniken ermöglichen es, die Muster des jeweiligen Datensatzes zu ermitteln. Im folgenden Abschnitt wird eine Möglichkeit vorgestellt, wie die gefundenen Strukturen in einen Kontext zur Lage im Gewebe gesetzt werden können.

⁷WIMP: **w**indow, **i**con, **m**enu, **p**ointing device - Fenster, Icon, Menü, Zeige-Eingabegerät.

Orientierung anhand eines Referenzvolumens

Grundlage der 2D-Visualisierung ist die Darstellung einer Phasenkontrastaufnahme⁸, die der MELK/TIS-Roboter bei der Datenaquirierung gewonnen hat. Die Aufnahme stellt eine Orientierungshilfe dar, um festzustellen, wo genau im Gewebe sich die analysierten Strukturen befinden. Üblicherweise wird hier ein Protein gewählt, für das genau bekannt ist, in welchen Zellstrukturen es zu finden ist.

In der 3D-Ansicht ist analog die Möglichkeit zu schaffen, ein Referenzvolumen als Orientierungshilfe anzuzeigen. Bei der Generierung wurde vergleichbar der Generierung des CMP-Volumens, die einzelnen Phasenkontrastbilder eines Epitops entlang der z-Achse übereinander “gestapelt” werden und die Zwischenräume der Datenpunkte durch trilineare Interpolationen angenähert. Dadurch, dass zwei Volumen übereinander gelegt werden, verdecken die Phasenkontrastbilder die CMP-Daten häufig gänzlich. Eine Lösung für dieses Problem ist das Herabsetzen der Opazität des Kontextvolumens. Es wird ein Defaultwert für die Transparenz eingestellt, der eine Balance zwischen Sichtbarkeit der relevanten Daten und Erkennbarkeit der Kontextdaten darstellt (Abbildung 4.7).

Ein mit der Transparenz einhergehendes Problem ist das “Verwischen” des Referenzvolumens bei geringer Opazität, wenn nah an das Volumen heran gezoomt, oder das Volumen in einem seitlichen Winkel betrachtet wird. Es drückt sich in einer unscharfen Darstellung des Datensatzes aus. Durch die Interpolation der Werte zwischen den Datenpunkten kommt es vor allem bei seitlichen Betrachtungen zu diesem Effekt. Dieses Problem wurde zunächst zurück gestellt. Ein Vorschlag für weitere Arbeiten wäre eine Darstellung des Referenzdatensatzes durch indirekte Volumenvisualisierung. Hierbei müsste ein geeignetes Verfahren entwickelt werden, das Isooberflächen aus den Daten erstellt, die anschließend - auch mit herabgesetzter Opazität - gerendert werden können. Ein weiteres Problem entsteht auch hier durch Transparenz erfolgte Farbverfälschung. Dadurch, dass es sich jedoch nur um Grautöne handelt, schwächen diese lediglich die Farbtintensität ab, sodass keine Farbtöne gemischt werden.

Neben den vorgestellten Werkzeugen zur Unterstützung der Exploration des Datensatzes ist es notwendig eine geeignete Technik zur nutzergerechten Navigation dieser Daten einzuführen. Welche Methoden hier angewandt werden, ist Gegenstand des folgenden Abschnitts.

4.4.4 Navigation im 3D-Raum

Im Abschnitt 3.2 wurden die Grundlagen und Anforderungen für eine effiziente Steuerung der Navigation in einer 3D-Szene diskutiert. Die zugrundeliegende Peripherie ist durch die Kombination von Maus und Tastatur vorgeschrieben. Alternative Eingabegeräte stehen nicht zur Verfügung.

Für die Navigation wird die in Abschnitt 3.2.3 vorgestellte *Two-Axis Valuator Track-*

⁸Phasenkontrastmikroskopie: Abbildungsverfahren der Lichtmikroskopie. Strukturen mit geringem Eigenkontrast werden abgebildet, indem die Phasenveränderung der Lichtwelle beim Durchgang durch das Medium gemessen wird.

ball-Technik verwendet, die nach [Bade et al., 2005] die besten Ergebnisse erzielt hat. Es wird jedoch nicht jede Mausbewegung sofort in eine Bewegung der Kamera umgewandelt, da sonst bei der Interaktion mit User-Interface Elementen immer der aktuell interessante Ausschnitt den Fokus verlieren würde. Die *Two-Axis Valuator Trackball*-Technik wird auf die Bewegung angewandt, wenn die linke Maustaste gedrückt und gehalten wird. Die rechte Maustaste aktiviert das Heranzoomen. Das ebenfalls zur Verfügung stehende Mausrad wird genutzt, um in den Datensatz hinein oder hinaus zu zoomen.

Es existieren zwei Modi der Interaktion. Der Standard-Modus verwendet die oben beschriebene Navigation. Der Selektionsmodus wird durch das Drücken der linken Maustaste als Indikator für eine Lasso-Auswahl eingeschaltet. Damit wurde die Aktivierung der Bewegung durch auf die rechte Maustaste verlegt. Zoomen ist aufgrund der redundanten Belegung der Funktion nur über das Mausrad möglich.

Die fehlende Schwerkraftmetapher kann es dem Nutzer, trotz eines angezeigten Kontextvolumens bisweilen schwer machen, korrekt einzuschätzen, in welchem Winkel er gerade auf das Volumen blickt und wo er sich im 3D-Raum befindet. Um diesem Problem zu begegnen, wird ein Orientierungswidget⁹ angezeigt, das die aktuelle Ausrichtung der Koordinatenachsen einblendet. Diese Technik wird in vielen Anwendungen der medizinischen Volumenvisualisierung sowie bei verschiedenen 3D-Modellierungssoftwarelösungen eingesetzt und bildet ein intuitives und effizientes Tool zur Orientierung des Nutzers.

Diese Visualisierungsmethode bildet die Grundlage für die Analyse der 3D-Toponomdaten und Basis von neuen, ausgeklügelteren computergestützten Hilfen bei der Exploration der biologischen Strukturen. Zusammengefasst bildet die hier vorgestellte Lösung ein weiteres Tool bei der Analyse der vorliegenden komplexen Daten und ermöglicht damit einen alternativen Einblick in die noch versteckte Semantik. Im folgenden Abschnitt wird ein weiteres Werkzeug vorgestellt, das eine von der Topologie losgelöste Analyse der Daten erlaubt und somit als unterstützende Darstellung zu verstehen ist.

4.5 Epitop-Graphvisualisierung

[Oeltze et al., 2011] haben bereits die Grundlagen dafür gelegt, wie Toponomdaten über eine Graphstruktur visualisiert werden können, sodass eine eindeutige Aussage darüber getroffen werden kann, wie ausgewählte Muster molekular zusammengesetzt sind, ohne stets den Umweg über die CMP-Liste gehen zu müssen. Anders als bei den klassischen 2D- und 3D-Ansichten werden bei dieser Ansicht nicht direkt die CMP-Daten in ihren gemessenen Orten visualisiert, sondern zu einer neuen Darstellung abstrahiert, die allein für sich kaum Schlüsse über Funktionskodierungen im Gewebe zulässt. Das Ziel dieses Werkzeuges ist die Bereitstellung zusätzlicher Informationen über eine gewählten Menge von molekularen Strukturen bezüglich ihres Aufbaus. Zwar wurde bereits, wie in Abschnitt 3.3 erläutert, umfangreich zu diesem Thema gearbeitet. die vorgestellten

⁹Widget (Nutzerschnittstelle): Element einer graphischen Nutzeroberfläche, das Informationen anzeigt, die durch den Nutzer beeinflusst werden können.

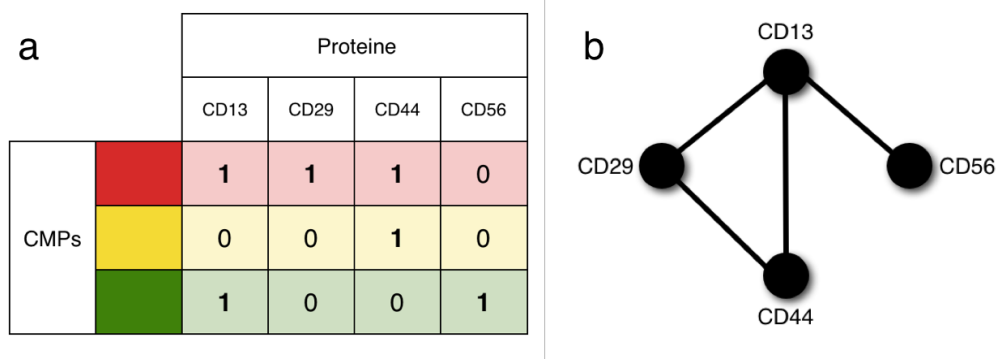


Abbildung 4.8: Beispiel für die Generierung eines Graphen. (a) CMP-Tabelle, die drei CMPs enthält, die aus der Messung von vier Proteinen entstanden sind. (b) Graph, der aus den drei CMPs aus (a) generiert wurde. Die Knoten repräsentieren die gemessenen Proteine. Eine Kante wird zwischen zwei Knoten gezogen, wenn in einem einzelnen CMP zwei oder mehr Proteine *anwesend* (1) sind. CMP 1 (Rot) generiert so drei Kanten: CD13-CD29; CD13-CD44; CD29-CD44. CMP 2 (Gelb) generiert keine Kanten, da nur CD44 anwesend ist. CMP 3 (Grün) generiert die Kante CD13-CD56.

Lösungen waren jedoch nicht direkt nutzbar, da sie nicht in MultiCompare integriert waren. Dies soll nunmehr realisiert werden.

Darüber hinaus ist es möglich, weitere statistische Merkmale über einen Graphen zu visualisieren, sodass dem Nutzer auf intuitivem Wege möglichst viele Informationen vermittelt werden, ohne dessen Auffassungsgabe zu überlasten. Graphen können zusätzlich dazu genutzt werden, verschiedene molekulare Strukturen miteinander zu vergleichen, indem Schnitt- oder disjunkte Mengen von Mustern auf einem einzelnen Graphen abgebildet werden. So ist schnell ersichtlich, welche Aspekte sich in verschiedenen (vielleicht sogar funktionell gleichen) Geweben unterscheiden.

Wie der Graph selbst aufgebaut ist und wie welche Informationen darin kodiert werden, bildet die Grundlage des ersten Abschnitts. Anschließend wird auf die Interaktion des Nutzers mit dem Graph eingegangen. Wie verschiedene Netzwerke durch Vergleichsgraphen analysiert werden können, wird abschließend in diesem Kapitel diskutiert.

4.5.1 Generierung des Graphen

Bei der Generierung des Graphen orientiert sich diese Arbeit an [Oeltze et al., 2011]. Die Knoten im Graphen repräsentieren die gemessenen Epitope. Wurden für die Datensatzgenerierung vom MELK/TIS-Roboter 20 Durchläufe pro Schicht durchgeführt, resultiert der Graph in 20 Knoten. Zwischen den einzelnen Knoten wird eine Verbindung (Kante) hergestellt, wenn sie gemeinsam in einem CMP vorkommen. Der Graph für eine Menge von molekularen Netzwerken repräsentiert also deren Zusammensetzung ohne Berücksichtigung der Topologie der enthaltenen CMPs.

Abbildung 4.8 zeigt ein vereinfachtes Beispiel für die Graphgenerierung. Um einen Graph aus CMPs zu erstellen, wird, nachdem für jedes gemessene Epitop ein Knoten gebildet wird, der sogenannte *CMP-String* verwendet, der in Form einer Kette von 0en und 1en kodiert, welche Proteine das CMP enthalten. Der *CMP-String* ist gleichzusetzen mit dem Eintrag des CMPs in der CMP-Liste, der für drei CMPs in Abbildung 4.8 (a) zu sehen ist. Das erste rot kodierte CMP hat nach dieser Definition den String 1110. Der Algorithmus zur Generierung der Kanten durchläuft nun alle möglichen Paare von *anwesenden* Proteinen (1en) und erstellt zwischen den zugehörigen Knoten eine Kante. Für das erste CMP wäre dies zunächst das Paar CD13 und CD29. Die weiteren Paare und damit Kanten, die durch dieses CMP erzeugt werden, sind CD13 und CD44 sowie CD29 und CD44. Im zweiten, gelben CMP wird keine Kante generiert, da keine Paare von *anwesenden* Proteinen vorliegen. Das letzte, grüne CMP erstellt eine Kante zwischen CD13 und CD56, die das einzige Paar in diesem CMP repräsentiert. Es resultiert der Graph, der in Abbildung 4.8 (b) dargestellt ist.

Dieses Beispiel repräsentiert nicht die Komplexität eines realen Toponomdatensatzes, der eine Vielzahl von Epitopen und mehreren tausend CMPs enthält. Es stellt sich die Frage, welche Netzwerke durch den Graph dargestellt werden sollen und zu welchem Zeitpunkt die Struktur aktualisiert wird. Werden stets die Verbindungen visualisiert, die von allen im Datensatz enthaltenen CMPs hergestellt werden, erhält man in den meisten Fällen einen vollständigen Graphen - eine Struktur, in der jeder Knoten durch eine Kante miteinander verbunden ist. Dies ist nicht der gewünschte Informationsgewinn.

Da die Graphdarstellung als ein unterstützendes Tool wahrgenommen eingesetzt werden soll, liegt der Schluss nahe, es an die 2D- bzw. 3D-Ansichten zu koppeln. Wie in Abschnitt 4.1 beschrieben, werden meist nur ausgewählte CMPs dargestellt, um die Übersicht zu bewahren und die für bestimmte Vorgänge interessante Muster Stück für Stück zu destillieren. Alle CMPs werden allenfalls für eine erste Orientierung in einem Datensatz auf einmal eingeblendet. Die Selektion der dargestellten CMPs erfolgt entweder über die CMP-Liste oder das Lassotool. Dies führt zu dem Ansatz, den Graphen immer für die Menge von CMPs zu generieren, die aktuell visualisiert werden, um hier zusätzliche Informationen einholen zu können. Die Aktualisierung des Graphen ist direkt an die ausgewählten CMPs in der CMP-Liste gekoppelt, da das Lassotool nur ermittelt, welche CMPs zu dem selektierten Bereich gehören und diese in der CMP-Liste selektiert.

4.5.2 Anordnung und Parameter

Bezüglich der Anordnung der Knoten des Graphen wurde den Ergebnissen von Oeltze et al. gefolgt, die ein kreisförmiges Layout favorisieren. Bezüglich der Überschneidungen der Kanten ist eine kreisförmige Darstellung selten die optimale Lösung. Würde die Darstellung nach Kantenüberschneidungen optimiert werden, wäre das Resultat nach jeder Selektion eine andere Anordnung der Knoten. Genau das ist jedoch ein Zustand, der nicht erstrebenswert ist. Im Zuge der Reduzierung der kognitiven Last des Nutzers ist es nicht ratsam, die Position der Knoten zu ändern, da diese bei Konsultation des

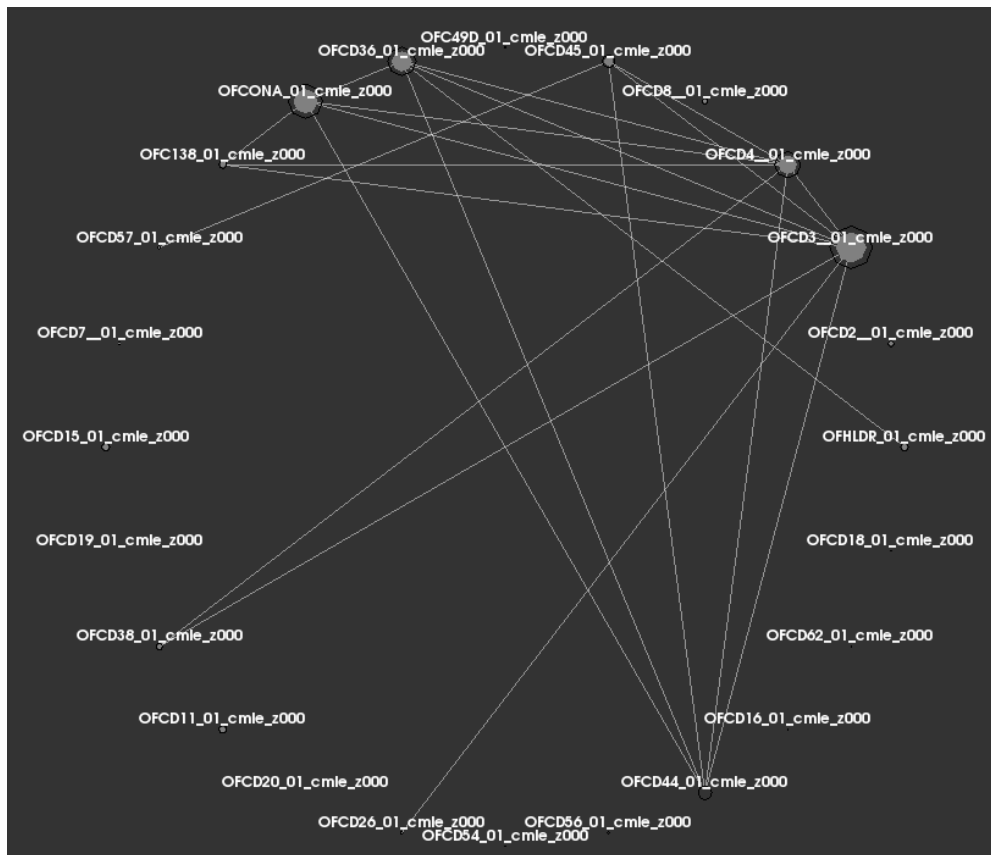


Abbildung 4.9: Graphdarstellung von 26 CMPs. Die Daten wurden aus einem Lymphozytendatensatz mit 32 gemessenen Epitopen entnommen. Es scheint hier eine Korrelation zwischen Gesamtvorkommen des Epitops (abgebildet auf Knotengröße) und der Anzahl der zugehörigen Kanten zu geben.

Graphen zu einer bestimmten Information stets neu lokalisiert werden müssen. Werden sie alphabetisch oder anhand der Reihenfolge im *CMP-String* entlang eines Kreises angeordnet, fällt die Suche nach interessanten Knoten leichter. Aus diesen Gründen wird das kreisförmige Graphlayout nach [Oeltze et al., 2011] ebenfalls favorisiert. Damit die einzelnen Knoten den Epitopen zugeordnet werden können, ist neben ihnen ein Label mit dem Epitopnamen zu sehen, wie in Abbildung 4.8 (b) beispielhaft und in Abbildung 4.9 in Anwendung dargestellt.

Werden viele CMPs selektiert, kommt es häufig vor, dass ein Großteil der Knoten jeweils miteinander verbunden ist. Es ist damit schwer zu erkennen, ob zwei Knoten nicht durch eine Kante verbunden sind, da der entstandene Graph sehr unübersichtlich ist. Abhilfe schafft die Berechnung eines Invers-Graph aus der aktuellen Struktur. So

können komplexe Graphen vereinfacht werden. Der Nutzer entscheidet durch eine Option, ob der Graph original oder invers gezeichnet werden soll.

Der Graph kann dazu verwendet werden, weitere Merkmale der Toponomdaten darzustellen, wenn man ihn mit weiteren visuellen Parametern ausstattet, die von der zugrunde liegenden Menge von CMPs beeinflusst werden. Ein Beispiel hierfür ist das Abbilden des Gesamtvorkommen eines einzelnen Proteins im gesamten Datensatz auf den zugehörigen Knoten. Als visueller Parameter wird die Knotengröße definiert. Es gibt eine minimale und maximale Größe, die jeweils von den Knoten angenommen wird, welche das geringste/höchste Vorkommen im Datensatz aufweisen. Die Elemente dazwischen werden relativ zum Minimum und Maximum in Abhängigkeit von ihrer Häufigkeit dargestellt. Diese Darstellung ist eine Alternative zu der von Oeltze et al. entwickelten Glyphendarstellung (Abschnitt 3.3.3). Auf diese Art ist es möglich, sowohl globale Eigenschaften wie Gesamtvorkommen eines Epitops im Datensatz, als auch lokale Eigenschaften wie die Zusammensetzung des aktuell ausgewählten Netzwerks in eine Darstellung zu integrieren. Wie später in Abschnitt 5.4.2 beschrieben, ist es auch möglich, die Struktur des aktuell im 3D- oder 2D-View ausgewählten CMPs mittels Einfärbung der Knoten in der Graphvisualisierung darzustellen (z.B. Epitop anwesend: roter Knoten, Epitop abwesend: blauer Knoten).

Oeltze et al. haben noch mit weiteren visuellen Parametern, wie z.B. der Kantendicke, experimentiert. Wenn mehrere CMPs dasselbe Paar von Proteinen aufweisen, ist dies bislang nicht ersichtlich, denn die Kante ist bereits durch das erste Vorkommen des Paares gezeichnet. Es wird vorgeschlagen diese Häufigkeit auf die Kantendicke abzubilden, also mit jedem weiteren Auftreten der Verbindung in einem CMP die Kantendicke zu erhöhen. Bei einer solchen Darstellung kann auch der Graph aller CMPs interessant sein, da von diesem trotz der geringen Übersicht zu erkennen ist, welche Verbindungen besonders häufig vorkommen. Diese Möglichkeit wurde jedoch noch nicht in die vorgestellte Lösung integriert, ist aber eine vielversprechende Funktion und sollte bei zukünftigen Arbeiten berücksichtigt werden. Darüber hinaus bildet die Graphdarstellung die Grundlage für viele weitere Möglichkeiten, von denen einige im Abschnitt 4.5.4 erläutert werden.

4.5.3 Interaktion mit dem Graphen

Die Beeinflussung des Graphen durch den Nutzer erfolgt streng genommen nur indirekt über die CMP-Liste, die wiederum indirekt durch das Lassotool beeinflusst werden kann. Es ist keine direkte Interaktion mit dem Graphen nötig, um die kontextuellen Informationen dieses Werkzeugs dargestellt zu bekommen. Dies ist auch das Ziel, da sie lediglich einem unterstützenden Zweck dient. Die Kamera wird über dem Graphen positioniert, sodass stets der gesamte Graph, sichtbar sind (Abbildung 4.9). Darüber hinaus ist es möglich, einen bestimmten Bereich des Graphen in den Fokus zu rücken, indem die Kamera ähnlich der 3D-Navigation beeinflusst wird. Die Maus bildet ihre beiden Freiheitsgrade bei gedrückter mittlerer Taste direkt auf die Verschiebung der x-y-Koordinaten der Kamera ab. Die linke Maustaste wurde für die Verschiebung der Kamera nicht verwendet, da diese für ein Auswahlwerkzeug vorgesehen ist, das im

nächsten Absatz beschrieben wird. Über die rechte Maustaste und das Musrad wird analog zur 3D-Navigation an die Struktur heran bzw. von ihr weggezoomt.

Neben der Navigationstechnik wurden zwei weitere Interaktionsmodi implementiert, die den Graph um weitere Funktionalitäten erweitern, die über eine reine Informationsdarstellung hinaus gehen. Im ersten Modus wird der Graph dazu verwendet, um ein Muster zu erstellen, nachdem die CMPs eines Datensatzes gefiltert werden können. Eine vergleichbare Möglichkeit wurde bereits in einem früheren Werkzeug zur Verfügung gestellt, indem für jedes Epitop in einem Dropdown-Menü ein Filtermuster festgelegt. Hierbei stehen folgende Möglichkeiten pro Epitop zur Verfügung: “1” (anwesend); “0” (abwesend); “?” (Epitop für Filter nicht relevant). Ein solches Muster lässt sich in diesem Interaktionsmodus auch direkt anhand des Graph erstellen. Ist der Modus aktiviert, haben alle Knoten den Wert “?”, und sind grau dargestellt. Ein Linksklick auf einen der Knoten ändert den Wert auf “1”, die Farbe ändert sich in rot. Ein weiterer Klick lässt den Knoten den Wert “0” annehmen und stellt ihn blau dar. Der Ausgangswert wird wieder durch einen Linksklick hergestellt. So kann das gewünschte Muster schnell erstellt werden. Weitere Details hierzu werden in Abschnitt 5.4.2 beschrieben.

Der zweite Interaktionsmodus hat keinen direkten Einfluss auf die dargestellten Daten sondern dient der besseren Erkennung von Kanten zwischen selektierten Knoten. Durch einen Mausklick oder dem Ziehen eines Rahmens bei gedrückter linker Maustaste werden Kanten und Knoten, die sich im selektierten Bereich befinden, farblich hervorgehoben. Werden mehrere Knoten selektiert, werden die Kanten, die eine Verbindung zwischen diesen herstellen, farblich hervorgehoben.

4.5.4 Generierung eines Vergleichsgraphen

Die Möglichkeiten der Graphvisualisierung im Anwendungsbereich der Toponomvisualisierung sind bisher wenig erkundet. Ein Ansatz zur Verwendung dieser Visualisierung ist die Generierung eines Vergleichsgraphen von Geweben, um so molekulare Unterschiede bzw. Gemeinsamkeiten erkennbar zu machen. Für diese Technik wird in dieser Arbeit die Grundlage geschaffen.

Grundlage für die Überlegungen hinter dieser Graphdarstellung sind Beobachtungen, die beim praktischen Einsatz der Software gemacht wurde. Das Lassotool eignet sich hervorragend, um die CMPs eines bestimmten Bereiches hervorzuheben und so genau zu erkennen, welche molekularen Strukturen vorliegen. Die parallele hierzu generierte Graphansicht verhilft dem Nutzer zusätzlich zu einem Einblick in die Verbindung der einzelnen Epitope des selektierten Bereichs. Häufig wird das Lassotool jedoch auch dafür eingesetzt, um Unterschiede zwischen zwei Bereichen des Datensatzes zu erkunden, z.B. bei der Analyse eines Lymphozytendatensatzes, bei dem mehrere weiße Blutkörperchen miteinander verglichen werden sollen. Neben den Unterschieden bezüglich der einzelnen CMPs - erkennbar durch die Schnittmenge der CMPs durch die Selektion beider Bereiche - ist hier auch die molekulare Zusammensetzung dieser Bereiche interessant. Welche Proteine kommen in einem Bereich vor, in einem anderen jedoch nicht? Welche Muster sind immer gleich? Eine Möglichkeit, diese Fragen zu

beantworten, ist ein Vergleichsgraph, generiert aus den selektierten CMPs der beiden Bereiche.

Als Basis für die Generierung eines Vergleichsgraphen müssen durch das Lassotool zwei oder mehr Bereiche ausgewählt werden, woraus eine Menge von CMPs generiert wird, aus denen jeweils für sich genommen ein Graph gemäß Abschnitt 4.5.1 entsteht. Anschließend werden die Unterschiede zwischen den Graphen berechnet, also Kanten ermittelt, die durch alle oder nur einer Teilmenge von Graphen repräsentiert werden sowie solche Kanten zugeordnet, die nur in einem einzelnen Graphen vorkommen. Daraufhin werden diese Daten in einem einzigen Graph dargestellt. Um Gemeinsamkeiten und Unterschiede in den Kantenverläufen zu verdeutlichen, eignet sich als visueller Parameter die Farbe der Kanten. Für einen Vergleich von zwei Bereichen existieren drei verschiedene Arten von Kanten, die dementsprechend mit drei verschiedenen Farben kodiert werden müssen: Kanten, die in beiden Graphen vorkommen und Kanten, die nur in jeweils einem Graph vorkommen. Fügt man weitere Bereiche hinzu, erreicht auch die Kodierung der Kanten mit jedem neuen Element eine erhöhte Komplexität. Es stellt sich die Frage, wie Kanten kenntlich gemacht werden, die von mehreren, aber nicht allen Graphen repräsentiert werden. Der einfachste Einsatz ist das Einführen entsprechend vieler neuer Farben und die Zuordnung dieser über eine Legende. Alternativ ist es denkbar, für alle Paare von Graphen einen Vergleichsgraphen zu erstellen und anschließend, ähnlich einer Scatterplot-Matrix, eine Graphen-Matrix zu erstellen, anhand derer jeweils zwei Bereiche verglichen werden. Diese Idee könnte auch mit einer interaktiven Komponente versehen werden, indem aus den einzelnen Graphen der Matrix eine Auswahl getroffen wird, wenn aus mehreren Bereichen ein komplexerer Vergleichsgraph, erstellt werden soll.

Ein Problem bei der Kodierung von Kanten durch Farbe ist die Wahl der Kantendicke. Ist eine Kante nur durch eine 1-Pixel breite Gerade repräsentiert, ist es für den Nutzer schwer, eine Farbe eindeutig zuzuordnen - je mehr Platz der Kante eingeräumt wird, desto besser ist die Farbe für den Nutzer erkennbar. Es muss allerdings ein Kompromiss zwischen Kantendicke und eindeutiger Zuordenbarkeit der Kanten gefunden werden, da dickere Kanten eine größere Gefahr von Überdeckungen bedeuten. Bei drei verschiedenen Farben, wird eine geringere Dicke vonnöten sein, als bei vielen verwendeten Farben, wenn mehrere Bereiche durch einen Graphen verglichen werden sollen.

Es bieten sich hier viele Möglichkeiten, weitere statistische Merkmale in den Vergleichsgraph zu implementieren, wie etwa die Häufigkeit der Epitope im jeweiligen Bereich (z.B. über Knotengröße sortiert) oder die Häufigkeit der Verknüpfungen über die Kantendicke, ähnlich wie in [Oeltze et al., 2011] vorgeschlagen.

Der nächste Abschnitt beschäftigt sich mit der algorithmischen Farbgenerierung, das in der Visualisierung der CMP-Daten Anwendung findet und eine möglichst hohe Unterscheidbarkeit der Daten sicherstellen soll. Weiterhin wäre es auch bei der Generierung eines Vergleichsgraphen einsetzbar, wenn man die Unterschiede von Kanten farblich hervorheben möchte.

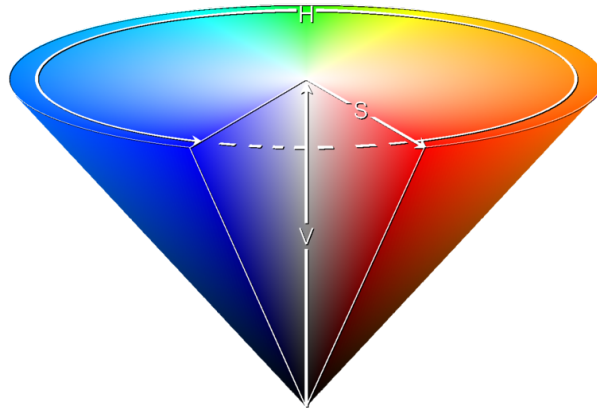


Abbildung 4.10: Darstellung des HSV-Farbraums. Dieser setzt sich aus folgenden Parametern zusammen: **H** Farbtön (hue); **S** Sättigung (saturation); **V** Helligkeitswert (value) [HSV, 2011].

4.6 Algorithmische Farbgenerierung

Die Notwendigkeit für das Generieren möglichst vieler perzeptuell unterscheidbarer Farben ergibt sich aus der Visualisierung der CMP-Datenpunkte durch farbkodierte Pixel/Voxel. Eine Abbildung der Klassen auf Farben ergibt dann einen Sinn, wenn die Farben vom Nutzer unterschieden werden können. Im Laboralltag werden die Biologen häufig zunächst alle molekularen Muster gleichzeitig darstellen, um sich einen Überblick zu verschaffen. Eine eindeutige Unterscheidung von mehr als 30.000 unterschiedlichen CMPs durch Farabbildung ist nach aktuellem Stand nicht möglich.

Das Ziel ist ein Algorithmus, der Farben mit großen perzeptuellen Abstand zueinander in Abhängigkeit von der Anzahl der dargestellten Muster erstellt. Die bisherige Implementation färbt die CMPs unabhängig von der Anzahl der zu visualisierenden Elemente in Tönen von rot nach grün, von grün nach blau und von blau nach rot. Ein solcher Zyklus zwischen den Farbtönen generiert 14 Farben. Anschließend beginnt der Algorithmus wieder von rot, jedoch wird die Farbhelligkeit reduziert, sodass mit jedem Durchlauf dunklerer Farben generiert werden. Am Ende werden so 252 verschiedene Farben erstellt. Der Algorithmus reizt das menschlich wahrnehmbare Farbspektrum nicht aus. Da der Farbgenerierungsalgorithmus jedoch einen Grundpfeiler bei der Visualisierung darstellt, ist er Gegenstand von Verbesserungsvorschlägen, die in diesem Kapitel hergeleitet werden.

4.6.1 Entwicklung des Algorithmus

Bevor die Anwendung des Algorithmus diskutiert wird, soll zunächst dessen Arbeitsweise beschrieben und Parameter und Abhängigkeiten herausgearbeitet werden, die

auch Limitierungen in der Anwendung bedeuten können.

Die Grundlage der Berechnungen bildet eine möglichst große Menge von verschiedenen Farben, die anschließend verwendet werden, um diejenigen zu bestimmen, die perzeptuell den größten Abstand zueinander besitzen. Dies wird über den HSV-Farbraum realisiert, dessen Aufbau in Abbildung 4.10 zu sehen ist. Der Algorithmus geht in einer kleinen Schrittweite entlang der Farbtonpalette (**H**). Ist dieser wieder bei der Ausgangsfarbe angelangt, wird der Sättigungswert (**S**) verringert und wieder erneut entlang der Farbpalette neue Farbwerte generiert. Dies wird so lange wiederholt, bis eine empirisch bestimmte Grenze der Sättigung erreicht wird. Dann wird der Helligkeitswert (**V**) verringert und der Algorithmus startet erneut - bis auch der Helligkeitswert eine vorher festgelegte Grenze erreicht. Der Algorithmus terminiert an dieser Stelle. Es werden - abhängig von den gewählten Schrittweiten - mehrere tausend Farben generiert.

Anschließend werden die Farben ermittelt, die den größten perzeptuellen Abstand zueinander besitzen. Als Eingabe wird die Anzahl der benötigten Farben angegeben. Je mehr Farben benötigt werden, desto geringer ist folglich der wahrgenommene Abstand zwischen den generierten Farben. Der in Abschnitt 3.4 vorgestellte Lab-Farbraum wird verwendet, um die perzeptuellen Abstände zu berechnen. Im Jahr 1994 wurde vom CIE-Konsortium ein erweiterter Algorithmus vorgestellt, der neben dem einfachen euklidischen Abstand weitere empirisch bestimmte Variablen in die Berechnung mit einfließen lässt. Die Methode wurde 2000 zuletzt aktualisiert und bildet nun eine umfangreiche Sammlung von Berechnungsvorschriften für den Farbunterschied. Sie stellt eine Reihe von Ausgleichsformeln für Unterschiede in Farbton, Helligkeit und Sättigung zur Verfügung. Die komplexe Implementation wurde von [Sharma et al., 2005] detailliert aufgearbeitet.

Für die Berechnung der Farbabstände wurden alle drei Methoden im Lab-Farbraum implementiert - sowohl der euklidische Abstand als auch die CIE-Berechnungsvorschriften von 1994 und 2000. Erwartungsgemäß wäre aufgrund der Berücksichtigung der verschiedensten Einflüsse die aktuellste Methode aus dem Jahr 2000 vorzuziehen, doch die besten Ergebnisse wurden mit CIE-1994 erreicht. Die Bewertung erfolgte über die Verteilung der generierten Farben auf dem Farbspektrum. Darüber hinaus sollten bei geringen Farbmengen (<20) möglichst wenige Farben des gleichen Farbtons generiert werden. Warum dieser Ansatz besser für den Algorithmus funktioniert konnte trotz intensiver Recherche nicht ermittelt werden. Eine Vermutung beruht auf der hohen Anzahl von empirisch bestimmten Konstanten in der CIE-2000 Methode, deren Bestimmung nicht ausführlich genug beschrieben ist und die vermutlich nicht für alle Anwendungsgebiete gleich sind. Die ältere Methode enthält nur ein kleines Set solcher Konstanten, dessen Werte für verschiedene Anwendungen genau vorgegeben sind.

Der Algorithmus berechnet den maximalen Abstand, den die Farben zueinander haben können, um ausreichend unterschiedliche Farben zu erhalten. Das Ergebnis ist eine Menge von in den RGB-Raum transferierten Farben. Wie der Algorithmus effizient angewendet wird und warum es nicht sinnvoll ist, immer den Farbraum im aktuell sichtbaren Bereich zu spreizen, ist Gegenstand des nächsten Abschnitts.

4.6.2 Anwendung

Primär wurde der Farbalgorithmus entwickelt, um die bisherige, nicht zufriedenstellende Einfärbung der CMPs zu ersetzen. Eine eindeutige Unterscheidung von mehreren Tausend unterschiedlichen CMPs allein durch Farbe ist jedoch auch durch diese Methode nicht möglich.

Durch die Eigenschaft des Algorithmus die Qualität der perzeptuellen Unterscheidbarkeit mit der Anzahl der zu generierenden Farben zu skalieren, ergeben sich weitere Anwendungsmöglichkeiten, die nachfolgend beschrieben werden.

Dynamische Farbspreizung

Der naheliegende Ansatz für den Einsatz dieses Algorithmus ist die stete Anwendung auf das aktuelle Sichtfeld im 2D- oder 3D-View. Die Farben der aktuell dargestellten CMPs werden mittels des Algorithmus so gespreizt, dass der Farbraum stets optimal ausgereizt wird und eine optimale Unterscheidbarkeit der Farben zu jedem Zeitpunkt gewährleistet wird. Zu diesem Zweck wird für jedes Bildpixel via Raycasting ermittelt, ob ein Voxel getroffen wird und zu welchem CMP es gehört. Das Ergebnis ist eine Menge von CMPs, über die der Farbalgorithmus neue Farben zuordnet. Dieses Vorgehen hätte ebenfalls zur Folge, dass die Farben der dargestellten CMPs häufig bei der Interaktion wechseln und sich somit auch die Repräsentation in der CMP-Tabelle ändert. Dies wäre kontraproduktiv bzgl. der kognitiven Belastung des Nutzers. Bei der Exploration der Datensätze ist es wichtig, dass den einzelnen Farben die CMPs zugeordnet werden, die von ihnen repräsentiert werden (z.B. durch die vorgestellte Pointing-Technik). Ändern sich nun stets die Farben der dargestellten CMPs, muss der Nutzer ggf. häufig die Abbildung der Farben auf die CMPs prüfen und neu verinnerlichen. Dieser Nachteil macht die dynamische Farbspreizung in der Praxis nur schwer einsetzbar. Ein Abhilfe würde hier ein Annotationssystem schaffen, welches später in Abschnitt 6.3 eingeführt wird.

Dass die Farbspreizung ein sinnvolles Tool sein kann, ist an einem Beispiel in Abbildung 4.11 zu erkennen. Während unter (a) der Eindruck entsteht, dass nur ein einzelnes CMP dargestellt ist, wird bei einer Farbspreizung sichtbar, dass es sich um zwei CMPs handelt (b). Zwar kann diese Information auch über die Kontextinformation erhalten werden, indem der Cursor über die einzelnen Voxel fährt, jedoch müsste für die einwandfreie Validierung der Farben jedes einzelne sichtbare Voxel betrachtet werden. Die Farbspreizung bildet ein nützliches Werkzeug, das dem Nutzer zur Verfügung gestellt wird, um Verdachtsmomente zu bestätigen. Da die ständige Anwendung der Farbspreizung bei aktuellem Funktionsumfang der Software nicht praktikabel ist, wird sie durch einen Button ausgelöst. Zwar existiert so immer noch das Problem, dass man sich für den sichtbaren Bereich die neuen Farben einprägen muss, jedoch umgeht man Probleme, wie sie in Abbildung 4.11 (a) dargestellt sind.

Die Einfärbung der CMPs ist dennoch längst nicht optimal. So wäre ein weiterer vielversprechender Ansatz, die topologischen Lage der CMPs zu berücksichtigen, sodass benachbarte CMPs möglichst von Farben repräsentiert werden, die einen großen

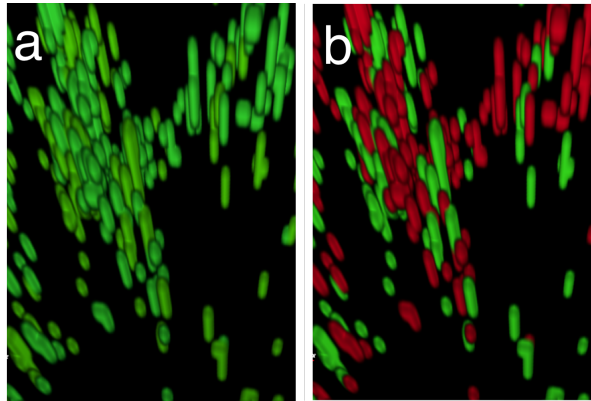


Abbildung 4.11: Farbspreizung von zwei CMPs. (a) Der betrachtete Bildausschnitt erweckt den Anschein, nur ein einziges CMP zu enthalten. (b) Die Farben der sichtbaren CMPs wurden gespreizt, sodass das menschlich wahrnehmbare Spektrum besser ausgereizt wird. Es ist zu erkennen, dass zwei CMPs im sichtbaren Ausschnitt dargestellt sind.

perzeptuellen Abstand zueinander haben. Eine solche Optimierung würde auch Probleme, wie in Abbildung 4.11 (a) abgebildet, umgehen.

4.7 Zusammenfassung

Auf den bewährten Arbeitsabläufen bei der Analyse der Toponomdaten aufbauend wurde eine auf direkten Volumenvisualisierung basierende Darstellung der Daten entworfen, die auf der Abbildung der unterschiedlichen CMPs auf Farbe fußt. Überdeckungsprobleme wurden durch lokale und globale Transparenz umgangen. Ebenso wurde eine interaktive Schnittstelle entworfen, sodass farbkodierte Muster schnell in ihre Bestandteile aufgeschlüsselt werden können. Die Navigation und Selektion bestimmter Strukturen wurde diskutiert. Die Graphvisualisierung baut als unterstützende Visualisierung von Oeltze et al. auf und führt die vorgestellten Konzepte weiter. Weiterhin wurde erklärt, wie durch einen Vergleichsgraphen Gewebsschnitte detailliert gegenübergestellt werden können. Die Entwicklung des auf dem CIELab-Farbraum als Unterscheidungsmesser aufbauenden Farbgenerierungsalgorithmus zusammen mit dessen Einsatzmöglichkeiten in der 3D- und Graphvisualisierung bildeten den Abschluss des Kapitels.

5 Implementation der Visualisierungen

In diesem Kapitel werden implementationsrelevante Details, wie das zugrundeliegende Dateiformat sowie die verwendeten Bibliotheken und Umgebungen vorgestellt. Das Hauptaugenmerk liegt hier in der Klärung von interessanten Fragen bezüglich der Einbindung der Visualisierungskomponenten in eine vorhandene komplexe Software. Ebenso wurden während der Implementation Schwachstellen lokalisiert, deren Behebung notwendig für zukünftige Funktionserweiterung sind. Den Abschluss des Kapitels bildet das User-Interface-Konzept, das nicht nur den Aufbau der einzelnen Nutzerschnittstellen beschreibt, sondern auch Ideen diskutiert, wie die einzelnen Werkzeuge effizient miteinander kommunizieren. Da die verwendete Software viele einzelne Ansichten besitzt, wird weiterhin vorgeschlagen, wie die einzelnen Werkzeuge sinnvoll auf dem Bildschirm anzuordnen sind, um einen effektiven Workflow zu gewährleisten.

5.1 Dateiformat

In Kapitel 2 wurde beschrieben, wie CMPs aus den gemessenen Fluoreszenzbildern erstellt werden. Im vorangegangenen Kapitel wurde in Abbildung 4.2 (a) dieser Schritt beschrieben. Nach der Binarisierung der Fluoreszenzbilder werden die Daten im XML¹-Format abgelegt [Klemm, 2010, S.21]. Das XML-Format wurde aufgrund seiner Offenheit und großer Interoperabilität gewählt, eine breite Unterstützung durch viele Programmiersprachen und Anwendungen ist gewährleistet. Die generierten Daten sind damit auch für andere Anwendungen zugreifbar. Innerhalb der XML-Datei werden die CMPs durch ihren CMP-String identifiziert, ihnen zugeordnet sind alle Pixel in Form ihrer Koordinaten. Für 3D-Daten liegt jeweils pro Schicht eine einzelne XML-Datei vor. Aufgrund der Komplexität der Daten können die XML-Dateien leicht mehrere Megabyte Speicher einnehmen.

Für die Generierung einer 3D-Darstellung werden die einzelnen XML-Dateien nacheinander abgearbeitet. Intern ist die Datenstruktur der Pixelpositionen der CMPs um eine Ebenen-Variable erweitert.

5.2 Programmierwerkzeuge

Da in dieser Arbeit auf der Software aufgesetzt wurde, die sich bereits im praktischen Einsatz befindet, sind bereits Vorentscheidungen bezüglich der Programmiersprache und des verwendeten Betriebssystems getroffen.

¹Extensible Markup Language: Auszeichnungssprache zur Darstellung hierarchisch strukturierter Datensätze in Form von Textdaten.

MultiCompare ist in der objektorientierten² Programmiersprache C#³ geschrieben. Als Softwareplattform fungiert *Microsoft .NET*⁴, wodurch das Programm nur auf Systemen ausgeführt werden kann, die eine entsprechende Unterstützung mit sich bringen, wie *Microsoft Windows XP*, *Microsoft Windows Vista* und *Microsoft Windows 7*. Eine Ausführung auf *Apple Mac OSX*- oder Open-Source-Systemen ist nicht möglich. Eine entsprechende Kompatibilität wurde unter Nutzung des *MONO-Projekts* [MON, 2011] angestrebt, jedoch verhindern die verwendeten Bibliotheken bei Stand dieser Arbeit einen entsprechenden Einsatz, da diese *MONO* nicht unterstützen.

Die Entwicklung von MultiCompare findet in *Microsoft Visual Studio 2008* statt, einer IDE⁵, die explizit für die Entwicklung von Anwendungen für das *.NET*-Framework ausgelegt ist und eine Reihe nützlicher Features enthält (Interface-Builder, Debugging-Routinen, Code-Vervollständigung u.v.m.).

Die wichtigste Entscheidung war die Wahl geeigneter Bibliotheken für die Volumen- und Graphvisualisierung. Es wurden zu Beginn dieser Arbeit verschiedene Frameworks auf Tauglichkeit bezüglich der Anforderungen evaluiert. Am Ende des Entscheidungsprozesses fiel die Wahl auf das *Visualization Toolkit* (kurz: *VTK*) von *Kitware* [VTK, 2011a]. Ausschlaggebend war der umfassende Funktionsumfang, sowohl im Gebiet der direkten Volumenvisualisierung, als auch bezüglich der Graphendarstellung. Darüber hinaus ist VTK in medizinischen Anwendungen sehr populär, wodurch im offiziellen Wiki sehr viele gut dokumentierte Beispiele zu finden sind, welche die Einarbeitungszeit reduzieren [VTK, 2011b]. VTK wurde ursprünglich für C++ entwickelt und ist so eigentlich inkompatibel zu MultiCompare. Mit *ActiViz* steht jedoch ein von *Kitware* entwickelter offizieller Wrapper⁶ zur Verfügung, der die Funktionalität von VTK in *DLLs* (*Dynamic Link Librarys*) zusammenfasst [Act, 2011]. Um die Vielseitigkeit von VTK in einer *.NET*-Anwendung nutzen zu können, benötigt man ein *Windows-Control*, das eine *RenderWindow*-Komponente erzeugt. Diese Komponente wird ebenfalls als Teil des *ActiViz*-Wrappers zur Verfügung gestellt. Die Komponente führte bei der Implementation zu Problemen, da MultiCompare als 64-Bit Anwendung auch ein 64-Bit VTK benötigt und das zugrundeliegende *RenderWindowControl* ebenfalls dieser Architektur folgt. Da der *Visual Studio Interface Builder* aufgrund seiner Spezifikation jedoch nur 32-Bit Komponenten enthalten kann, muss das *RenderWindowControl* stets dynamisch zur Laufzeit mit all seinen Eigenschaften und Events generiert werden.

Ein weiteres Problem stellt der threadexklusive Zugriff von *.NET*-Komponenten auf die UI-Container dar, dessen Lösung u.A. im folgenden Abschnitt beschrieben wird.

²Objektorientierung: Sichtweise auf komplexe Systeme, bei der ein System durch das Zusammenspiel kooperierender Objekte beschrieben wird.

³C#: Von Microsoft im Rahmen der *.NET*-Strategie entwickelte Programmiersprache.

⁴.NET: Von Microsoft entwickelte Software-Plattform. Sie umfasst eine Laufzeitumgebung, eine für Programmierer bestimmte Sammlung von Klassenbibliotheken (API) und angeschlossene Dienstprogramme.

⁵Integrierte Entwicklungsumgebung (IDE von *integrated development environment*): eine Sammlung von Anwendungsprogrammen, mit denen die Aufgaben der Softwareentwicklung bearbeitet werden können, ohne weitere Anwendungen benutzen zu müssen.

⁶Wrapper: Software, die anderes Stück Software umgibt. Wrapper fungieren häufig als Schnittstelle zwischen unterschiedlichen Programmiersprachen.

5.3 Einbindung der Visualisierungen in MultiCompare

Die Lösungsvorschläge für die Umgehung der Threadexklusivität sind Gegenstand dieses Abschnittes. Weiterhin erfolgt eine Auflistung aller Probleme bezogen auf die Implementation, die in MultiCompare vorliegen.

5.3.1 Stand und Erweiterung der Software

MultiCompare entstand im Jahr 2006 zur Visualisierung von Toponomdaten, nachdem Matlab⁷-Programme aufgrund der Einschränkung der Sprache nicht mehr in der Lage waren, die komplexen Daten zu verarbeiten und dies nicht akzeptable Laufzeiten zur Folge hatte. Die Software wurde seitdem kontinuierlich erweitert und angepasst, um mit den aktuellen Messdaten sowie neuen Visualisierungstechniken umgehen zu können. Dementsprechend ist die Softwarearchitektur an vielen Stellen unübersichtlich und bedarf einer intensiven Einarbeitungszeit. 2010 wurde im Rahmen der Arbeit [Klemm, 2010] die interne Struktur der Software überarbeitet sowie alte Datenstrukturen in neue, effizientere *.NET*-Typen überführt.

Optimierung der Zugriffszeiten und Picking-Algorithmus

Die Zugriffszeiten auf einzelne Komponenten (wie CMPs) ist konstant zu halten, um eine schnelle Laufzeit auch bei komplexen Berechnungen zu gewährleisten. Dies wird hauptsächlich über Hashtabellen realisiert. Im *CMP-Storage*-Objekt werden beispielsweise alle CMPs eines Datensatzes kodiert mit ihrem *CMP-String* gespeichert. Wird ein CMP in der *CMP-Tabelle* ausgewählt, muss diese nur den *CMP-String* zurückgeben, damit mit linearem Aufwand alle Pixelpositionen des CMPs ermittelt werden können.

Dieses Konzept wurde bei der umgekehrten Adressierung von Pixeldaten zu CMP-Objekten in 2D- und 3D-Darstellungen aufgegriffen. Hierfür wurde ein 2D-/3D-Array erstellt, das an die Pixel/Voxel des Datensatzes gebunden ist. Jeder Eintrag repräsentiert den Wert des Datensatz an der entsprechenden Stelle. Der Eintrag des Arrays an einer Stelle entspricht dem *CMP-String* an der korrespondierenden Stelle im Datensatz. Wenn durch das Selektionstool eine Menge Pixelkoordinaten zurückgegeben werden, können die getroffenen Pixel anhand dieser Datenstruktur mit linearem Aufwand ermittelt, in der *CMP-Tabelle* lokalisiert und dargestellt werden.

Der Picking-Algorithmus selbst hat in der 3D-Ansicht zu verschiedenen Problemen geführt, die auf die Voxeldarstellung der Daten zurückzuführen sind. Die verwendete *vtkVolumePicker*-Klasse lieferte im praktischen Einsatz in der z-Achse stets eine zu hohe Schicht zurück, was auf die Ausdehnung des Voxel in diese Richtung zurückzuführen ist. Das Problem tritt unabhängig davon auf, ob man das Volumen von oben oder unten betrachtet. Eine Lösung hierfür wurde gefunden, indem die gewählte Schicht um

⁷Matlab: Programmiersprache und Software für numerische Berechnungen. Durch Plugins stark erweiterbar ist die Anwendung in der Forschung aufgrund des großen Funktionsumfang weit verbreitet.

einen entsprechenden Faktor korrigiert wird - Abhängig von der Ausrichtung des Volumens zur Kamera ist die ID der zurückgegebenen Schicht reduziert. Das Ergebnis ist eine korrekte Abbildung der Volumen- in Pixelkoordinaten.

Anbindung von VTK

Die Architektur von *.NET* erlaubt für die einzelnen Windows-Forms⁸ nur einen Threaddexklusiven Zugriff, wodurch auf das *RenderWindowControl* des 3D-Views nur innerhalb der Form-Klasse zugegriffen werden kann. Ein externer Zugriff, beispielsweise durch die *CMP-Tabelle*, ist jedoch nicht möglich. Eine solche Anpassung der Daten ist jedoch essentiell, wenn durch eine Änderung der Selektion der *CMP-Liste* sich die anzuzeigenden CMPs ändern.

Das Problem wurde gelöst, indem für verschiedene Operationen unterschiedliche Flags⁹ im Code gesetzt werden. Möglich wird dies durch die *Paint()*-Routine, die durch ein *Form* bei Änderungen der darzustellenden Komponenten aufgerufen wird. Diese Methode wird vom Thread des *Forms* ausgeführt, kann jedoch durch einen externen Thread durch die *Invalidate()*-Routine ausgelöst werden. Innerhalb der *Paint()*-Methode wird dann anhand von verschiedenen Flags ermittelt, welche Operationen vorgenommen werden sollen. Ist kein Flag gesetzt, wird am dargestellten Datensatz nichts geändert. Ist jedoch das Flag auf *repaintNeeded* gesetzt, wird das Volumen aus den aktuell gesetzten Parametern neu erzeugt und das *RenderWindow* entsprechend angepasst. Um die darzustellenden CMPs zu ändern, setzt die *CMP-Listen-Klasse* lediglich die erforderlichen Parameter für den 3D-View und führt anschließend dessen *Invalidate()*-Methode aus.

Ein weiteres Problem bei der Integration von VTK in MultiCompare trat bei der beleuchteten Darstellung der einzelnen 2D-Bilder als Volumen auf. Die jeweils erste und letzte Schicht des Datensatz wurde korrekt beleuchtet, während andere zu dunkel und damit farbverfälscht gerendert wurden. Der Fehler wurde behelfsmäßig umgangen, indem jeweils über- und unterhalb des Datensatzes eine durchsichtige Schicht hinzugefügt wurde und die globale Beleuchtung angehoben wurde, um die Farben originalgetreu darzustellen - da nun alle Voxel gleichmäßig dunkel angezeigt wurden. Wenn der Fehler in einer zukünftigen Versionen von VTK behoben wird, kann diese Hilfe wieder entfernt werden. Funktionell hat diese Lösung jedoch keine direkten Auswirkungen auf die Visualisierung, da die Farben durch die Anhebung der Beleuchtung originalgetreu dargestellt werden.

5.3.2 Probleme von MultiCompare

Als im Jahr 2006 mit der Entwicklung von MultiCompare begonnen wurde, hatten die Toponomdaten einen Umfang von etwa 22 gemessenen Epitopen bei 657×517 Bildpunkten mit 16 Bit Tiefe, wobei etwa 22.000 CMPs entstanden sind. Allerdings wurde auch

⁸Windows-Forms: Schnittstelle des *.NET*-Frameworks, die einen Zugriff auf die einzelnen Komponenten des Windows-Interface zur Verfügung stellt.

⁹Flag: Bezeichnung einer Variable als Hilfsmittel zur Kennzeichnung von Zuständen.

ein Datensatz mit ca. 100 verschiedenen Epitopen gemessen [Schubert et al., 2006]. Mittlerweile umfassen die Messungen über 100.000 CMPs bei 1.056×1.027 Bildpunkten mit 16 Bit Tiefe. Zum damaligen Zeitpunkt konnten mit MultiCompare-Software aufgrund der internen Architektur nicht alle CMPs gleichzeitig dargestellt werden, Berechnungen dauerten meist mehrere Tage. Dies wurde erst durch die Optimierung von Komponenten möglich [Klemm, 2010]. Nun wird die Software um die 3D-Komponente erweitert, was die Verarbeitung von noch umfangreicheren Datensätzen ermöglicht. Zu welchen Problemen dies in der Architektur von MultiCompare führt, wird nachfolgend beschrieben.

Speicherverbrauch durch interne Repräsentation der CMP-Daten

Ein großes Problem von MultiCompare ist der verschwenderische Umgang mit Arbeitsspeicher. Während in [Klemm, 2010] mit der `DataGridView`-Komponente, die zur Darstellung der CMP-Tabelle verwendet wird, durch die Verwendung einer alternativen Datenstruktur der Speicherverbrauch signifikant reduziert wurde, kommt es durch die interne Repräsentation des Datensatzes bei Verwendung von großen Toponomdaten zu einem massiven Speicherverbrauch. Der Grund hierfür ist die häufige Verwendung von auf Hash-Tabellen basierenden Datentypen sowie die redundante Speicherung von Daten. Der zum Zeitpunkt dieser Arbeit umfangreichste Datensatz umfasst 21 gemessene Epitope in 10 Schichten, die jeweils zwischen 5.000 und 36.000 CMPs enthalten. Werden alle Schichten in das Programm geladen, beträgt der Gesamtspeicherverbrauch 2.400 MB. Werden diese Daten zu einem 3D-Datensatz zusammengefügt, werden 5.925 MB Speicher allokiert. Es existieren hier sowohl die einzelnen als Schichten geladenen Daten, als auch die 3D-Repräsentation, da auf den Schichten noch weitere Operationen ausführbar sind. Während bei kleineren Datensätzen der Speicherverbrauch bei dieser Methode noch vertretbar ist, wird es bei zunehmend umfangreicheren Datensätzen zu Speicherproblemen kommen.

Um den Speicherverbrauch zu reduzieren, müsste die interne Repräsentation der CMPs, konkret die `SingleCMP`- und `CMPStorage`-Klasse überarbeitet werden, sodass mit Referenzen zwischen den einzelnen Objekten gearbeitet wird, anstatt die Daten bei den Operationen zu kopieren.

Überladung mit Werkzeugen

Um die Daten besser analysieren zu können, wurden immer weitere Tools ergänzt [Klemm, 2010]. Die sich teilweise überschneidende Funktionalität wirft Fragen bzgl. der Eignung früherer Designentscheidung für die Visualisierung auf - wie beispielsweise die Integration des 2D-Views in die CMP-Liste. Darüber hinaus muss geklärt werden, wie die einzelnen Werkzeuge besser miteinander kommunizieren können und wie sie auf dem Bildschirm angeordnet werden sollen.

Verschiedene Vorschläge zu diesem Thema erfolgen im Rahmen einer Diskussion über das User-Interface-Konzept der Visualisierungswerkzeuge.

5.4 User-Interface-Konzept

Für die Steuerung der neuen Werkzeuge waren bereits Rahmenbedingungen durch die vorhandene Software vorhanden. Darüber hinaus musste festgelegt werden, durch wen die Software zum Einsatz kommt, um so ggf. informatisches Fachwissen voraussetzen zu können. Neben einer möglichst intuitiven Bedienung der Werkzeuge ist die nahtlose Integration der neuen Tools in die Programmoberfläche ein wichtiger Aspekt.

Es ist aufgrund der großen Anzahl der verfügbaren Werkzeuge wichtig eine Redundanz der Information in den einzelnen Werkzeugfenstern zu schaffen. Stets alle verfügbaren Fenster geöffnet zu haben, nur weil bestimmte Informationen, wie z.B. das Proteinmuster des aktuell selektierten CMPs, nur in einzelnen Anzeigen zu finden sind, ist nicht zielführend.

Die Vielzahl von verfügbaren Informationsquellen führt zu Problemen der Anordnung der einzelnen Fenster auf dem Bildschirm. Fragen nach dem Platzanspruch der Anwendung sowie über die Anzahl der benötigten Monitore und die Anordnung der Werkzeuge werden abschließend in diesem Abschnitt behandelt.

5.4.1 Entwurf der Benutzeroberfläche

Bei Messungen der Arbeitsgruppe “Molekulare Mustererkennung“ wird die Software von einem Experten bedient, der auch für die Programmierung verantwortlich zeichnet. Es kann also ein umfassendes Fachwissen um die interne Funktionalität der Software vorausgesetzt werden. Es ist sinnvoll, die Parameter (z.B. zur Beeinflussung des Raycasting-Algorithmus) in die Nutzeroberfläche zu integrieren. Gleichzeitig werden Default-Werte für die Visualisierung so eingestellt, dass diese Parameter nicht verändert werden müssen, um eine gute Darstellung des Datensatzes zu erhalten. Lediglich bei Klärung von spezifischen Fragen wird es sinnvoll, diese zu verändern.

In diesem Teilabschnitt wird ein User-Interface-Konzept erarbeitet, um die vorgestellten 3D- und Graphvisualisierungen steuern zu können und gleichzeitig auf der bisherigen Nutzeroberfläche der Software aufbaut, um eine konsistente Bedienung der Software zu erlauben.

3D-View

Bei der Erarbeitung der Nutzerfläche wurde sich am bisher für 2D-Datensätze verwendeten *Fullscreen-View* orientiert, der eine vergleichbare Funktionalität bietet. Auch die Steuerung einzelner Funktionen in 3D weist beabsichtigte Ähnlichkeiten zu diesem Werkzeug auf. In Abbildung 5.1 ist die Nutzeroberfläche des 3D-Views dargestellt. Analog zum Fullscreen-View (vgl. [Klemm, 2010, S.37, Abb. 4.5]) sind fast alle Elemente in einem Container untergebracht, der sich rechts neben der 3D-Darstellung befindet. Die hier vorzufindenden Elemente dienen größtenteils der Steuerung von Funktionen, die im Abschnitt 4.4 vorgestellt wurden. Die einzelnen Funktionen wurden auf Interface-Komponenten abgebildet, wie sie in den ‘Windows User Experience Interaction Guidelines’ vorgeschlagen werden [Mic, 2011]. Slider werden verwendet, um Werte für die Opazität oder des aktuell dargestellten Slices einzustellen, während Checkbox-

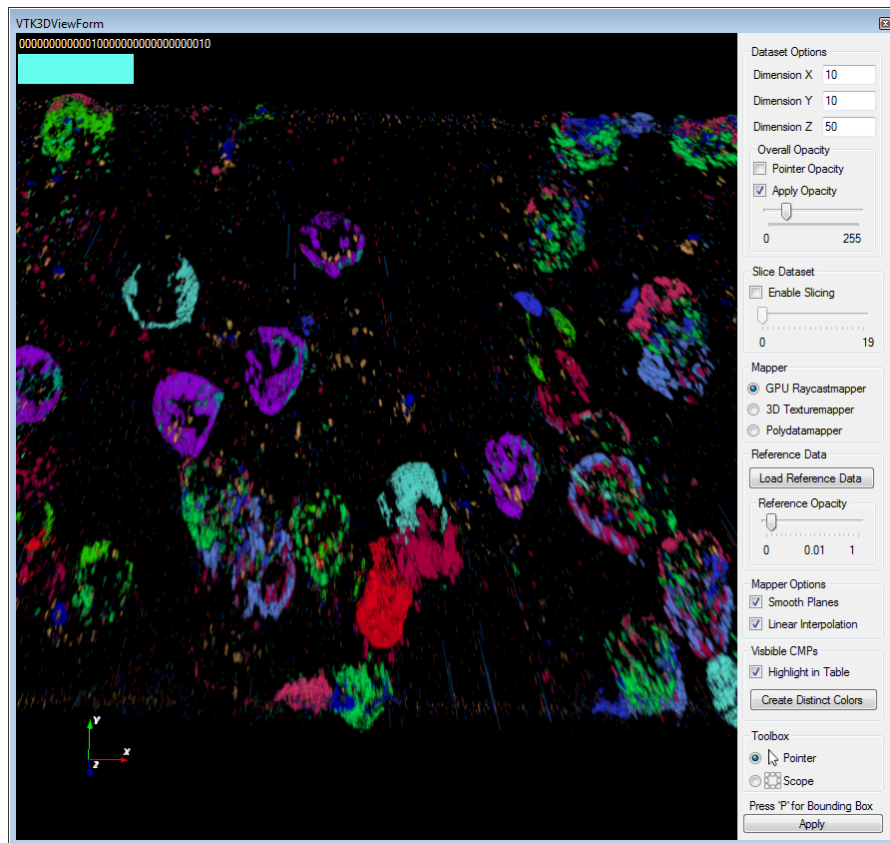


Abbildung 5.1: User Interface des 3D-View Tools. Alle für die Interaktion relevanten Komponenten befinden sich stets in einem Container rechts neben der 3D-Darstellung. Folgende Einstellung können hier getroffen werden: Abbildung der in x-, y- und z-Koordinate des Datensatz auf das Volumen (hier Maßstab 1:1:5); Opazität der einzelnen CMPs/des Gesamtvolumens; Durchslicen des Datensatzes; Anpassen des Mappers für die Darstellung des Datensatzes; Laden eines Referenzdatensatzes und Einstellen seiner Opazität; Optionen bezüglich des Mappers und Auswahl der Werkzeuge zur Interaktion mit dem Datensatz.

Komponenten bestimmte Einstellungen und Funktionen aktivieren/deaktivieren. Für die Auswahl der aktuellen Funktion des Cursors und des Mappers, werden Radiobuttons verwendet.

Muss das User-Interface für Nutzer ohne informatischen Hintergrund erweitert werden, wäre die Einführung eines Experten-Modus sinnvoll, der nur die Optionen enthält, für die ein detailliertes Wissen vorausgesetzt werden muss, während essentielle Komponenten, wie die Auswahl des Pointer-Werkzeugs, im normalen Modus angezeigt werden. Bislang gibt es jedoch für diese Unterteilung keinen Bedarf.

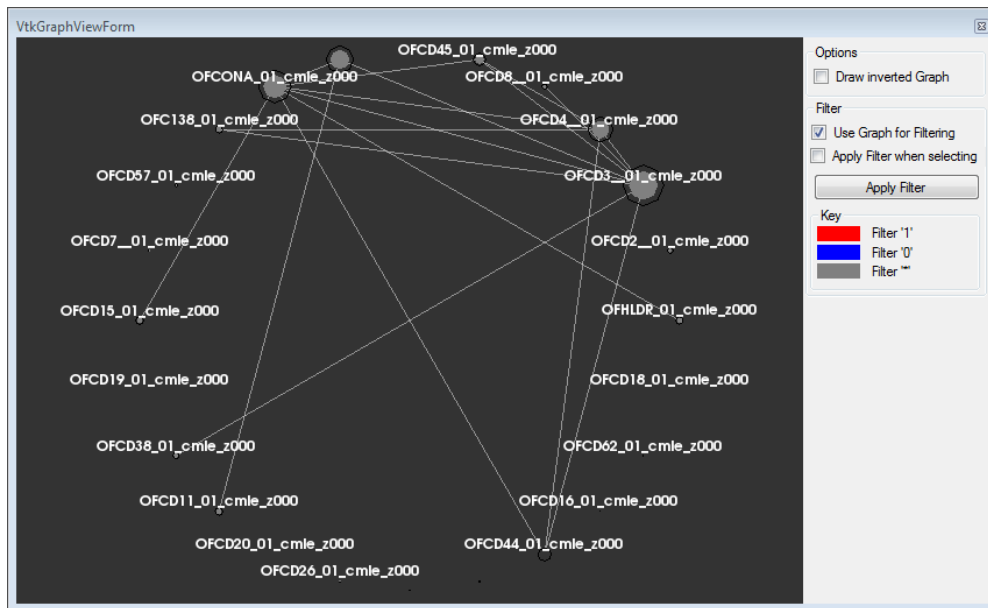


Abbildung 5.2: User Interface des Graph-View Tools. Analog zum Fullscreen-View und 3D-View befinden sich die für die Interaktion relevanten Komponenten in einem Container rechts zur Visualisierung. Da die Visualisierung an wenige Parameter gebunden ist, stehen lediglich Optionen für die Filterung des Graphen, zur Invertierung und eine Legende der Darstellung zur Verfügung.

Graph-View

Der Graph-View (dargestellt in Abbildung 5.2) folgt dem selben Interface-Design, wie der vorgestellte 3D-View. Auch hier nimmt die Visualisierung den Großteil des Werkzeugfensters ein, während die Parameter in einem Container am rechten Rand angepasst werden können. Da die Graphgenerierung automatisch erfolgt und die Darstellung einen unterstützenden Charakter hat, sind die Anzahl der Einstellungsmöglichkeiten im Vergleich zum 3D-View begrenzt. So hat der Nutzer die Auswahl, ob der Graph normal oder invertiert gezeichnet oder als Filtermuster verwendet werden soll (vgl. Abschnitt 4.4). Es wird ebenfalls eine Legende angezeigt, die aufschlüsselt, welche Farben welcher Kodierung entsprechen.

5.4.2 Interaktion zwischen den Tools

Da im Zuge dieser Arbeit zwei weitere Werkzeugfenster ergänzt wurden, liegt das Augenmerk darauf, wie die neuen und vorhandenen Werkzeuge besser ineinander greifen. Redundanz ist bezogen auf die Exploration des Datensatzes ein erwünschter Effekt.

Das Kernelement bei der Visualisierung des Toponoms ist die CMP-Tabelle, die stets zu sehen ist (mehr dazu im folgenden Abschnitt 5.4.3). Sie bildet die Schaltzentrale der Software, von hier aus werden alle weiteren Werkzeugfenster aufgerufen. Die Wahl der darzustellenden CMPs wird hier getroffen. Eine Änderung der Auswahl der selektierten CMPs hat fast immer eine Update-Routine aller Werkzeugfenster zur Folge, um die ausgewählten Netzwerke zu analysieren und Informationen zu extrahieren. Das Ziel ist nun, bestimmte Informationen, die nur einem einzelnen Werkzeugfenster zu entnehmen sind, in andere zu überführen. Erste Ansätze hierfür waren bereits in der Software vorhanden. Der Filter-View enthält eine Liste der Namen aller Epitope und versieht diese mit einem Drop-Down Menü, anhand dessen das aktuelle Filtermuster eingestellt wird (vgl. Abschnitt 4.5.3). Im Filter-View wurde anhand der Hintergrundfarbe der Epitopnamen kodiert, ob im aktuell ausgewählten CMP das Epitop anwesend (roter Hintergrund) oder abwesend (normaler Hintergrund) ist. So kann der Filter-View gleichzeitig verwendet werden, um bequem die Zusammensetzung eines CMPs zu erhalten, ohne den Umweg über die CMP-Liste zu gehen.

Eine vergleichbare Abbildung der Zusammensetzung des CMPs wurde im Graph-View vorgenommen. Ist der Filter-Modus nicht aktiviert, wird die Zusammensetzung des ausgewählten CMPs auf die Farbe der jeweiligen Knoten abgebildet. Ist das Epitop anwesend, wird der zugehörige Knoten rot dargestellt, bei Abwesenheit blau. Sind mehrere CMPs ausgewählt, wird immer der Binärcode des zuletzt aktivierten CMPs dargestellt. Gleichzeitig bietet der Graph-View die Möglichkeit zur Filterung der CMP-Tabelle nach einem bestimmten Binärmuster, wie in Abschnitt 4.5.3 beschrieben. Die Funktionalität des Filter-View bildet dabei eine Untermenge der Möglichkeiten im Graph-View, obwohl dieser für einen völlig anderen Zweck geschaffen wurde.

Die Selektionswerkzeuge, wie das Lassotool, die den Fullscreen-View in der Anwendung sehr beliebt gemacht haben, sind nun auch im 3D-View wieder zu finden. Darüber hinaus wurde, wie in Abschnitt 4.4.3 vorgestellt, die Hervorhebung von CMPs mittels des Cursors an die CMP-Tabelle gekoppelt. Da bisher immer durch die Selektion von CMPs in der Tabelle genau zugeordnet wurde, wo sich welche CMPs befinden, verfolgt dieser Ansatz den umgekehrten Weg - die Rückverfolgung des visualisierten CMPs zur CMP-Liste. Da die CMP-Liste in den meisten Fällen immer sichtbar ist, wird sie aktiv in die Zuordnung der aktuell dargestellten CMPs eingebunden.

Für zukünftige Erweiterungen der Software sollte überlegt werden, wie die CMP-Liste vom Funktionsumfang entschlackt werden kann. Neben der Listenansicht aller CMPs befindet sich in diesem Fenster auch eine 2D-Ansicht sowie die Auswahl des verwendeten Hintergrund-Messbildes. Die Integration dieser Visualisierung in das Fenster ergibt angesichts der steigenden Anzahl von spezialisierten Visualisierungswerkzeugen keinen Mehrwert mehr. Dies ist ein Ergebnis der agilen Entwicklungsstrategie, durch welche die Software immer neuen Anforderungen angepasst wurde. Eine weitere Modularisierung der Software sollte an dieser Stelle voran getrieben werden, ist jedoch aufgrund des Programmieraufwands ein sehr umfangreicher Arbeitsschritt, da der Hauptteil der Funktionalität der Visualisierung in der Komponente der CMP-Liste implementiert ist.

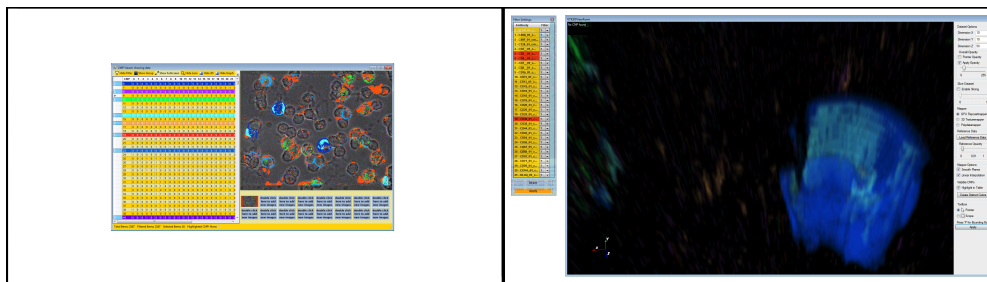


Abbildung 5.3: Anordnung der UI-Komponenten im Laboreinsatz der Arbeitsgruppe “Molekulare Mustererkennung”. Die CMP-Tabelle bildet auf dem Hauptmonitor den Dreh- und Angelpunkt bei der Visualisierung, während auf dem sekundären Monitor die 2D- oder 3D-Visualisierung in nahezu Vollbildskalierung zu sehen ist. Weitere Werkzeuge, wie das Filter- oder Gruppen-Fenster werden ebenfalls bei Bedarf auf dem sekundären Monitor dargestellt.

5.4.3 Anordnung der Werkzeuge und Multimonitorlösungen

Wie oben genannt ist die CMP-Liste bildet das Kernelement der Visualisierung, sie muss stets sichtbar sein. Jedoch ist es nicht nötig, dass sie im Zentrum der Aufmerksamkeit des Nutzers steht. Sobald ein Visualisierungstool wie der Fullscreen- und 3D-View geöffnet wird, muss eine möglichst sinnvolle Anordnung der Werkzeuge gefunden werden, sodass sich die Programmfenster nicht überdecken.

Die aktuell in der Arbeitsgruppe “Molekulare Mustererkennung” in Magdeburg verwendete Anordnung beruht darauf, dass entweder der 2D- oder 3D-View auf dem sekundären Monitor in Vollbild zu sehen ist, während auf dem primären Monitor nur die CMP-Tabelle angezeigt wird (Abbildung 5.3). Weitere Werkzeuge, wie der Filter-View, werden nach Bedarf mit auf dem sekundären Monitor angezeigt. Bei einer Analyse des praktischen Einsatzes der Software zeigte sich ebenfalls, dass bestimmte Funktionen, wie das Gruppieren einzelner CMPs, nicht mehr häufig genutzt wird und dass das entsprechende Tool standardmäßig zunächst ausgeblendet werden kann, wenn ein Datensatz angezeigt wird. Da bisher die Graphvisualisierung noch nicht eingesetzt wurde, war ein Vorschlag nötig, wie dieser integriert werden kann (Abbildung 5.4). Das CMP-Form in seiner jetzigen Form kann jedoch nicht in seiner Größe angepasst werden kann, sodass der Graph-View nicht genügend Platz auf dem Bildschirm findet. Eine Lösung hierfür ist die Verdeckung der Schaltflächen für das Hinzufügen von Fluoreszenzbildern im 2D, da diese in der 3D-Visualisierung nicht genutzt werden. Alternativ könnte der Graph-View zusammen mit der 3D-Visualisierung auf dem sekundären Monitor platziert werden, wobei eine Vollbildansicht von letzterem nicht mehr möglich ist. Das Filter-Tool kann hierbei auf dem primären Monitor platziert werden.

Auf lange Sicht hin sollte die 2D-Visualisierung von der CMP-Tabelle gelöst werden, damit eine überdeckungsfreie Anordnung der Werkzeugfenster möglich ist.

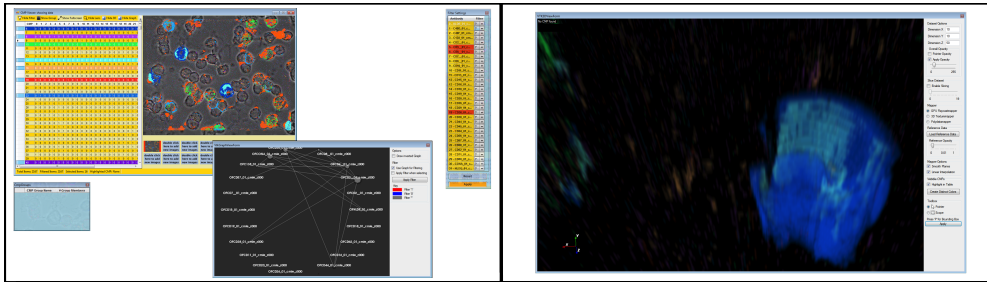


Abbildung 5.4: Vorschlag für Anordnung der UI-Komponenten mit aktiviertem Graph-View-Werkzeug. Die im 3D nicht verwendeten Schaltflächen für das Hinzufügen von Fluoreszenzbildern als Render-Grundlage wird nicht benötigt und durch den Graph-View verdeckt.

5.5 Zusammenfassung

Das Ziel dieses Kapitels war es, einen Überblick über die verwendeten Werkzeuge zu geben sowie Lösungsansätze für die Probleme bei der Integration der Bibliothek *VTK* in eine 64-Bit *.NET*-Anwendung, wie *MultiCompare*, aufzuzeigen. Für die weiteren Arbeiten an diesem Programm müssen die herausgearbeiteten Probleme beachtet werden, um auch mit zukünftigen Datensätzen die Leistungsfähigkeit der Visualisierungen nutzen zu können. Bei der Entwicklung des User-Interfaces wurde ein besonderes Augenmerk auf die einheitliche Funktionalität der bereits vorhandenen Werkzeuge gelegt, was bestimmte Vorgaben bei der Erstellung der Konzepte mit sich brachte - ebenso wie der Umstand, dass die Software von einem Experten bedient wird. Wie die Werkzeuge sinnvoll miteinander kommunizieren und auf dem Bildschirm angeordnet werden, wurde abschließend diskutiert.

6 Zusammenfassung und Ausblick

Das Ziel dieser Arbeit war die Entwicklung einer geeigneten Visualisierung für dreidimensionale Toponomdatensätze unter Beachtung der bisherigen Methodik. Darüber hinaus sollte eine über die Darstellung von Toponomnetzwerken als Graph eine zusätzliche Informationsquelle bei der Analyse zur Verfügung gestellt werden, wobei statistische Merkmale - an visuelle Parameter gekoppelt - ablesbar sein sollten. Die für diesen Zweck erstellten Konzepte sollten in die Software MultiCompare integriert werden, die bisher für die Visualisierung von 2D-Toponomdaten Verwendung fand. Die interaktive Komponente der Visualisierung sollte sich hierbei an den bereits etablierten Konzepten innerhalb der Software orientieren, sodass eine homogene Bedienbarkeit gewährleistet wird.

Hierzu wurden zunächst die biomedizinischen Vorraussetzungen erörtert, wobei geklärt wurde, was unter dem Toponom zu verstehen ist und warum das Wissen um dessen Beschaffenheit essentiell für die Diagnose und Behandlung von verschiedensten Krankheiten ist. Gleichzeitig ist das bildgebende Verfahren der auf Fluoreszenzmikroskopie basierenden MELK/TIS-Technologie erläutert worden, mit dem Teile des Toponoms extrahiert werden können.

Im Grundlagenkapitel wurden informatische Themengebiete und deren aktueller Forschungsstand abgesteckt, die relevant für den Entwurf der verschiedenen eingesetzten Visualisierungen sind. Hier lag der Fokus zunächst auf der direkten Volumenvisualisierung sowie der Interaktion im 3D-Raum, um die dargestellten Daten effizient explorieren zu können. Wie ein Graph definiert ist und wie diese Datenstruktur bereits genutzt wird, um Proteinnetze abzubilden, wurde danach beleuchtet, bevor abschließend verschiedene Farbräume, Maße und Vorgehensweisen zur Generierung perzeptuell unterscheidbarer Farben ermittelt wurden. Dies ist eine Folge der farbkodierten Darstellung von vielen verschiedenen funktionellen Clustern bei der späteren Visualisierung.

Das Fundament der Arbeit bildet die Anforderungen, die zusammen mit der Arbeitsgruppe "Molekulare Mustererkennung" in Magdeburg erarbeitet wurden, die für die Entwicklung der MELK/TIS-Technologie und die Visualisierung der Daten verantwortlich zeichnen. Die extrahierten Rahmenbedingungen stellen klar, welche Funktionen von der 3D- und Graphvisualisierung abgedeckt werden müssen. Da im Lauf der Arbeit weitere Problemfelder erkundet wurden, sind diese Anforderungen um Aspekte der farblichen Kodierung von Proteinclustern und der Berücksichtigung der kognitiven Last des Nutzers erweitert worden.

Auf der Basis von Untersuchungen der Arbeitsabläufe bei der Analyse der Toponom-

daten im Laboralltag wurden Konzepte und Algorithmen für die Visualisierung von Toponomdaten in 3D erstellt, die auf der bisherigen 2D-Darstellung aufbauen. Hierbei musste auf neue Herausforderungen Rücksicht genommen werden, die eine Folge der zusätzlich dargestellten Dimension auf einem 2D-Ausgabegerät sind, wie die Navigation im 3D-Raum und Verdeckungen von Mustern. In diesem Rahmen wurden verschiedene Ideen bezüglich transparenter Darstellung und Kontextinformationen mittels des Mauszeigers vorgestellt.

Die Graphvisualisierung greift Ideen von [Oeltze et al., 2011] auf und überführt sie in eine alternative Darstellung des Graphen, die neben statistischen Merkmalen auch mit der entwickelten 3D-Komponente interaktiv zusammenarbeitet und so etwa die Struktur selektierter Netzwerke abbildet. Es werden ebenso Operationen auf Grundlage des Datensatzes möglich gemacht - wie das Filtern nach bestimmten Epitopen.

Den Abschluss des Entwurfskapitels bildete die Entwicklung eines Algorithmus, der eine Menge von perzeptuell unterscheidbarer Farben auf Basis von deren Anzahl generiert. Er wurde für die 3D-Visualisierung implementiert, um eine bessere Unterscheidbarkeit von molekularen Strukturen zu gewährleisten. Gleichzeitig wurde sich in diesem Rahmen mit Themen wie der dynamischen Farbspreizung der angezeigten Proteinnetzwerke und deren Auswirkung auf die kognitive Last des Nutzers befasst.

Implementationsrelevante Details, wie gewählte Programmiersprache und Grafikbibliotheken sowie bereits existierende Vorgaben durch die existierende Software MultiCompare, wurden im nachfolgenden Kapitel thematisiert. Hierbei wurden ausgewählte Probleme herausgegriffen, die einzelne Inkompatibilitäten und deren Umgehung adressieren. Diese beziehen sich zum Großteil auf die Integration der vorgestellten Lösungen in MultiCompare. In diesem Zusammenhang wurde auch ein mit bisherigen Konzepten der Software konform gehendes User-Interface-Konzept erstellt, das neben Anordnung von einzelnen Komponenten auch versucht, eine bessere Kommunikation der einzelnen Visualisierungswerkzeuge untereinander zu gewährleisten. Im Rahmen der Benutzbarkeit wurden auch verschiedene Anordnungen der einzelnen Programmfenster vorgeschlagen, welche die Handhabung der Software vereinfachen und den Informationsfluss beschleunigen.

Nachfolgend soll eine Bewertung der Ergebnisse der Arbeit aus biologischer und informatischer Sicht vorgenommen werden. Welche Auswirkungen diese neuen Möglichkeiten für die Erforschung des Toponoms mit sich bringen, wird danach erläutert, bevor abschließend zukünftige Problemfelder abgesteckt werden - wobei sowohl kurzfristige Aufgaben, als auch noch weit entfernte Ziele gleichermaßen in den Fokus rücken.

6.1 Bewertung der Ergebnisse

Durch die im Rahmen dieser Arbeit geschaffenen Visualisierungen und deren Implementation in die Software MultiCompare ist es möglich, auf einer interaktiven Basis die gemessenen 3D-Toponomdaten zu analysieren, wobei die Arbeitsabläufe bei der Handhabung der Software auf bewährten, bereits in der Praxis angewandten Konzepten be-

ruhen. Für die Erforschung von funktionellen Zusammenhängen ist nicht nur die reine Anzeige von verschiedenen molekularen Netzwerken gekoppelt an ihren topologischen Ursprung wichtig - für den Explorationsprozess ist es ebenso von großer Bedeutung, dass durch die interaktive Komponente bestimmte Muster an- und abgewählt werden können, sodass unterschiedliche Kombinationen von Geweben in schnellen Abfolgen gezeigt werden können.

Für die Analyse der 3D-Daten können dieselben Werkzeuge eingesetzt werden, die auch schon für 2D-Messungen Verwendung fanden, da die Visualisierung vollständig in das Programm integriert wurde. Darüber hinaus wurden weitere Möglichkeiten geschaffen, mit Verdeckungen und anderen Effekten in der 3D-Darstellung umzugehen. Da das eigentliche Problem bei der Analyse des Toponoms die Analysierbarkeit der Daten aufgrund der hohen Komplexität ist, sind gleiche Konzepte bei der Bedienung unterschiedlicher Werkzeuge von essentieller Bedeutung. An dieser Stelle setzt die Graphvisualisierung an, die erstmals eine Darstellung der ausgewählten Netzwerke losgelöst von deren Lage im Gewebe anwendet, um die Proteinstruktur zu veranschaulichen. Diese unterstützende Darstellung bietet als weitere Informationsquelle zusätzliche Einblicke in sowohl 2D- als auch 3D-Daten.

Durch den verbesserten Farbalgorithmus sowie das Farbspreiztool wird die Gesamtqualität der 2D- und 3D-Visualisierung angehoben, da nun Farbtöne basierend auf dem gesamten vom Menschen wahrnehmbaren Farbspektrum gewählt werden und perzeptuelle Abstände als Unterscheidungsfaktor dienen. Über die Farbspreizung wird die Unterscheidbarkeit der Netzwerke verbessert, sodass weniger Interaktionsschritte nötig sind, um den Datenpunkten ihr korrektes Proteinmuster zuzuordnen.

Ein weiterer wichtiger Punkt, dem zukünftig weitere Beachtung zukommen wird, ist die Verzahnung der einzelnen Werkzeuge untereinander mit dem Ziel, die kognitive Last des Nutzers zu reduzieren. Der Interaktionsaufwand für die Erlangung einer Interaktion wird so auf ein Minimum reduziert, damit das Hauptaugenmerk auf dem primären Ziel liegt - dem Verstehen der hochkomplexen zellulären Proteinstrukturen.

Darüber hinaus haben die implementierten Visualisierungen noch einige Limitationen. Eine dynamische Farbspreizung aller sichtbarer CMPs ist nicht möglich, da der Nutzer schnell die Übersicht verlieren würde. In Abschnitt 6.3 wird ein Lösungsvorschlag für dieses Problem über die Annotation von CMPs gemacht. Ähnlich verhält es sich mit der Transparenz, die zwar das Problem der Verdeckung von Strukturen zu Teilen löst, jedoch durch die Farbverfälschung die Assoziation zwischen 3D-Visualisierung und CMP-Liste erschwert.

Das Lasso-Tool erlaubt bisher nur die Selektion von CMPs in einem Rechteck. Hier wäre eine Freihand-Selektion flexibler. Diese ermöglicht einen genaueren Vergleich von CMP-Strukturen. Hierbei ist der vorgestellte Vergleichsgraph sinnvoll, der jedoch noch nicht Teil der Visualisierung ist.

6.2 Folgen für die Forschung

Mit der echtzeitnahen Visualisierung von 3D-Toponomdaten ist die Analyse komplexer zellulärer Strukturen möglich. Während in der 2D-Ansicht hauptsächlich Zelloberflächen detailliert untersucht wurden, ist es nun möglich, den Austausch von Mo-

lekülen zwischen großen zellulären Systemen zu analysieren. Im 3D-Raum sind die tatsächlichen zellulären Strukturen besser erkennbar, da die einzelnen Zellstrukturen - abhängig von der Schichtdicke - detailliert dargestellt werden.

Durch die vorgestellten Visualisierungen sind erstmals im 3D-Raum Proteincluster interaktiv darstellbar. Dies ermöglicht zusätzlich eine quantitative Analyse des Toponomraumes, wodurch Muster direkt miteinander verglichen werden können. Eine Rekonstruktion von Zeitabläufen innerhalb von Zellen oder ganzen Gewebestrukturen ist denkbar. So können verschiedene Zustände miteinander verglichen werden. Mit dieser Arbeit wurde die Grundlage für den Vergleich von Mustern in unterschiedlichen Datensätzen geschaffen, um somit in 3D verschiedene Zellvorgänge nachvollziehen zu können.

Zeitersparnis durch 3D-Visualisierung

Zusammen mit dem neu geschaffenen 3D-View geht eine große Zeitersparnis einher, eine 3D-Darstellung von Toponomdaten zu erhalten. Bisher war ein umfangreicher Prozess der Vorbereitung der Daten erforderlich, um nur einen Ausschnitt der gewonnenen Daten zu visualisieren. Zu diesem Zweck wurde für die 30 häufigsten Gewebearten jeweils ein Bild in jeder Schicht manuell erstellt (was bei 10 Schichten 300 per Hand zu speichernde Bilder ergab). Diese wurden anschließend jeweils im Rot- Grün- und Blaukanal binarisiert und danach durch die externe Software *Bitplane Imaris* [Ima, 2011] eingelesen. Da *Imaris* bei einer hohen Anzahl von Kanälen instabil wurde, konnten nie mehr als 30 Netzwerke dargestellt werden. Der Aufwand für die Erstellung der Daten belief sich auf 1-2 Arbeitstage.

Durch die nun vorhandene Visualisierung kann eine beliebige Zusammenstellung von Strukturen innerhalb von Sekunden dargestellt werden. Das Laden eines 3D-Datensatzes benötigt je nach Anzahl der Schichten und der vorhandenen Rechenleistung zwischen 20 und 40 Sekunden. Hierbei wurde die Zeitersparnis durch die bessere Analysierbarkeit der Daten mittels eigens dafür geschriebenen Tools noch nicht einberechnet.

6.3 Zukünftige Problemfelder

Die zukünftige Arbeit bezüglich des informatischen Aspekts der Toponomforschung umfasst viele unterschiedliche kurz- und langfristige Gesichtspunkte, die über die reine Visualisierung der Daten weit hinausgehen.

Steigender Umfang der Datensätze

Mit der stetigen Weiterentwicklung der MELK/TIS-Technologie sehen sich die Visualisierungskomponenten wachsenden Anforderungen gegenüber, denen durch intelligentes skalierendes Software-Engineering entsprochen werden muss. Konkret lassen diese sich in drei Aspekte unterteilen:

- Die Auflösung der Kamera des MELK/TIS-Roboters wird steigen und somit größere Messbilder erzeugen.
- Die Anzahl der gemessenen Epitope wird steigen.
- Es werden für 3D-Messungen mehr Schichten berücksichtigt.

Im allgemeinen adressieren diese Herausforderungen zunächst bereits bekannte Probleme, wie den massiven Speicherverbrauch bei sehr umfangreichen Datensätzen. Dieser kann zwar durch die Aufrüstung der Arbeitsspeichers abgefangen werden, jedoch ist in der Softwarearchitektur ebenfalls Potential vorhanden, diesen Verbrauch zu reduzieren.

Hierbei ist zur Diskussion zu stellen, ob manche Visualisierungstools für die wachsende Anzahl von molekularen Netzwerken noch geeignet sind, da die Abbildung der Proteinkomplexe auf Farben ab einer bestimmten Anzahl nicht mehr sinnvoll erscheint. Ebenso verhält es sich mit der Listenansicht. In diesem Rahmen sollte geprüft werden, ob bestimmte früher getroffene Designentscheidungen noch tragbar sind, wie die Einbindung des 2D-Views in die Listenansicht. Weitere Werkzeuge wie die Linse wurden durch die neu geschaffenen Visualisierungen bereits obsolet gemacht.

Neue, intelligentere Visualisierungstechniken für die Abbildung der Toponomdaten sind an dieser Stelle ein vielversprechendes Forschungsfeld.

Erweiterung der vorhandenen Visualisierungen

In der Arbeit wurden eine Reihe von Vorschlägen für künftige Erweiterungen der vorgestellten Visualisierungen gemacht, die an dieser Stelle noch einmal zusammengefasst und erweitert werden sollen.

- **Annotationsfunktion für Strukturen in der 3D-Visualisierung**

Bezüglich der Visualisierung der Proteinnetzwerke als farbkodierte Datenpunkte im 3D-Raum wurde eine Annotationsfunktion vorgeschlagen, die dem Nutzer hilft, einen Überblick über die dargestellten Strukturen zu wahren. Durch bisherige Implementationen ist es zwar möglich, durch den Cursor den einzelnen CMPs schnell die Proteinstruktur zuzuordnen. Jedoch wäre es nützlicher, diese Information über eine Annotationsfunktion dauerhaft einzublenden, um den kognitiven Workload des Nutzers zu verringern. Durch eine solche Funktion wäre auch der Einsatz einer dynamischen Farbspreizung, abhängig vom dargestellten Ausschnitt, denkbar.

- **Farbabstände von benachbarten Strukturen berücksichtigen**

Eine bereits erwähnte Optimierung des Farbverteilungsalgorithmus sollte die Einfärbung der Proteinnetzwerke an deren benachbarte Strukturen koppeln. Ein optimales Ergebnis wäre ein möglichst großer perzeptueller Abstand der Farben von jeweils benachbarten Komplexen, um zu verhindern, dass aneinandergrenzende Netzwerke auf ähnliche Farben gemappt werden.

- **Vergleich von verschiedenen Strukturen durch Graphvisualisierung**

Die Graphvisualisierung bietet viel Potential für die Vergleichbarkeit des Aufbaus von unterschiedlichen Netzwerken (Abschnitt 4.5.4). Die Möglichkeiten, die Graphdarstellung an die 2D- und 3D-Visualisierung zu koppeln, wäre ein lohnenswertes Ziel weiterer Forschungen in diesem Gebiet.

Darüber hinaus sind neue Ansätze für Visualisierungen sowohl für die Darstellung der CMPs als Liste und als farbkodierte Datenpunkte sinnvoll. Gerade letzteres stößt oft aufgrund der großen Anzahl gleichzeitig dargestellter Netzwerke in Ermangelung ausreichend perzeptuell unterscheidbarer Farben an seine Grenzen. Ähnlich wie bei der Graphvisualisierung sind abstraktere Formen der Datenabbildung denkbar, die neue Herangehensweisen und Einsichten in die Daten ermöglichen.

Algorithmische Krankheitserkennung

Das große Ziel, das sowohl Visualisierung als auch Bildverarbeitung, Mustererkennung und viele weitere informatische Disziplinen vereint, ist die computergestützte Diagnostik anhand von Toponomdaten. Voraussetzung hierfür ist eine Beschreibung, wie pathologisches Gewebe für eine bestimmte Erkrankung charakterisiert ist - die Definition eines molekularen "Gesichts". Bis dahin sind jedoch weitere Unterstützungen auf diesem Gebiet denkbar, die zwar keine automatische Krankheitserkennung ermöglichen, jedoch interessante Strukturen hervorheben.

- **Detektieren von Zellen mit ähnlichen molekularen Mustern**

Anhand der Anordnung der molekularen Muster könnte Gewebsausschnitte ermittelt werden, die eine ähnliche Proteinstuktur aufweisen, um so Vergleiche der unterschiedliche Zustände dieser Apparate anstellen zu können. Hierbei sind auch Ähnlichkeitsmaße denkbar sowie die Extraktion von Mustern, die nur in wenigen Ausschnitten vorkommen.

- **Automatischer Vergleich von Unterschieden zwischen Datensätzen**

Die Visualisierung von Unterschieden zwischen verschiedenen Datensätzen würde einen Rückschluss auf die verschiedenen Zustände erlauben, die von den einzelnen Datensätzen observiert wurden. Subtraktionsdatensätze werden von MultiCompare für 2D-Daten bereits unterstützt.

- **Detektion von abnormen hoch pathogenen Zellformen**

Der Vergleich von verschiedenen Zellapparaten, wie oben angedeutet, könnte ebenso dazu verwendet werden, derartige Zellen, mittels ihrer besonderen 3D-Struktur zu erkennen.

Wie diesem Abschnitt zu entnehmen, liegen die zukünftigen Bemühungen nicht mehr nur in der reinen Visualisierung der Daten, sondern auch der Erkennung von ungewöhnlichen Mustern. Das von der Toponomforschung abgedeckte informatische Spektrum wird daher in naher Zukunft stets immer weiter aufgefächert werden.

6.4 Schlusswort

Mit der Erschließung der dritten Dimension gewinnt die noch junge Toponomforschung neben einer noch höheren Komplexität die Möglichkeit, Zellstrukturen als Volumen darzustellen. Neben einem Zugewinn an Informationen für die Detektion pathologisch relevanter Proteinstrukturen steht nun die Tür für weitere computergestützte Analysen offen, wie der Erkennung von verdächtigen Mustern anhand von Vergleichsalgorithmen. Trotz der enormen biologischen und informatischen Fortschritte in diesem Gebiet innerhalb der vergangenen Jahre ist die vollautomatische Detektion von pathologischen Strukturen ein entferntes Ziel, das stets neue wissenschaftliche Disziplinen mit einbeziehen wird.

Die in dieser Arbeit vorgestellten neuen Techniken zur Darstellung von 3D-Toponomdaten als auch die abstrakte Abbildung der gemessenen Proteinstrukturen auf einen Graphen sind ein weiterer Schritt auf diesem Weg und dienen hauptsächlich der Erschließung biologischer Erkenntnisse durch Spezialisten. Dieses Wissen bildet wiederum die Grundlage für weitere, neue Ansätze, sowohl in der Visualisierung der Daten als auch der computergestützten Mustererkennung. Die künftige Arbeit im informatischen Aspekt sollte sowohl auf den bisherigen Möglichkeiten aufbauend erweiterte Darstellungs- und Interaktionsmöglichkeiten bieten als auch durch alternative Visualisierungen neue Türen öffnen.

Literaturverzeichnis

- [Col 2011] *ColorBrewer.org Webseite von Harrower et al.* www.colorbrewer.org.
Version: November 2011
- [Act 2011] *Offizielle Webseite des ActiViz-Wrapper.* www.kitware.com/products/activiz.html. Version: November 2011
- [MON 2011] *Offizielle Webseite des MONO Projekts.* www.mono-project.com.
Version: November 2011
- [VTK 2011a] *Offizielle Webseite des Visualization Toolkits.* www.vtk.org.
Version: November 2011
- [VTK 2011b] *Offizelles VTK-Wiki von Kitware.* www.vtk.org/Wiki/VTK.
Version: Dezember 2011
- [Ima 2011] *Offizielle Webseite von Bitplane Imaris.* www.bitplane.com/go/products/imaris. Version: November 2011
- [CIE 2011] *Webseite der Commission Internationale de l'Eclairage.* www.cie.co.at.
Version: November 2011
- [Flu 2011] *Wikipedia Artikel 'Fluoreszenzmikroskopie'.* de.wikipedia.org/wiki/Fluoreszenzmikroskopie. Version: Dezember 2011
- [HSV 2011] *Wikipedia Artikel 'HSV-Farbraum'.* de.wikipedia.org/wiki/HSV-Farbraum. Version: November 2011
- [Ray 2011] *Wikipedia Artikel 'Volume ray casting'.* en.wikipedia.org/wiki/Volume_ray_casting. Version: November 2011
- [Vox 2011] *Wikipedia Artikel 'Volumengrafik'.* de.wikipedia.org/wiki/Volumengrafik. Version: November 2011
- [Mic 2011] *Windows User Experience Interaction Guidelines.* msdn.microsoft.com/en-us/library/aa511258.aspx. Version: November 2011
- [Bade et al. 2005] BADE, Ragnar ; RITTER, Felix ; PREIM, Bernhard: Usability comparison of mouse-based interaction techniques for predictable 3d rotation. In: *Smart Graphics* (2005)
- [Bertel 2008] BERTEL, François: GPU Ray Casting in VTKEDGE. In: *Kitware Quarterly* (2008)

- [Boyd et al. 1998] BOYD, Dana ; SCHIERLE, Clark ; BECKWITH, Jon: How many membrane proteins are there? In: *Protein science* 7 (1998), Nr. 1, S. 201–205
- [Cohen-Or et al. 2006] COHEN-OR, Daniel ; SORKINE, Olga ; GAL, Ran ; LEYVAND, Tommer ; XU, Ying-Qing: Color harmonization. In: *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2006* 25 (2006), Nr. 3
- [Conner et al. 1992] CONNER, Brookshire D. ; SNIBBE, Scott S. ; HERNDON, Kenneth P. ; ROBBINS, Daniel C. ; ZELEZNIK, Robert C. ; DAM, Andries van: Three-dimensional widgets. In: *I3D '92 Proceedings of the 1992 symposium on Interactive 3D graphics*, 1992
- [Di Battista et al. 1998] DI BATTISTA, Giuseppe ; TOLLIS, Ioannis G. ; EADES, Peter ; TAMASSIA, Roberto: *Graph Drawing: Algorithms for the Visualization of Graphs*. 1. Prentice Hall, 1998
- [Diestel 2010] DIESTEL, Reinhard: *Graphentheorie*. 4. Aufl. Springer, 2010
- [de Duve 1989] DUVE, Christian de: *Die Zelle: Expedition in die Grundstruktur des Lebens*. Spektrum Akademischer Verlag, 1989. – 455 S.
- [Friedenberger et al. 2007] FRIEDENBERGER, Manuela ; BODE, Marcus ; KRUSCHE, Andreas ; SCHUBERT, Walter: Fluorescence detection of protein clusters in individual cells and tissue sections by using toponome imaging system: sample preparation and measuring procedures. In: *Nature protocols* 2 (2007), Nr. 9, S. 2285–2294
- [Gansner und Koren 2007] GANSNER, Emden R. ; KOREN, Yehuda: Improved circular layouts. In: *Graph Drawing* (2007)
- [Hand 1997] HAND, Chris: A survey of 3D interaction techniques. In: *Computer Graphics Forum* (1997)
- [Harrower und Brewer 2003] HARROWER, Mark ; BREWER, Cynthia A.: ColorBrewer.org: An Online Tool for Selecting Colour Schemes for Maps. In: *The Cartographic Journal* 40 (2003), Juni, Nr. 1, S. 27–37
- [Healey 1996] HEALEY, Christopher G.: Choosing effective colours for data visualization, 1996, S. 263–270
- [Houde 1992] HOUDE, Stephanie: Iterative design of an interface for easy 3-D direct manipulation. In: *CHI '92 Proceedings of the SIGCHI conference on Human factors in computing systems*, 1992
- [Itten 1961] ITTEN, Johannes: *Kunst der Farbe*. Ravensburg, 1961
- [Klemm 2010] KLEMM, Paul: Echtzeitnahe Visualisierung von molekularen Netzwerken (Toponome) in Geweben und Zellen, 2010
- [Krüger und Westermann 2003] KRÜGER, J. ; WESTERMANN, R.: Acceleration techniques for GPU-based volume rendering. In: *Visualization, 2003. VIS 2003. IEEE* (2003), S. 287–292

- [Kulik 2009] KULIK, Alexander: Building on realism and magic for designing 3D interaction techniques. In: *Computer Graphics and Applications* (2009)
- [Lichtman und Conchello 2005] LICHTMAN, Jeff W. ; CONCHELLO, José-Angel: Fluorescence microscopy. In: *Nature methods* 2 (2005), Dezember, Nr. 12, S. 910–919
- [Liu und Layland 1973] LIU, C. L. ; LAYLAND, James: Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment. In: *Journal of the ACM (JACM)* 20 (1973), Januar, Nr. 1
- [Martinet et al. 2010] MARTINET, Anthony ; CASIEZ, Géry ; GRISONI, Laurent: The design and evaluation of 3D positioning techniques for multi-touch displays. In: *2010 IEEE Symposium on 3D User Interfaces (3DUI)* (2010), S. 115–118
- [Mine et al. 1997] MINE, Mark R. ; BROOKS JR, Frederick P. ; SQUIN, Carlo H.: Moving objects in space: exploiting proprioception in virtual-environment interaction. In: *SIGGRAPH '97 Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, 1997
- [Oeltze et al. 2011] OELTZE, Steffen ; FREILER, Wolfgang ; HILLERT, Reyk ; DOLEISCH, Helmut ; PREIM, Bernhard ; SCHUBERT, Walter: Interactive, Graph-Based Visual Analysis of High-Dimensional, Multi-Parameter Fluorescence Microscopy Data in Toponomics. In: *IEEE Transactions on Visualization and Computer Graphics (in press)* (2011), S. 1–10
- [Preim und Bartz 2007] PREIM, Bernhard ; BARTZ, Dirk: *Visualization in medicine*. Morgan Kaufmann, 2007 (theory, algorithms, and applications)
- [Preim und Ritter 2002] PREIM, Bernhard ; RITTER, Felix: Techniken zur interaktiven Hervorhebung von Objekten in medizinischen 3D-Visualisierungen. In: *Simulation und Visualisierung 2002* (2002)
- [Rezk-Salama 2004] REZK-SALAMA, Christoph: Volume Rendering Techniques for General Purpose Graphics Hardware. (2004)
- [Schubert 1990] SCHUBERT, Walter: Multiple antigen-mapping microscopy of human tissue. In: *Excerpta Medica. Advances in Analytical Cellular Pathology* (1990), Nr. 9
- [Schubert 2003] SCHUBERT, Walter: Topological proteomics, toponomics, MELK-technology. In: *Advances in biochemical engineering/biotechnology* 83 (2003), S. 189–209
- [Schubert 2004] SCHUBERT, Walter: Molekulare Netzwerke - Herausforderung für Biologie und Pharmaforschung. In: *Magdeburger Wissenschaftsjournal* (2004), Nr. 2, S. 47–54
- [Schubert 2010] SCHUBERT, Walter: On the origin of cell functions encoded in the topomome. In: *Journal of Biotechnology* 149 (2010), September, Nr. 4, S. 252–259

- [Schubert et al. 2006] SCHUBERT, Walter ; BONNEKOH, Bernd ; POMMER, Ansgar J. ; PHILIPSEN, Lars ; BÖCKELMANN, Raik ; MALYKH, Yanina ; GOLLNICK, Harald ; FRIEDENBERGER, Manuela ; BODE, Marcus ; DRESS, Andreas W M.: Analyzing proteome topology and function by automated multidimensional fluorescence microscopy. In: *Nature Biotechnology* 24 (2006), Oktober, Nr. 10, S. 1270–1278
- [Schubert et al. 2009] SCHUBERT, Walter ; GIESELER, Anne ; KRUSCHE, Andreas ; HILLERT, Reyk: Toponome mapping in prostate cancer: detection of 2000 cell surface protein clusters in a single tissue section and cell type specific annotation by using a three symbol code. In: *Journal of Proteome Research* 8 (2009), Juni, Nr. 6, S. 2696–2707
- [Schubert et al. 2011] SCHUBERT, Walter ; GIESELER, Anne ; KRUSCHE, Andreas ; SEROCKA, Peter ; HILLERT, Reyk: Next generation biomarkers based on 100-parameter functional super-resolution microscopy TIS. In: *New Biotechnology (in press)* (2011)
- [Schumann und Müller 1999] SCHUMANN, Heidrun ; MÜLLER, Wolfgang: *Visualisierung: Grundlagen und allgemeine Methoden*. Springer Berlin Heidelberg, 1999. – 370 S.
- [Sharma et al. 2005] SHARMA, Gaurav ; WU, Wencheng ; DALA, Edul N.: The CIE-DE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. In: *Color Research and Application* (2005)
- [Wang et al. 2008] WANG, Lujin ; GIESEN, Joachim ; McDONNELL, Kevin T. ; ZOLLIKER, Peter ; MUELLER, Klaus: Color design for illustrative visualization. In: *Ieee Transactions on Visualization and Computer Graphics* 14 (2008), Oktober, Nr. 6, S. 1739–1746
- [Ware 2004] WARE, Colin: *Information Visualization, Second Edition: Perception for Design (Interactive Technologies)*. 2. Morgan Kaufmann, 2004. – ISBN 1558608192
- [Wilkins et al. 1996] WILKINS, Marc R. ; SANCHEZ, Jean-Charles ; GOOLEY, Andrew A. ; APPEL, Ron D. ; HUMPHERY-SMITH, Ian ; HOCHSTRASSER, Denis F. ; WILLIAMS, Keith L. ; PASQUALI, Christian ; OU, Keli ; GOLAZ, Oliver ; YAN, Jun X. ; HUGHES, Graham: Progress with proteome projects: why all proteins expressed by a genome should be identified and how to do it. In: *Biotechnology & genetic engineering reviews* 13 (1996), S. 19–50
- [Zhai 1998] ZHAI, Shumin: User performance in relation to 3D input device design. In: *ACM Siggraph Computer Graphics* (1998)