Vergleich von Proteinbindetaschen unter Verwendung neuronaler Netze: von der Struktur zur Funktion und zum Ligandendesign

Dissertation

zur Erlangung des Doktorgrades
der Naturwissenschaften
(Dr. rer. nat.)

dem Fachbereich Mathematik u. Informatik
der Philipps-Universität Marburg
vorgelegt von

Katrin Kupas
aus Duisburg

Marburg, Lahn 2006

Vom Fachbereich Mathematik & Informatik der Philipps-Universität Marburg

als Dissertation angenommen am 22.08.2006

Erstgutachter: Prof. Dr. Alfred Ultsch

Zweitgutachter: Prof. Dr. Gerhard Klebe

Tag der mündlichen Prüfung 29.08.2006

Danksagung

Die Untersuchungen in der vorliegenden Arbeit wurden in Kooperation zwischen der AG Datenbionik von Prof. Ultsch am Fachbereich Mathematik und Informatik und der Arbeitsgruppe von Herrn Prof. Dr. G. Klebe am Institut für Pharmazeutische Chemie des Fachbereichs Pharmazie der Philipps Universität Marburg in der Zeit von Oktober 2003 bis Juni 2006 durchgeführt. Mein Dank gilt Prof. Ultsch und Prof. Klebe für die besonders interessante Themenstellung. Die immer gewährte Unterstützung, das richtige Maß an Kritik und die Freiheit, selbstständig Ideen zu entwickeln, trugen zur einer hervorragenden Betreuung und Arbeitsatmosphäre bei.

Meinem Kollegen Nils Weskamp danke ich für die schöne Zusammenarbeit, die wichtigen Diskussionen, das Korrekturlesen von Teilen meiner Arbeit und nicht zuletzt für die vielen schönen Sushi- und Cocktailabende und einen unvergesslichen Urlaub in New York.

Für die wichtigen Diskussionen während der drei Jahre und das Korrekturlesen von Teilen meiner Arbeit danke ich meinem Kollegen Fabian Mörchen ganz herzlich.

Ich danke allen Mitarbeitern am Fachbereich Mathematik und Informatik für die schöne Arbeitsatmosphäre, insbesondere David Kämpf, Markus Vincon, Oliver Dippel und Elko Jacobs als harten Kern bei allen möglichen außeruniversitären Aktivitäten.

Roman Meisl danke ich für den endlosen Nachschub an Kaffee, Keksen und neuen Kartenspielen und den immer sehr netten Support beim Doppelkopf.

Den Mitgliedern der AG Klebe gilt mein Dank für die nette Aufnahme in die Gruppe trotz des Status der externen Mitarbeiterin und die schöne Zeit in Hirschegg.

Tina Ritschel, Conny Koch, Patrick Pfeffer, Naomi Tidten, Lisa Englert, Bernhard Baum und Nils Weskamp danke ich für die schönen Parties und netten Abende.

Mein ganz besonderer Dank gilt Angela Scholz für die immer perfekte Organisation und ihren ständigen Einsatz.

Ich danke meiner Familie, meiner besten Freundin Janine und meinem Freund Boris für die viele emotionale Unterstützung während meiner Arbeit, vor allem in den letzten Wochen.

Inhaltsverzeichnis

1. Einleitung **4**

2. Grundlagen **7**

2.1. Data Mining 7

2.2. Mathematische Grundlagen 18

2.3. Biologische Grundlagen 23

3. Ansätze zur Klassifizierung von Proteinen **28**

3.1. Modellierung der Proteinstrukturen anhand der Aminosäuren 29

3.2. Modellierung der Lösungsmittel-zugänglichen Oberfläche des Proteins bzw. der Bindetasche 34

3.3. Zusammenfassung der Ansätze zum Vergleich von Proteinstrukturen . . . 38

4. Datensätze **41**

4.1. Datensatz aus 109 Bindetaschen zu 12 ausgewählten EC-Nummern ohne Ähnlichkeiten untereinander (*DS V1*) 41

4.2. Datensatz aus 708 Bindetaschen zu 13 ausgewählten EC-Nummern ohne Ähnlichkeiten (*DS V2*) 42

4.3. Datensatz aus 1494 Bindetaschen zu den 13 EC-Nummern mit den meisten Einträgen in CAVBASE (*DS T1*) 42

4.4. Datensatz aus 283 Bindetaschen von Endopeptidasen (*DS T2*) 46

4.5. Erweiterung des Datensatzes aus Bindetaschen von Endopeptidasen (*DS P*) 47

4.6. Zusammenfassung der vorgestellten Datensätze 48

5. Klassifizierung von Oberflächenpatches **49**

5.1. Algorithmus zur Klassifizierung der Substrukturen 50

5.2. Beschreibung der Oberfläche durch Mittelwert und Varianz von Patchsektoren (SectorGeoMap) 54

5.3. Beschreibung der Oberfläche durch Wavelets (WaveGeoMap) 56
5.4. Charakterisierung der physikochemischen Eigenschaften der Patches . . . 57
5.5. Parameterstudien zur Patchgröße 58
5.6. Zusammenfassung der Klassifizierung von Oberflächenpatches 60

6. Ergebnisse der Klassifizierung von Oberflächenpatches 63
6.1. Ergebnisse der SectorGeoMap zum Vergleich der Patches 65
6.2. Ergebnisse der Methode der WaveGeoMap 68
6.3. Vergleich mit anderen Clusteralgorithmen 105
6.4. Zusammenfassung der Ergebnisse der Klassifizierung von Oberflächenpatches 110

7. Diskussion 113

8. Zusammenfassung 120

A. Mathematische Symbole 122

B. Algorithmen zur Klassifikation der Oberflächenpatches 125

C. Klassifizierung minimaler Substrukturen 132

Abbildungsverzeichnis 149

Tabellenverzeichnis 151

Literaturverzeichnis 153

Lebenslauf 161

1. Einleitung

Eines der ehrgeizigsten Forschungsprojekte der modernen Biowissenschaften konnte im Sommer 2004 [IHG, 2004] nach der initialen Entschlüsselung 2001 [Lander, 2001, Venter, 2001] endgültig abgeschlossen werden: die Entzifferung des humanen Genoms. Die Hoffnung besteht, durch die gezielte Isolierung einzelner Gene völlig neue Zielproteine (*Targets*) für die Arzneistoff-Forschung zu identifizieren. Der Weg vom Genom zum therapeutischen Target und zum anwendbaren Arzneistoff umfasst eine komplexe Prozesskette, die an vielen Stellen durch Ansätze der Bioinformatik angegangen werden kann. Computerprogramme zum intelligenten Vergleich von Aminosäuresequenzen der Proteine und zum Erkennen von Ähnlichkeiten in diesen Sequenzen existieren bereits. Algorithmen, die die dreidimensionale Struktur der Proteine aus ihrer Primärsequenz vorhersagen, stehen leider noch weitgehend am Anfang ihrer Entwicklung. Das immer häufigere Auftreten von Redundanzen bei den Faltungsmustern von Proteinen aus unterschiedlichen Familien zeigt jedoch einen neuen Ansatz: So lassen sich die Strukturen anhand der Faltungsmuster ähnlicher Proteine vohersagen. Diese Homologiemodellierung wird in Zukunft eine immer wichtigere Rolle spielen. Zusätzlich konnten die experimentellen Techniken wie Röntgenstrukturanalyse und NMR-Spektroskopie zur Aufklärung der Raumstruktur von Proteinen in den letzten Jahren deutlich verbessert und beschleunigt werden. Somit werden wir zunehmend mit der Tatsache konfrontiert werden, dass für immer mehr Sequenzen eine Homologiemodellierung aus einer bekannten Röntgen- oder NMR-Struktur aussichtsreich erscheint. Infolge dessen wird sich immer häufiger die Situation ergeben, dass zwar ein Vorschlag für eine Raumstruktur vorliegt, die eindeutige Funktionszuweisung des untersuchten Proteins aber noch unbekannt ist. Eine Funktionsvorhersage aus Sequenz und Faltung ist keinesfalls trivial, denn die Funktion eines Proteins ist nicht notwendigerweise durch ein bestimmtes Faltungsmuster festgelegt bzw. ein gleiches Faltungsmuster muss nicht für dieselbe Funktion stehen.

Proteine besitzen auf ihrer Oberfläche Taschen, in denen endogene Liganden oder Substrate erkannt, gebunden und chemisch verändert werden können. In Enzymen laufen zum Beispiel gezielt einzelne Schritte chemischer Reaktionen ab, die eine fest vorgegebe-

ne Anordnung der Reaktionspartner zueinander erfordern. Dies verlangt wiederum auf der Seite der Proteine ein hochkonserviertes Muster an molekularen Erkennungseigenschaften, die diese räumliche Ausrichtung der Reaktionspartner aufeinander festlegen. Daher liegt es auf der Hand anzunehmen, dass ein so definiertes Muster von Erkennungseigenschaften in Bindetaschen mit einer bestimmten Funktion eines Proteins einhergeht. Somit ist vorauszusetzen, dass Proteine vergleichbarer Funktion ein verwandtes Muster molekularer Erkennungseigenschaften in ihren Bindetaschen abbilden - und zwar unabhängig von dem weiteren Faltungs- und Sequenzmuster des betrachteten Proteins. Existieren andere Proteine mit vergleichbaren molekularen Erkennungseigenschaften in ihren Bindetaschen, so kann es auch dort zu einer Bindung des Liganden kommen. Daraus kann man zwei Dinge ableiten. Zum einen können von ihrem chemischen Aufbau strukturell verschiedene Liganden identifiziert werden, die in unterschiedliche Proteine mit ähnlichen Bereichen in den Taschen binden. Wenn Ligandenfragmente in Substrukturen binden, die physikochemisch ähnliche Eigenschaften aufweisen, werden dadurch wertvolle Ideen für das Design und den Austausch von bioisoteren Verbindungen geliefert. Zum anderen können sich Ähnlichkeiten in den Bindetaschen von zwei im Organismus in unterschiedlichen Kompartimenten vorliegenden Proteinen in einer unerwünschten Nebenwirkung manifestieren.

Durch den Vergleich von Substrukturen in diesen Bindetaschen können die Funktion von Proteinen und potentielle Ligandenfragmente vorhergesagt werden. Künstliche Neuronale Netze sind in der Lage, den algorithmischen Aufwand für einen effektiven Vergleich dieser Substrukturen in den Bindetaschen zu leisten. Eine grundsätzliche Eigenschaft der neuronalen Netze ist es jedoch, dass diese Verfahren in der Regel nicht zu einem tieferen Verständnis der durch sie gelösten Aufgabe führen. Um die Mechanismen der Strukturanpassungen tiefer verstehen zu können ist daher *Data Mining* bzw. *Knowledge Discovery* erforderlich. *Data Mining* ist die explorative Analyse eines großen Datensatzes, der zentrale Punkt des *Data Mining* ist der Übergang von Daten zu Wissen (*Knowledge Discovery*). Dabei ist Wissen eine Repräsentation von Tatsachen in einer formalen Sprache, damit ein kompetenter Interpreter dieses Wissen weiterverwenden kann. Auch dem menschlichen Nutzer muss es möglich sein, dieses Wissen zu lesen, zu verstehen und auszuwerten. Darüber hinaus sollte das Wissen auch in wissensbasierten Systemen nutzbar sein, so dass eine Analyse, Diagnose, Simulation oder Prognose des Prozesses möglich ist, der die Daten generiert hat. Zum *Data Mining* werden die verschiedensten Techniken genutzt. Das Spektrum reicht von traditionellen statistischen Techniken zur Datenanalyse über naturanaloge Verfahren wie neuronale Netze und evo-

lutionäre bzw. genetische Algorithmen sowie Methoden des *Machine Learning* bis hin zu *Fuzzy*-Verfahren [Hand et al., 2001]. Inzwischen ist die Entwicklung von *Data-Mining*-Algorithmen so weit fortgeschritten, dass sie sich in einer beachtlichen Zahl von kommerziellen *Data-Mining*-Tools manifestiert hat. Alle darin benutzten Techniken bieten Lösungsansätze für die Analyse von Strukturen in hochdimensionalen Daten, das Wissen wird allerdings häufig vernachlässigt.

Im Rahmen dieser Arbeit wurden Algorithmen für den Vergleich von Substrukturen in Proteinbindetaschen entwickelt und verschiedene Ansätze zur Charakterisierung solcher Substrukturen vorgestellt. Die erhaltenen Deskriptoren werden mit Hilfe einer emergenten selbstorganisierenden Merkmalskarte (ESOM) topologietreu auf ein zweidimensionales Neuronengitter projiziert und die Distanzstruktur zum Auffinden von Clustern in den hochdimensionalen Daten visualisiert. Mit der erhaltenen ESOM wird eine automatische Clusterung durchgeführt, die lokale Distanz- und Dichteinformationen berücksichtigt, um Gemeinsamkeiten in den Substrukturen zu finden.

Die Arbeit ist wie folgt gegliedert: In Kapitel 2 werden die Prinzipien des *Data Mining* sowie die mathematischen und biologischen Grundlagen für die Algorithmen erläutert. Der aktuelle Stand der Forschung wird in Kapitel 3 dargelegt. Die in dieser Arbeit verwendeten Datensätze zur Validierung der Methode werden in Kapitel 4 beschrieben. Der Kernteil der Arbeit findet sich in den Kapiteln 5 und 6, in denen ein neuer Ansatz zur Zerlegung von Proteinbindetaschen in Substrukturen und ihrer Charakterisierung sowie die Ergebnisse der Clusterung dieser Substrukturen vorgestellt werden. Eine Diskussion ist in Kapitel 7, eine Zusammenfassung dieser Arbeit in Kapitel 8 enthalten.

2. Grundlagen

2.1. Data Mining

Künstliche Neuronale Netze

Definition 2.1.1 *Ein künstliches Neuron ist ein Tupel* (x, w, f_a, f_o, o) *bestehend aus einem Eingabevektor* $x = (x_1, ..., x_n) \in \mathbb{R}^n$*, einem Gewichtsvektor* $w = (w_1, ..., w_n) \in \mathbb{R}^n$*, einer Aktivierungsfunktion* f_a *mit* $f_a : \mathbb{R}^n \times \mathbb{R}^n \longrightarrow \mathbb{R}$ *und einer Ausgabefunktion* f_o*, für die* $f_o : \mathbb{R} \longrightarrow \mathbb{R}$ *gilt. Dabei wird durch* $f_o(f_a(x, w)) = o$ *der Ausgabewert des Neurons erzeugt, der an die nachfolgenden Neuronen weitergegeben wird. Die Aktivierungs- und Ausgabefunktion ist meist eine nichtlineare, häufig sigmoide Funktion.*

Ein neuronales Netz entsteht durch die Verbindung künstlicher Neuronen. Es ist also ein Paar (N, V) mit einer Menge N von Neuronen und einer Menge V von Verbindungen. Es kann aus mehreren Schichten bestehen, einer Ein- und Ausgabeschicht und beliebig vielen Zwischenschichten. Neuronale Netze besitzen die Struktur eines gerichteten Graphen:

- Die Knoten des Graphen sind die Neuronen.
- Die Kanten sind die Verbindungen zwischen den Neuronen.
- Jedes Neuron kann durch eine beliebige Menge von Kanten verbunden sein, über die es seine Eingaben erhält.
- Jedes Neuron kann genau eine Ausgabe über eine beliebige Menge von Verbindungen aussenden.
- Das Neuronale Netz erhält aus Verbindungen von außen Eingaben und gibt seine Ausgaben über außen endende Verbindungen ab.

Alle Verbindungen, die von anderen Neuronen zu einem einzelnen Neuron j gehen, ergeben den Eingabevektor x_j von j. Da bei den meisten neuronalen Netzen die Eingabe

gewichtet wird, kann man die Verbindungsstruktur (Topologie) in Form einer Matrix beschreiben. Das Gewicht der Verbindung von Neuron i zu Neuron j ist dann das Element w_{ij} der Matrix. Der Wert von w_{ij} gibt an, ob die Verbindung existiert, und wenn ja, ob sie anregend oder hemmend ist:

$$\begin{aligned} w_{ij} = 0 &: \text{keine Verbindung von Neuron i zu Neuron j} \\ w_{ij} < 0 &: \text{hemmende Verbindung der Stärke } |w_{ij}| \\ w_{ij} > 0 &: \text{anregende Verbindung der Stärke } |w_{ij}| \end{aligned}$$

Man unterscheidet Netze ohne (*feedforward*-Netze) und Netze mit Rückkopplung (rekurrente Netze). Bei Netzen ohne Rückkopplungen existiert kein Pfad, der von einem Neuron direkt oder über zwischengeschaltete Neuronen wieder zurück zu diesem Neuron führt. Daten werden also nur in eine Richtung weitergegeben. Bei den Netzen mit Rückkopplung unterscheidet man solche mit direkter, indirekter Rückkopplung, mit Rückkopplung innerhalb einer Schicht und vollständig verbundene Netze.

Ein neuronales Netz „lernt“, indem es sich gemäß einer fest vorgegebenen Lernregel selbst modifiziert. Dabei unterscheidet man zwischen überwacht, bestärkend und unüberwacht lernenden Netzen.

Beim überwachten Lernen (*supervised learning*) wird dem Netz zu jeder Eingabe die korrekte Ausgabe oder die Differenz der tatsächlichen zur korrekten Ausgabe angegeben. Anhand dieser Differenz wird dann das Netz über die Lernregel modifiziert. Diese Technik setzt allerdings voraus, dass Trainingsdaten exisitieren, die aus Paaren von Ein- und Ausgabedaten bestehen.

Ein typisches überwachtes Lernverfahren wie z.B. *Backpropagation* durchläuft für alle Paare von Ein- und Ausgabemustern folgende Schritte:

1. Das Eingabemuster wird dem Netz durch entsprechende Aktivierung der Eingabeneuronen präsentiert.

2. Die angelegte Eingabe läuft vorwärts durch das Netz. Dadurch wird ein Ausgabemuster für die aktuelle Eingabe erzeugt.

3. Tatsächliche und korrekte Ausgabe werden verglichen und die Differenz berechnet.

4. Die Fehler laufen rückwärts von der Ausgabe- zur Eingabeschicht. Dabei werden die Verbindungsgewichte verändert, so dass der Fehler verringert wird.

5. Die Gewichte aller Neuronen werden um die vorher berechneten Werte verändert.

Beim bestärkenden Lernen (*reinforcement learning*) wird im Gegensatz zum überwachten Lernen dem Netz lediglich mitgeteilt, ob seine Ausgabe korrekt oder inkorrekt war. Das Netz erfährt nicht den exakten Wert des Unterschiedes.

Beim unüberwachten Lernen (*unsupervised learning*) gibt es überhaupt keine externe Bewertung der Ausgabe während des Lernens, daher heißt dieses Lernparadigma auch *self-organized learning*. Das Netz versucht ohne Beeinflussung von außen die präsentierten Daten in Ähnlichkeitsklassen aufzuteilen.

Selbstorganisierende Merkmalskarten (SOM)

Die selbst-organisierende Merkmalskarte (SOM), auch Kohonenkarte genannt, wird dem Gebiet der Künstlichen Neuronalen Netze (KNN) zugeordnet. Entwickelt wurde sie 1982 vom finnischen Physiker Teuvo Kohonen [Kohonen, 1982]. Das Lernverfahren der Kohonenkarte ist unüberwacht, da sie die Struktur des Eingaberaumes in interne Strukturen abbildet. Die Netzwerkstruktur ist einschichtig, mit einer *Input-* und einer *Output-*Schicht.

Definition 2.1.2 *Eine Selbstorganisierende Merkmalskarte (SOM, [Kohonen, 1982]) besteht aus einer Neuronenmenge $N = P \times W$. Jedes Neuron $i \in N$ ist ein Tupel (w_i, p_i) bestehend aus dem Gewichtsvektor $w_i \in W$ und einer Position $p_i \in P$.*

Der Ausgaberaum der SOM (= Karte) besteht aus der Menge $P = p_i : i \in \mathbb{N}$ der Neuronenpositionen. Die SOM besteht aus einem regelmäßigen Gitter von Neuronen N, d.h. jedes Neuron $i \in N$ hat eine feste Anzahl äquidistanter Nachbarn $NN^M(i) \subset I$. Im Folgenden wird P als 2-dimensional vorausgesetzt.

Auf P ist das Abstandsmaß *md* definiert. Das Abstandsmaß *md* berechnet die Abstände der Neuronenpositionen anhand des Netzgitters. Für *md* existieren unterschiedliche Realisationen. Sofern nicht anders angegeben, ist der Abstand zweier Neuronenpositionen die minimale Länge eines Pfades im Netzgitter zwischen beiden Neuronenpositionen [Kohonen, 1997].

Die selbstorganisierende Merkmalskarte (SOM) ist ein Projektionsverfahren, bei dem die Datenvektoren aus dem Eingaberaum $D \subset \mathbb{R}^n$ mit der Distanzfunktion $d : D \times D \rightarrow \mathbb{R}_0^+$ auf eine niedrig-dimensionale Mannigfaltigkeit eingebettet in $\mathbb{R}^l$, $l < n$, mit der Distanzfunktion *md* abgebildet werden. Die Projektion erhält die Nachbarschaftsbeziehungen des Eingaberaums D so gut wie möglich.

Die am häufigsten benutzten Varianten des Ausgaberaums P sind das zweidimensionale Quadgrid mit vier direkten Nachbarn pro Neuron und das zweidimensionale Hexgrid mit sechs unmittelbaren Nachbarn pro Neuron [Ultsch and Herrmann, 2005]. Diese beiden Gitterstrukturen sind in Abbildung 2.1 dargestellt.

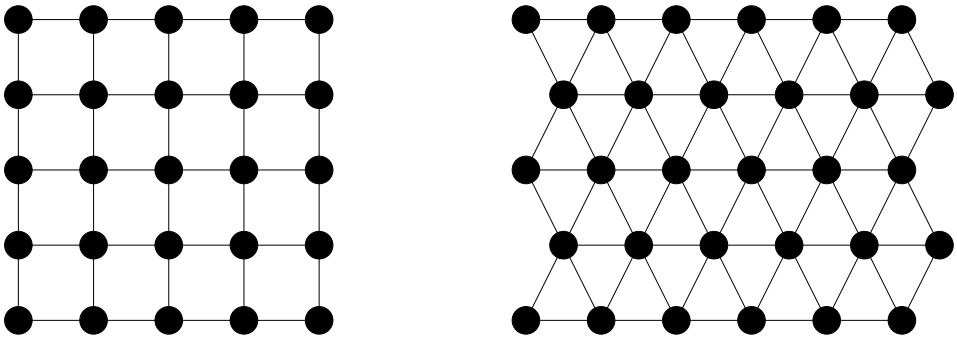


Abbildung 2.1.: Quadgrid vs. Hexgrid

Die Gewichtsvektoren $W \subset D$ werden für die Vektorquantisierung, die Positionen $P \subset M$ werden für die Vektorprojektionen

$$m : D \to P, \quad m(x) = p_{bm(x)}$$

mit

$$bm(x) = argmin_{i \in I}\ d(x, w_i)$$

verwendet.

Der Lernalgorithmus der SOM ist der folgende: Sei r der Lernradius und $\theta \in [0, 1]$ die Lernrate. Die Gewichtsvektoren $w_i \in W$ der Neuronen werden anfänglich beliebig aus der Menge der Eingabevektoren D initialisiert. Der folgende Lernzyklus wird so oft durchlaufen, bis die vorgegebene Anzahl an Zyklen erreicht ist. Zu jedem Eingabevektor $x_i \in D$ wird der *Bestmatch* $bm(x_i)$ bestimmt. Alle Gewichtsvektoren $w_l \in W$ der Neuronen $i \in I$ mit $md(i, bm(x_i)) \leq r$ werden dem Eingabevektor durch

$$\Delta w_l = \theta * (x_i - w_l)$$

angepasst.

Die *Bestmatch*-Suche und die Anpassung in der Nachbarschaft wird pro Lernzyklus für alle Eingabedaten durchgeführt. Die Reihenfolge ist zufällig, aber jeder Eingabedatensatz kommt gleich häufig vor. Der Nachbarschaftsradius r wird zu Beginn des SOM-Trainings sehr groß gewählt und mit zunehmender Anzahl von Lernzyklen sukzessive

verkleinert. Neben dem Nachbarschaftsradius kann auch die Änderungsrate während des Prozesses verändert werden.

Man kann zwei verschiedene SOM-Typen unterscheiden. Der erste Typ, die sogenannte „*K-means*-SOM“, enthält nur sehr wenige Neuronen und wird ausschliesslich zum Clustern verwendet [Kaski, 1997]. Jedes Neuron entspricht dabei einem Cluster. Die SOM kann hierbei als eine Form des k-means-Cluster-Algorithmus angesehen werden. Die Anzahl der Neuronen entspricht der Anzahl der angenommenen Cluster. Im Gegensatz dazu unterscheidet man die sogenannte emergente SOM (ESOM) mit einer sehr großen Anzahl von Neuronen, üblicherweise mehr als 4000 [Ultsch, 1999]. Durch die große Anzahl von Neuronen ist es möglich, Strukturen in den Eingabedaten zu finden, wie z.B. hochdimensionale Clusterstrukturen. Die emergente SOM ist ein Verfahren zur Projektion hochdimensionaler Eingabedaten, die die Nachbarschaftsbeziehungen des Eingaberaums erhält. U- bzw. P-Matrix-Verfahren können zur Visualisierung der Strukturen verwendet werden [Ultsch, 2003c]. Im Gegensatz zu vielen anderen nichtlinearen Projektionsverfahren wie z.B. *Sammons Mapping* oder *Multidimensionale Skalierung* leistet die ESOM eine „Entflechtung“ des Datenraumes.

Bei planaren Karten entsteht oft das Problem der Randeffekte. Neuronen am Rand der Karte haben weniger Nachbarn als solche in der Mitte der Karte und können somit weniger Nachbarneuronen beeinflussen. Das Resultat ist häufig, dass sich die interessanten Strukturen am Rand der Karte befinden und dort auch häufig voneinander getrennt sind. Um diese Effekte zu vermeiden, werden toroide Karten verwendet, die einen randlosen Kartenraum bieten [Ultsch, 2003c].

Visualisierungsmethoden von emergenten SOM

Um die Strukturen in den hochdimensionalen Gewichtsvektoren der Neuronen erkennen zu können, braucht man eine Visualisierung der ESOM. Dazu wurden die U-, P- und U*-Matrix entwickelt. Die U-Matrix gibt die Distanzbeziehungen, die P-Matrix die Dichtebeziehungen und die U*-Matrix gibt kombinierte Distanz- und Dichtebeziehungen zwischen den Vektoren wieder.

U-Matrix:

Die U-Matrix ist eine Landschaft der Distanzstruktur der Eingabedaten der ESOM [Ultsch and Siemon, 1990]. Dafür wird für jedes Neuron i mit dem zugehörigen Gewichtsvektor w_i die U-Höhe $UH(i)$ berechnet. Sie ist die Summe der Abstände des Ge-

wichtsvektors w_i zu den Gewichtsvektoren w_j der unmittelbar benachbarten Neuronen.

Definition 2.1.3 (U-Höhe) *Sei $i \in I$, $NN^M(i) \subset I$ die direkt benachbarten Neuronen des Neurons i, w_k die zugehörigen Gewichtsvektoren. Die U-Höhe $UH(i)$ ist definiert als*

$$UH(i) = \sum_{j \in NN^M(i)} md(w_i, w_j).$$

Die U-Matrix ist die Visualisierung dieser U-Höhen auf den Positionen der zugehörigen Neuronen. Niedrige Distanzen zu den benachbarten Vektoren ergeben niedrige U-Höhen, große Distanzen ergeben große U-Höhen. Diese U-Höhen werden ähnlich einer Landkarte eingefärbt. Niedrige U-Höhen sind Täler auf der Karte und werden in blau und grün dargestellt. Große U-Höhen ergeben Gebirgszüge und sind dementsprechend braun und weiß dargestellt. Die Farbcodierung ist in Abbildung 2.2 dargestellt. Datenpunkte, die ihre Bestmatches in Regionen mit niedrigen U-Höhen haben, liegen auch benachbart im Eingaberaum. Gebirgszüge trennen die weiter voneinander entfernt liegenden Daten. Somit sind Clusterstrukturen, falls sie existieren, in den Tälern auf der U-Matrix zu finden, die Berge sind die Grenzen zwischen den Clustern.

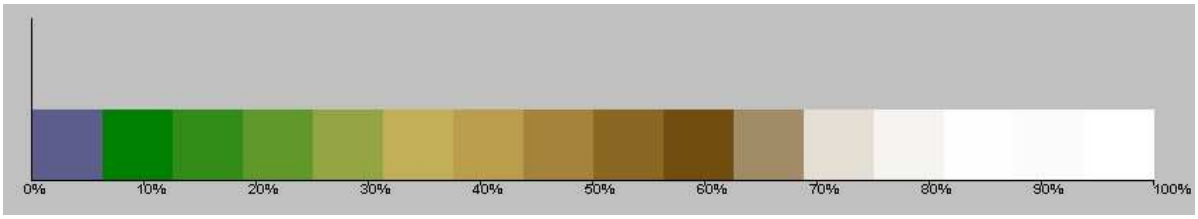


Abbildung 2.2.: Farbcodierung der Einfärbung der auf [0,1] normierten U-Höhen ähnlich einer Landkarte. Niedrige U-Höhen entsprechen Tälern, große U-Höhen Gebirgszügen.

P-Matrix:

Die P-Matrix ist eine Visualisierung der Dichtebeziehungen zwischen den Datenpunkten [Ultsch, 2003c]. Sie ist eine Darstellung von P-Höhen auf den Positionen der Neuronen. Für jedes Neuron i ist die P-Höhe $PH(i)$ die Anzahl an Datenpunkten in einer Hypersphäre um den zugehörigen Gewichtsvektor w_i im Datenraum. Der Radius dieser Hypersphäre, der sogenannte Paretoradius, ermittelt sich auf der Basis der Informationsoptimalität von Mengen [Ultsch, 2003a].

Definition 2.1.4 (Information einer Menge) *Sei S eine Teilmenge einer Menge von Punkten und p sei die Wahrscheinlichkeit, dass ein Punkt zu S gehört. Die Information von S skaliert auf [0,1] ist*

$$I(S) = -e \cdot p \cdot \ln(p).$$

Eine informationsoptimale Menge ist so klein wie möglich und enthält dabei möglichst viel Information.

Definition 2.1.5 (unrealisiertes Potential) *Das unrealisierte Potential $URP(S)$ einer Menge S ist die Distanz vom idealen Punkt (0,1) zum Punkt $(p, I(S))$ der gegebenen Menge.*

Das Minimum von URP liegt bei p_u =20.13%. Diese Mengengröße enthält 88% Information. Teilmengen einer ungefähren Größe von p_u sind also informationsoptimal.

Definition 2.1.6 (Pareto-Radius) *Sei $d(x_i, x_j)$ ein Abstandsmaß definiert auf der Menge $D = x_1, ..., x_d$, $|D| = d$. $N(x, r) = |\{x_i \in D \mid d(x, x_i) \leq r\}|$ ist die Anzahl an Punkten innerhalb einer Sphäre mit dem Radius r um x. Der Pareto-Radius r_p ist der Radius, für den gilt*

$$median(N(x_i, r_p)) = p_u \cdot d \; \forall x_i \in D$$

Das bedeutet, dass die Sphären im Mittel informationsoptimale Teilmengen der Datenpunkte enthalten.

Große P-Höhen bedeuten eine große Datendichte, niedrige P-Höhen bedeuten Regionen mit geringer Datendichte. Clusterstrukturen sind somit in Regionen mit großen P-Höhen zu finden, während sich Clustergrenzen durch niedrige P-Höhen auszeichnen.

U*-Matrix:

Die U*-Matrix verbindet die Distanz- und Dichtedarstellung der U- und der P-Matrix [Ultsch, 2003b].

Definition 2.1.7 (U*-Höhe) *Sei meanP der Mittelwert aller P-Höhen und minP das Minimum aller P-Höhen. Dann ist die U*-Höhe $UH^*(i)$ des Neurons i*

$$UH^*(i) = UH(i) * \delta(i)$$

mit

$$\delta(i) = \frac{PH(i) - meanP}{meanP - minP}$$

Das bedeutet:

- Entspricht die Dichte der Datenpunkte am *Bestmatch* des jeweiligen Neurons der mittleren Datendichte, so ist die U*-Höhe gleich der U-Höhe.
- Ist die Dichte niedriger als die mittlere Datendichte, so ist die U*-Höhe höher als die entsprechende U-Höhe.
- Ist die Dichte größer als die mittlere Datendichte, so ist die U*-Höhe niedriger als die entsprechende U-Höhe.
- Ist die Dichte der Datenpunkte am Bestmatch des jeweiligen Neurons gleich dem Minimum der Dichte, so ist die U*-Höhe gleich null.

Dadurch werden die Täler auf der U-Matrix, in denen die Daten dicht liegen, vertieft, und die Cluster begrenzenden Berge, die auch geringere Datendichte aufweisen, werden höher. Clusterstrukturen in den Daten werden somit basierend auf Dichte und Distanz deutlicher dargestellt.

Visualisierung von toroiden Karten:

Um die randlose toroide ESOM planar darstellen zu können, wird die sogenannte gekachelte Darstellung verwendet [Ultsch, 2003c]. Dazu wird die U-, P- oder U*-Matrix viermal nebeneinander dargestellt, eine wird rechts an die Originalmatrix und zwei nebeneinander darunter angefügt. Durch diese Vierfachdarstellung erhält man in der Mitte der vier Matrizen eine randlose Visualisierung der toroiden Karte. Der Nachteil ist jedoch, dass jedes Neuron und damit auch jede Struktur auf der Karte viermal vorhanden sind. Man kann aber aus der Vierfachdarstellung so eine Insel ausschneiden, dass zusammenhängende Strukturen zusammen bleiben und jedes Neuron nur einmal dargestellt ist [Ultsch, 2003c]. Diese Darstellung wird im Folgenden Inseldarstellung genannt.

Cluster-Algorithmen

Clusteralgorithmen[1] dienen der Entdeckung von abgrenzbaren Gruppen von Daten, die möglichst viele Gemeinsamkeiten besitzen. Ziel ist es, zusammengehörige Gruppen zu identifizieren und die Daten entsprechend ihrer Zugehörigkeit einzuordnen. Man unterscheidet hierarchische und partitionierende Clusteralgorithmen, wobei man erstere noch in anhäufende (*bottom-up*) oder unterteilende (*top-down*) Algorithmen unterteilt.

Je nach Algorithmus muss eine Distanzfunktion zur Bestimmung des Abstands zweier Elemente (zum Beispiel die euklidische Distanz) und/oder eine Methode zur Berechnung des Mittelpunktes oder Zentroiden eines Clusters (zum Beispiel der Mittelwert) bekannt sein. Anstatt einer Distanzfunktion arbeiten einige Algorithmen auch mit einer Ähnlichkeitsfunktion.

Hierarchisches Clustern

Grundsätzlich lassen sich anhäufende Verfahren und teilende Verfahren unterscheiden. Bei den anhäufenden Verfahren, die in der Praxis häufiger eingesetzt werden, werden schrittweise einzelne Objekte zu Clustern und diese zu größeren Gruppen zusammengefasst, während bei den teilenden Verfahren größere Gruppen schrittweise immer feiner unterteilt werden. Die bei der hierarchischen Clusterung entstehende Baumstruktur wird in der Regel mit einem Dendrogramm visualisiert.

Beim anhäufenden Clustern wird zunächst jedes Objekt als ein eigener Cluster mit einem Element aufgefasst. Nun werden in jedem Schritt die jeweils einander nächsten Cluster zu einem Cluster zusammengefasst. Das Verfahren kann beendet werden, wenn alle Cluster eine bestimmte Distanz zueinander überschreiten oder wenn eine genügend kleine Zahl von Clustern ermittelt worden ist. Aus verschiedenen Methoden zur Bestimmung des Abstands zweier Cluster ergeben sich verschiedene Verfahren.

Der Algorithmus für die anhäufenden Verfahren ist wie folgt:

1. Initialisierung: Jeder Datenpunkt ist ein Cluster.
2. Zuordnung: Fasse zwei Cluster mit minimalem Clusterabstand zu einem zusammen und ordne diesem Cluster einen neuen Punkt im $\mathbb{R}^n$ zu.
3. Neuberechnung: Beginne von neuem bei Schritt 2, bis die gewünschte Clusteranzahl erreicht ist.

[1] Aus http://de.wikipedia.org/wiki/Clusteranalyse.

Für den Abstand zweier Cluster A und B lassen sich unter anderem folgende Abstände verwenden:

- der minimale Abstand zweier Elemente aus den beiden Clustern (*single linkage clustering*),
- der maximale Abstand zweier Elemente aus den beiden Clustern (*complete linkage clustering*),
- der durchschnittliche Abstand aller Elementpaare aus den beiden Clustern (*average linkage clustering*),
- der durchschnittliche Abstand aller Elementpaare aus der Vereinigung von A und B(*average group linkage*)
- der Abstand der Mittelwerte der beiden Cluster (*centroid method*),
- die Zunahme der Varianz beim Vereinigen von A und B (*Ward's method*).

Für größere Datensätze kann der Rechenaufwand für die hierarchischen Clusterverfahren zu groß werden. In diesen Fällen können dann nur die partitionierenden Verfahren eingesetzt werden.

Partitionierende Clusterverfahren

Partitionierende Clusteralgorithmen zerlegen die Gesamtmenge der Daten in disjunkte Teilmengen. Das bekannteste Beispiel ist der sogenannte *K-means*-Algorithmus. Beim *K-means*-Algorithmus ist eine gewünschte Anzahl k von Clustern und eine Funktion zur Bestimmung des Mittelpunktes eines Clusters bekannt. Der Algorithmus läuft folgendermaßen ab:

1. Initialisierung: (Zufällige) Auswahl von k Clusterzentren.
2. Zuordnung: Jedes Objekt wird dem ihm am nächsten liegenden Clusterzentrum zugeordnet.
3. Neuberechnung: Es werden für jedes Cluster die Clusterzentren neu berechnet.
4. Wiederholung: Falls sich nun die Zuordnung der Objekte ändert, weiter mit Schritt 2, ansonsten Abbruch.

Dabei hängt das Endergebnis kritisch von der Anfangswahl der Zentroide sowie der Reihenfolge der Auswahl der Daten ab. Es ist jedoch erheblich schneller als hierarchische Verfahren und braucht weniger Speicherplatz. Typische Realisierungen benötigen einen Aufwand O(n).

U*C-Cluster-Algorithmus

Die ESOM selbst ist kein Cluster-Algorithmus, sondern ein Projektionsverfahren, bei dem Cluster in den hochdimensionalen Daten in koherenten Regionen auf der Karte abgebildet werden. Punkte innerhalb eines Clusters werden auch innerhalb dieses Bereichs abgebildet, während Punkte am Rand des Clusters auch auf den Rand des Clustergebiets auf der Karte abgebildet werden. Für das Clustern mit der ESOM ist der U*C-Cluster-Algorithmus entwickelt worden [Ultsch, 2005]. Lokale Distanz- und Dichteinformation werden dabei zum Clustern verwendet.

Algorithmus 1 U*C-Cluster-Algorithmus [Ultsch, 2005]

Gegeben: ESOM mit U-Matrix, P-Matrix, U*-Matrix, I = {}

Immersion:
for all Neuronen n einer ESOM **do**
    Gradientenabstieg auf der U-Matrix bis ein Minimum in Neuron u erreicht ist.
    Gradientenaufstieg auf der P-Matrix bis ein Maximum in Neuron p erreicht ist.
    I = I ∪ {p}; Immersion(n) = p.
end for

Clusterzuweisung:
Berechnung der Wasserscheiden [Luc and Soille, 1991] für die U*-Matrix.
Teile I mit Hilfe der Wasserscheiden in Cluster $C_1, \ldots, C_c$ auf.
Zuweisung von Datenpunkt x zum Cluster C_j wenn Immersion(bm(x)) $\in C_j$.

Aufbereitung der Eingabedaten

Für die Berechnung der Abstände zwischen den Vektoren beim ESOM-Training ist eine symmetrische Verteilung der Daten um den Mittelpunkt notwendig. Daher ist eine Aufbereitung der Rohdaten unerlässlich.

Grobe Ausreißer aus den Daten müssen entfernt werden, da sie eine Unterrepräsentation der Distanzen zwischen den übrigen Daten hervorrufen und so das Ähnlichkeitsmaß verzerren.

Zur Sichtung der Verteilungen der einzelnen Merkmale sind verschiedene Abbildungen möglich: Histogramme, *QQ-Plots* und *PDE-Plots* (PDE = Pareto Density Estimation) [Ultsch, 2003a]. Bei den Histogrammen wird der Wertebereich eines Merkmals in *Bins* fester Größe unterteilt und die Anzahl der Daten innerhalb eines festen *Bins* bestimmt. Diese Werte werden abgebildet. Die Häufigkeit der Daten in den *Bins* kann direkt an der Höhe der Rechtecke abgelesen werden. Bei den *QQnorm-Plots* werden die Quantile der Verteilung eines Merkmal gegen die Quantile der Standard-Normalverteilung aufgetragen. Ist diese Abbildung eine Gerade, so sind die beiden Verteilungen gleich, d.h. die Verteilung des Merkmals ist symmetrisch um den Mittelpunkt. Bei einer Krümmung unterscheiden sich die beiden Verteilungen voneinander. Bei den *PDE-Plots* wird die Datendichte an allen verschiedenen Datenpunkten der Datenmenge abgeschätzt [Ultsch, 2003a, 2004]. Hierzu werden für einen Punkt x alle Datenpunkte y mit $d(x, y) < r_p$ gezählt. d ist eine Distanzfunktion und r_p der Pareto-Radius (siehe Kapitel 2.1). Der Pareto-Radius r_p zur Abschätzung der Datendichte wird so gewählt, dass der Median der Datenpunkte in den Paretokugeln um alle Datenpunkte 22,13% der Datenpunkte ist. Um die Suche nach dem Pareto-Radius effizient zu gestalten, werden ausschließlich die Perzentile aller Abstände als mögliche Kandidaten für einen Pareto-Radius betrachtet.

Ist die Verteilung eines Merkmals schief, so ist eine nichtlineare Transformation der Werte dieses Merkmals notwendig, um eine symmetrische Verteilung um den Mittelpunkt zu erreichen. Bei linksschiefen Verteilungen werden stauchende Transformationen wie Wurzelung oder Logarithmieren verwendet. Bei rechtsschiefen Verteilungen sind streckende Transformationen wie z.B. Quadrieren oder Exponieren notwendig.

Nach der Entfernung von Ausreißern und der nichtlinearen Transformation ist noch eine lineare Transformation, wie z.B. die z-Transformation, der Werte der Merkmale notwendig, damit sie den gleichen Wertebereich haben und so die Gewichtung aller Merkmale bei der euklidschen Abstandsbestimmung während des ESOM-Trainings gleich ist. Auch das Eliminieren von Ausreißern ist sinnvoll, um eine Überbewertung dieser im Gegensatz zu den anderen Daten zu verhindern.

2.2. Mathematische Grundlagen

Rotation eines Objekts um eine beliebige Achse im $\mathbb{R}^3$

Die Rotation eines dreidimensionalen Objektes, bestehend aus Punkten im $\mathbb{R}^3$, ist die Multiplikation eines jeden Punktvektors mit einer Rotationsmatrix RM.

Definition 2.2.1 (Rotationsmatrix) *Sei α der Rotationswinkel, $c = \cos(\alpha)$, $s = \sin(\alpha)$ und $v = (v_x, v_y, v_z)$ eine beliebige Rotationsachse. Die Rotationsmatrix RM für eine Rotation um den Winkel α um die Rotationsachse v ist definiert als*

$$RM = \begin{pmatrix} c + ({v_x}^2)(1-c) & v_x v_y (1-c) - v_z s & v_x v_z (1-c) + v_y s \\ v_y v_x (1-c) + v_z s & c + ({v_y}^2)(1-c) & v_y v_z (1-c) - v_x s \\ v_z v_x (1-c) - v_y s & v_z v_y (1-c) + v_x s & c + ({v_z}^2)(1-c) \end{pmatrix}$$

Definition 2.2.2 (Rotation im $\mathbb{R}^3$) *Sei [x,y,z] ein Vektor im $\mathbb{R}^3$ und RM die Rotationsmatrix. Dann ist die Rotation definiert als*

$$(x', y', z')^T = RM \cdot (x, y, z)^T$$

Wavelet-Transformation von Oberflächen im $\mathbb{R}^3$

Sei S eine Oberfläche im $\mathbb{R}^3$ gegeben durch eine endliche Menge von Punkten. Die Oberfläche soll durch ein Dreiecksnetz dieser Punkte genügend gut approximiert sein, welches durch eine Delaunay-Triangulierung der zugrundeliegenden Menge von Oberflächenpunkten entsteht.

Die verschiedenen Iterationen der Wavelet-Transformation erreicht man durch Unterteilung dieses Dreiecksnetzes. Das Startnetz V_0 besteht nur aus wenigen Punkten, die die grobe Gestalt der Oberfläche widerspiegeln. Bei jeder Stufe der Wavelet-Transformation wird nun ein neuer Punkt m zwischen zwei Punkten v_1 und v_2, die zum selben Dreieck gehören, approximiert.

$$m = \frac{1}{2}(v_1 + v_2)$$

Die Abbildung 2.3 zeigt eine solche Unterteilung eines Gitters.

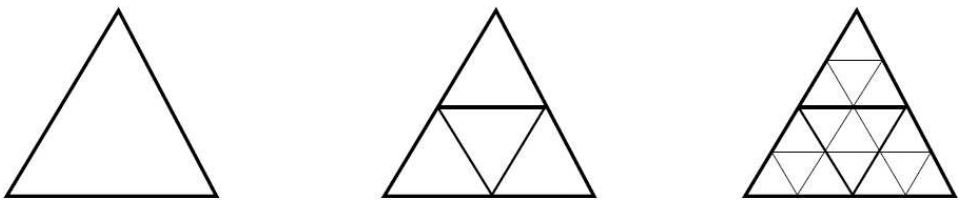
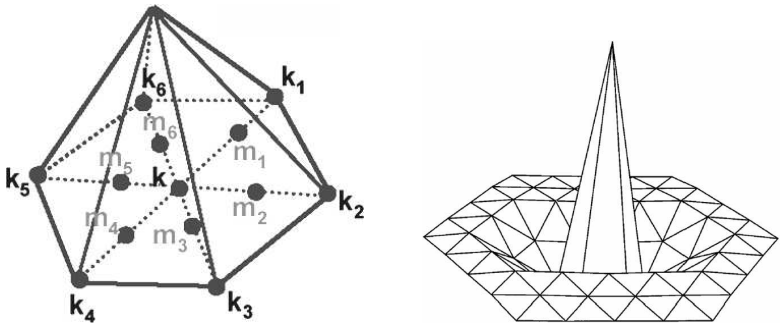


Abbildung 2.3.: Zwei Stufen der Unterteilung eines Dreiecksnetzes [Eck et al., 1995]

Auf den Punkten des Startnetzes V_0 sind die sogenannten Skalierungsfunktionen (siehe

Eck et al. [1995], Schröder and Sweldens [1995], Lounsbery et al. [1997]) definiert. Die zugehörigen Koeffizienten sind die Approximationskoeffizienten, die für alle Level der Wavelet-Transformation beibehalten werden. Sie sind die Koordinaten der Punkte des Gitters V_0 und liefern eine sehr grobe Beschreibung der Oberfläche.

Die Wavelets (siehe Eck et al. [1995], Schröder and Sweldens [1995], Lounsbery et al. [1997]) sind auf den approximierten Zwischenpunkten der verschiedenen Auflösungsstufen definiert. Die Abbildung 2.4 zeigt eine solche Skalierungsfunktion auf dem Gitterpunkt k und die dazugehörige Waveletfunktion auf einem approximierten Zwischenpunkt m. Dabei sind die k_i die benachbarten Original-Gitterpunkte und die m_i die approximierten Zwischenpunkte.



(a) Polyhedrale Skalierungsfunktion (b) Wavelet-Funktion

Abbildung 2.4.: Skalierungsfunktion auf einem Gitterpunkt k mit Nachbarpunkten k_i und zugehörige Waveletfunktion auf einem der approximierten Zwischenpunkte m_i [Lounsbery et al., 1997].

Die Wavelet-Koeffizienten w berechnen sich aus der Differenz zwischen dem approximierten Punkt m und dem Oberflächenpunkt v mit der geringsten Distanz zu dem approximierten Punkt (siehe Abbildung 2.5). Sie sind ein Maß dafür, wie stark die Oberfläche lokal von einer glatten Approximation abweicht. Sie kodieren die Details eines jeden Levels der Wavelet-Transformation.

Für das Netz V_{j+1} müssen nun die Punkte k_i des Gitters V_j noch geliftet werden, damit die Oberfläche für die nächste Iteration der Wavelet-Transformation glatt ist. Dazu wird zu jedem Punkt k eine Linearkombination der Wavelet-Koeffizienten w_i der N benachbarten Zwischenpunkte m_i addiert.

$$k_{neu} = k_{alt} + \frac{3}{4} \sum_{i=1}^{N} w_i$$

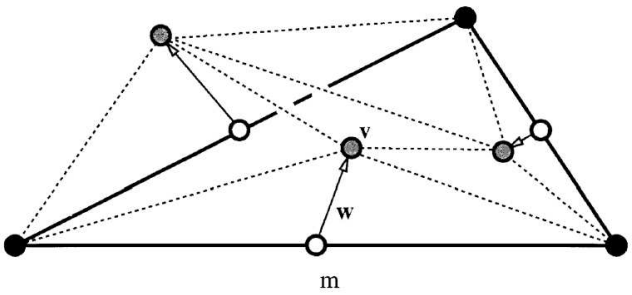


Abbildung 2.5.: Berechnung der Waveletkoeffizienten [Lounsbery et al., 1997]

V_{j+1} ist dann gleich $V_j \cup M_j$ mit M_j die Menge aller zu den approximierten Zwischenpunkten gehörenden Oberflächenpunkten zwischen den Punkten aus V_j.

Um die Wavelet-Koeffizienten miteinander vergleichen zu können, müssen diese noch skaliert werden, da sie von der Dichte des Gitters abhängen. Dazu wird ein Skalierungsfaktor berechnet. Bei großer Dichte der Gitterpunkte sind die Koeffizienten klein, da zwischen den Gitterpunkten keine großen Variationen der Oberfläche möglich sind. Der Skalierungsfaktor muss deshalb die Koeffizienten vergrößern. Bei geringer Dichte der Gitterpunkte tritt der gegenteilige Fall auf und die Koeffizienten müssen verkleinert werden.

Das die Oberfläche in der jeweiligen Resolutionsstufe approximierende Gitter entsteht durch Triangulierung der Gitterpunkte. Jeder neu approximierte Zwischenpunkt, welcher der Ort des Waveletkoeffizienten ist, liegt auf der Grundseite zweier angrenzender Dreiecke. Jalobeanu [2003] schlägt die folgende Berechnung des Skalierungsfaktors aus der Gestalt dieser Dreiecke vor (siehe Abbildung 2.6).

Definition 2.2.3 (Skalierungsfaktor) *Seien D_1, D_2 die beiden Dreicke, auf deren Kante m liegt. k_1, k_2 und k_{pi} seien die Punkte eines Dreiecks, m liege zwischen k_1 und k_2, h_i sei der Fußpunkt der jeweiligen Höhe von k_{pi} auf die Seite k_1k_2.*
Es sei $L = |k_2 - k_1|$, $l_i = |k_{pi} - m|$ und $\alpha_i = \measuredangle(h_i, k_{pi}, m)$, $i = 1, 2$. Der Skalierungsfaktor s ist dann $s = \frac{1}{2}(s_1 + s_2)$ mit

$$s_i = L\left(\frac{3}{4}\frac{L^2}{l_i^2}(cos(\alpha_i) + sin(\alpha_i))^2 + 4\right)^{-\frac{1}{2}}, \; i = 1, 2.$$

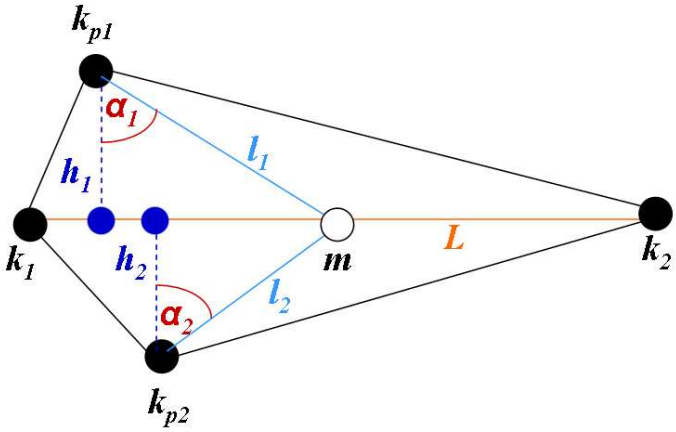


Abbildung 2.6.: Berechnung des Skalierungsfaktors

Wavelet-Transformation einer Oberfläche

Der Algorithmus der Wavelet-Transformation einer Oberfläche $S \subset \mathbb{R}^3$ setzt sich aus den vorher beschriebenen Schritten zusammen.

- **Initialisierung:**

  Starte mit einer Menge von Punkten V_0, die die grobe Struktur der Oberfläche wiedergeben. Die Approximationskoeffizienten sind die Koordinaten der Punkte $v \in V_0$.

- **Transformation von Level j zu Level j+1:**

  1. Berechne Menge von Zwischenpunkten M_j.
  2. Berechne Waveletkoeffizienten = Abstand des Zwischenpunktes von realem Oberflächenpunkt.
  3. Lifte die Punkte $v \in V_j$.
  4. Berechne die Skalierungsfaktoren für alle $m \in M_j$ und skaliere die einzelnen Wavelet-Koeffizienten.
  5. Berechne V_{j+1} als Vereinigung von der Menge der gelifteten Punkte V_j und der Menge der Oberflächenpunkte M_j.

2.3. Biologische Grundlagen

Proteine

Proteine[2] sind Makromoleküle, die aus Kohlenstoff, Wasserstoff, Sauerstoff und Stickstoff aufgebaut sind und auch andere Elemente wie Schwefel und Selen beinhalten können. Sie gehören zu den Grundbausteinen aller Zellen. Die Aufgaben der Proteine im Organismus sind vielfältig:

- Als Strukturproteine bestimmen sie den Aufbau der Zelle und damit letztlich die Beschaffenheit von Geweben, beispielsweise der Haarstruktur, und den gesamten Körperaufbau.
- Als Enzyme ermöglichen und beschleunigen sie chemische Reaktionen unter physiologischen Bedingungen.
- Als Ionenkanäle regulieren sie die Ionenkonzentration in der Zelle, und damit deren osmotische Homöostase sowie die Erregbarkeit von Nerven und Muskelzellen.
- In den Muskeln verändern bestimmte Proteine ihre Form und sorgen so für die Kontraktion der Muskeln und damit für Bewegung.
- Als Transportproteine übernehmen sie den Transport körperwichtiger Substanzen wie z.B. Hämoglobin, das im Blut für den Sauerstofftransport zuständig ist, oder Transferrin, das Eisen in unserem Blut transportiert.
- Manche (meist kleinere Proteine) steuern als Hormone Vorgänge im Körper.
- Als Antikörper dienen sie der Infektionsabwehr.

Für die Wirkungsweise der Proteine ist ihre räumliche Struktur besonders wichtig. Die Proteinstruktur lässt sich auf vier Betrachtungsebenen beschreiben:

- Als Primärstruktur eines Proteins wird die Abfolge der einzelnen Aminosäuren innerhalb der Polypeptidkette bezeichnet. Die Primärstruktur stellt lediglich die Aminosäurensequenz, jedoch nicht den räumlichen Aufbau dar.
- Als Sekundärstruktur bezeichnet man die räumliche Anordnung der Aminosäuren eines Proteins. Man unterscheidet dabei zwischen folgenden Strukturtypen:

[2] Aus http://de.wikipedia.org/wiki/Proteine.

Alpha-Helix, Beta-Faltblatt, Beta-Turn und ungeordnete, so genannte Random-Coil-Strukturen. Diese Raumstrukturen ergeben sich durch Wasserstoffbrückenbindungen, die sich zwischen den Peptidbindungen des Polypeptid-Rückgrates ausbilden.

- Die Tertiärstruktur ist eine der Sekundärstruktur übergeordnete räumliche Anordnung der Polypeptidkette. Sie wird von den Kräften und Bindungen zwischen den Resten (d. h. den Seitenketten) der Aminosäuren bestimmt. Als Bindungskräfte, die diese dreidimensionale Struktur stabilisieren, wirken beispielsweise Disulfidbrücken, (kovalente Bindungen zwischen den Schwefelatomen zweier Cysteinreste) oder vor allem nicht-kovalente Wechselwirkungen wie die zuvor genannten Wasserstoffbrückenbindungen. Zusätzlich spielen hydrophobe, ionische und van-der-Waals-Wechselwirkungen eine wichtige Rolle.

- Viele Proteine müssen sich, um funktionsfähig sein zu können, zu einem Proteinkomplex zusammenlagern, der so genannten Quartärstruktur. Dies kann entweder eine Zusammenlagerung von unterschiedlichen Proteindomänen sein oder ein Verband aus zwei oder mehr Polypeptidketten die aus ein und derselben Polypeptidkette, dem sog. Precursor, hervorgegangen sind (vgl.: Insulin). Dabei sind die einzelnen Proteine häufig durch Wasserstoffbrücken und Salzbrücken aber auch durch kovalente Bindungen miteinander verknüpft. Die einzelnen Untereinheiten eines solchen Komplexes werden als Protomere bezeichnet. Einige Protomere können ihre Funktion auch als eigenständige Proteine besitzen, aber viele erreichen ihre Funktionalität nur im Komplex. Als Beispiel für aus mehreren Proteinen zusammengelagerte Komplexe können die Immunglobuline (Antikörper) dienen, bei denen jeweils zwei identische schwere und zwei identische leichte Proteine über insgesamt vier Disulfidbrücken zu einem funktionsfähigen Antikörper verbunden sind.

EC-Klassifizierung der Proteine

Die EC-Klassifizierung beruht auf einer Einteilung der Enzyme nach ihrem Katalysemechanismus und wurde von einer internationalen Kommission vorgeschlagen. Der Klassifizierungscode setzt sich aus vier Zahlen zusammen. Die erste Ziffer steht für die Hauptklasse des Enzyms und gibt den Reaktionstyp an. Es existieren sechs solcher Hauptklassen:

- EC 1 - Oxidoreduktasen
- EC 2 - Transferasen
- EC 3 - Hydrolasen
- EC 4 - Lyasen
- EC 5 - Isomerasen
- EC 6 - Ligasen

Die zweite Ziffer gibt an, welche chemische Gruppe umgesetzt wird. Die dritte Ziffer steht für den beteiligten Kofaktor und die letzte Ziffer spezifiziert das entsprechende Enzym nach der katalysierten Reaktion. Enzyme derselben EC-Unterklasse, d.h. mit derselben EC-Nummer, katalysieren also die gleiche Reaktion und haben deswegen sehr ähnliche Bindetaschen. Die EC-Nummer 1.1.1.1 z. B. bedeutet:

- 1.-.-.-: Oxidoreduktase
- 1.1.-.-: Angriffspunkt sind CH-OH-Gruppen von Donoren
- 1.1.1.-: mit dem Kofaktor NAD(+) oder NADP(+) als Akzeptor
- 1.1.1.1: Alkoholdehydrogenase, katalysierte Reaktion:
  Alkohol + NAD(+) $\Leftrightarrow$ Aldehyd oder Keton + NADH

Die Proteinbindetaschen-Datenbank CAVBASE

Alle in der Datenbank CAVBASE befindlichen Bindetaschen sind mit dem Detektionsalgorithmus LIGSITE [Hendlich et al., 1997] aus den in der Datenbank RELIBASE [Hendlich et al., 2003] gespeicherten Proteinstrukturen berechnet worden. Die Vorgehensweise ist Folgende: Um das Protein, bei dem die Bindetaschen detektiert werden sollen, wird ein regelmäßiges Gitter gelegt. Zu Beginn werden alle Gitterpunkte als Lösungsmittel gekennzeichnet und erhalten den Wert 0. Alle Gitterpunkte, die dem Lösungsmittel nicht zugänglich sind, d.h. Gitterpunkte, die im Protein liegen, erhalten den Wert -1. Das Programm mustert dann entlang der Koordinatenachsen und der vier Raumdiagonalen die Gitterpunkte nach Regionen ab, die von beiden Seiten dieser Achsen vom Protein eingeschlossen sind. Für jede Achse, bei der die Bedingung erfüllt ist, wird der Zähler für den entsprechenden Gitterpunkt um eins erhöht. Nach dem Durchmustern haben alle Lösungsmittel-zugänglichen Gitterpunkte einen „Vergrabenheitswert“ zwischen 0 und 7, 0 bedeutet dabei keine Vergrabenheit und 7 bedeutet starke Vergrabenheit.

Alle Punkte mit einem Vergrabenheitswert ≥ 2 können potentiell zu einer Bindetasche gehören. Der Algorithmus startet mit einem Gitterpunkt, der einen Vergrabenheitswert ≥ 2 hat. Alle nächsten Nachbarn mit einem Vergrabenheitswert ≥ 2 werden der Region hinzugefügt, für jeden dieser Punkte werden dann auch wieder die nächsten Nachbarn überprüft usw., der Algorithmus stoppt, wenn alle benachbarten Gitterpunkte mit diesem Vergrabenheitswert der Region hinzugefügt wurden. Sind Gitterpunkte noch nicht berücksichtigt worden, die auch eine große Vergrabenheit haben, so wird der Vorgang für diese Punkte wiederholt. Um bedeutungslose, sehr kleine Vertiefungen herauszufiltern, wird zusätzlich noch ein unterer Grenzwert für die Anzahl zusammenhängender Gitterpunkte gesetzt. Regionen, die kleiner sind, werden nicht berücksichtigt. Dadurch bleiben Kavitäten, die kleiner als für eine Aufnahme von Wassermolekülen sind, unberücksichtigt.

Die Randpunkte einer solchen Vereinigungsmenge von Gitterpunkten beschreiben genähert die Oberfläche einer Tasche. Auch die benachbarten Aminosäuren, die die Tasche begrenzen, sind leicht zu identifizieren. Die Atomkoordinaten der Aminosäuren, die die Bindetasche begrenzen, werden auf einen Satz repräsentativer Pseudozentren reduziert. Diese Pseudozentren werden nach fünf Eigenschaften klassifiziert, die essentiell für die molekulare Erkennung sind: Wasserstoff-Donoren, Akzeptoren, ambivalente Donoren/Akzeptoren (Hydroxylgruppen oder Seitenkettenstickstoffe des Histidins), hydrophobe aliphatische und aromatische Eigenschaften. Die Pseudozentren werden außerdem noch dahingehend untersucht, ob sie ihre Eigenschaften auf die Bindetaschenoberfläche

ausrichten. Dieses Verhalten wird dann den benachbarten Gitter-Oberflächenpunkten zugewiesen. Die Form der Bindetasche, die zugeordneten Deskriptoren, deren Oberflächenzugänglichkeit und die zugewiesenen Oberflächenbereiche werden zusammen mit Informationen über den ggf. gebundenen Liganden in CAVBASE abgespeichert [Schmitt et al., 2002].

Alle in dieser Arbeit verwendeten Proteinbindetaschen stammen aus der Datenbank CAVBASE, die ungefähr 120000 Taschen enthält. Sie sind charakterisiert durch diese Menge von Pseudozentren und durch eine Menge von Gitterpunkten, die eine genaue Approximation der Oberfläche der Bindetasche darstellen und mit den physikochemischen Eigenschaften des zugehörigen Pseudozentrums versehen sind. Beide Mengen werden durch die Koordinaten der einzelnen Punkte sowie deren physikochemischen Eigenschaften charakterisiert. Den Pseudozentren ist zudem noch ein Wirkvektor, der die Wirkrichtung der physikochemischen Eigenschaft des Pseudozentrums angibt, den Oberflächenpunkten ist noch ein Vergrabenheitswert zugeordnet, der angibt, wie tief der Punkt im Protein liegt. Eine Bindetasche enthält im Schnitt 80 Pseudozentren und 3000 Oberflächenpunkte.

3. Ansätze zur Klassifizierung von Proteinen

In der Literatur finden sich zahlreiche Verfahren, mit denen Proteine bzw. deren Bindetaschen verglichen werden können. Dabei unterscheidet man zwischen verschiedenen Verfahren. Eine Gruppe sind Sequenz-basierte Ansätze, die einen Vergleich der Proteine auf Basis der Aminosäuresequenzen vornehmen. Sekundärstrukturelement-basierte Ansätze sind Verfahren, die einen Vergleich mit Hilfe der Art und Anordnung der Sekundärstrukturelemente des Proteins berechnen. Die dritte Gruppe sind solche Methoden, die die dreidimensionale Struktur des Proteins bzw. seiner Bindetasche vergleichen.

Für den Vergleich auf Basis der Aminosäuresequenz sind eine Reihe von Algorithmen entwickelt worden, die möglichst große zusammenhängende, identische oder ähnliche Abschnitte in den Aminosäureabfolgen der Proteine ermitteln [Needleman and D., 1970, Waterman, 1984, Taylor and Orengo, 1989]. Daraus sind eine Reihe von Programmen wie FASTA [Pearson and Lipman, 1988, Pearson, 1990] sowie verschiedene Datenbanken wie SWISS-PROT [Bucher and Bairoch, 1994, Boeckmann et al., 2003] entstanden, die schnelle Sequenzvergleiche auf Basis dieser Algorithmen erlauben. In der Regel bedeutet hohe Sequenzhomologie auch hohe strukturelle Homologie. Die Methoden sind aber nicht in der Lage, ähnliche Faltungsmuster, die durch unterschiedliche Aminosäureabfolgen entstehen, zu detektieren.

Sekundärstrukturelement-basierte Ansätze sind Methoden, die Proteine anhand von Art und Anordnung der Sekundärstrukturelemente klassifizieren. Wichtige Klassifizierungsschemata sind SCOP [Murzin et al., 1995, Lo Conte et al., 2002] und CATH [Orengo et al., 1997]. Davon setzen sich die ENZYME-Datenbank [Bairoch, 2000], die Datenbank BRENDA [Schomburg et al., 2002] und die Datenbank MEROPS [Rawlings et al., 2006] ab, die Proteine anhand der katalysierten Reaktion einteilen. Insbesondere bei Enzymen besteht keine zwingende Korrelation eines bestimmten Faltungsmusters mit einer bestimmten Funktion [Anantharaman et al., 2003, Martin et al., 1998, Nagano et al., 2002, Orengo et al., 2001, Todd et al., 2001].

Im Folgenden werden die in der Literatur vorgestellten Verfahren zum Vergleich der 3D-Strukturen von Proteinen bzw. ihren Bindetaschen, die unabhängig von der Aminosäuresequenz sind, erläutert.

3.1. Modellierung der Proteinstrukturen anhand der Aminosäuren

Bei den Vergleichen der 3D-Strukturen von Proteinen gibt es verschiedene Ansätze, die Struktur zu modellieren. Eine Reihe von Verfahren verwendet dreidimensionale Template, z.B. die katalytische Triade bei den Serinproteasen, zur Mustersuche. Eine sehr häufig verwendete Methode ist es, nur die C_α-Atome der Aminosäuren zu betrachten und mit dieser vereinfachten Struktur eine Ähnlichkeitssuche durchzuführen. Zudem gibt es Methoden, die pro Aminosäure mehrere Pseudoatome, teilweise spezifiziert durch Vektoren, zur Beschreibung der Struktur verwenden. Eine weitergehende Modellierung ist die Heranziehung von chemischen Eigenschaften und deren Anordnung zur Charakterisierung der Rezeptorstruktur, die bei den vorher genannten Modellierungen überhaupt nicht berücksichtigt wurden.

Strukturvergleiche anhand von Distanzmatrizen

Das Programm DALI von Holm & Sander vergleicht Proteinstrukturen auf Basis intramolekularer C_α-C_α-Abstände [Holm and Sander, 1994, 1996a,b, Holm, 1998]. Für jedes Protein wird eine Distanzmatrix berechnet, in der alle intramolekularen C_α-C_α-Abstände gespeichert werden. Zum Vergleich zweier Proteine wird dann nach ähnlichen Untermatrizen einer bestimmten Größe in den jeweiligen Distanzmatrizen gesucht. Die einzelnen Untermatrizen werden dann zu größeren Matrizen kombiniert. Die verschiedenen Kombinationen von Untermatrizen implizieren unterschiedliche Überlagerungsmöglichkeiten der jeweiligen zugrundeliegenden Proteinstrukturen, wobei die Güte dieser Überlagerungen anhand des Ähnlichkeitsmaßes der Untermatrizen bewertet wird. Über eine Monte-Carlo-Routine werden die einzelnen Matrizen so miteinander verknüpft, dass damit jeweils eine Verbesserung der Bewertung der daraus resultierenden Überlagerung verbunden ist.

Lessel & Schomburg haben einen vergleichbaren Ansatz entwickelt [Lessel and Schomburg, 1994]. Auch hier werden Distanzmatrizen intramolekularer C_α-C_α-Abstände für jedes Protein berechnet und dann nach ähnlichen Untermatrizen einer vorgegebenen

Mindestgröße gesucht. Diese ähnlichen Untermatrizen werden anschließend sukzessive so lange erweitert, bis ein bestimmter Grenzwert erlaubter mittlerer Distanzabweichungen überschritten wird. Für den Vergleich werden jeweils zwei solcher maximal großen Untermatrizen miteinander kombiniert. Die Optimierung der Überlagerung wird hier nicht durch Monte-Carlo-Methoden, sondern durch ein einfaches iteratives Überlagerungsverfahren erreicht.

Einsatz von Graph-Algorithmen zum Tertiärstrukturvergleich

Grindley et al. [1993] beschreiben ein Verfahren, bei dem jede der zu vergleichenden Proteinstrukturen als Graph betrachtet wird. Die Sekundärstrukturelemente des Proteins werden durch einen Vektor beschrieben, der Start-, Mittel- und Endpunkt des Strukturelements enthält. Diese Vektoren sind durch Distanz- und Winkelabhängigkeiten zueinander angeordnet und bilden die Knoten der Graphen. Aus den Graphen dieser beiden Objekte wird ein neuer Graph gebildet, indem die Knoten aus Paaren von Knoten der beiden Objekte erzeugt werden. Zwei solche Knoten sind dann verbunden, wenn die beiden Strukturelemente ähnliche Distanz- und Winkelverhältnisse zueinander haben, also wenn die geometrische Lage zweier solcher Elemente in beiden Proteinen ähnlich ist. In diesem neuen Graphen sucht man nun nach dem größten gemeinsamen Subgraphen (Clique). Projiziert man diesen Subgraphen auf die jeweiligen Objekte zurück, so erhält man Paare von Knoten, die in beiden Objekten auf die gleiche Weise verknüpft sind.

In dem Programm ASSAM [Artymiuk et al., 1994] werden die Seitenketten der Aminosäuren durch zwei Pseudoatome, die den funktionellen Teil der Aminosäure beschreiben, repräsentiert. Ein Seitenkettenmotiv setzt sich aus mehreren Aminosäuren und einer Distanzmatrix zwischen allen Pseudoatomen zusammen und umfasst typischerweise drei Aminosäuren. Zusätzlich können zur Definition eines Seitenkettenmotivs einige generische Aminosäurentypen definiert werden (sauer, basisch, Glu/Gln, etc.). Das Protein wird als ein gewichteter Graph mit den Pseudoatomen als Knoten (ausgestattet mit dem Aminosäuretyp) und den Distanzen zwischen den einzelnen Pseudoatomen als Kanten dargestellt. Ein Ullmann-Subgraph Algorithmus wird verwendet, um alle Subgraphen zu permutieren und um so die An- oder Abwesenheit eines bestimmten Musters in einem Graphen zu erkennen. Spriggs et al. [2003] haben einige Verbesserungen am ursprünglichen Verfahren vorgenommen. So verfügt ASSAM nun über eine größere Zahl von generischen Aminosäuretypen, die allgemeinere Suchen möglich machen. Es werden u. a. nur solche Aminosäuren als ähnlich angesehen, die eine vergleichbare Distanz zu

einer Bindestelle (definiert durch die Anwesenheit von einem Heteroatom) besitzen, aus einem vergleichbaren lokalen Faltungsmotiv entstammen, oder eine ähnliche Lösungsmittelzugänglichkeit aufweisen.

Russell [1998] stellt ein Verfahren vor, bei dem die Aminosäuren durch ihre C_α-, C_β-Atome und ein an der Seitenkette lokalisiertes „funktionelles Atom“ dargestellt werden. Diese Modellierung wird von Wangikar et al. in dem Programm DRESPAT aufgenommen, mit dem eine Ähnlichkeitssuche mit Hilfe des Clique-Algorithmus in SCOP-Datenbanken durchgeführt werden kann [Wangikar et al., 2003].

In dem Verfahren von Schmitt et al. [2002] werden die physikochemischen Eigenschaften der die Bindetasche des Proteins begrenzenden Aminosäuren in ihrer räumlichen Anordnung durch sogenannte Pseudozentren modelliert. Jede Bindetasche wird dann als Graph dargestellt, bei der die Pseudozentren die Knoten und die Distanzen zwischen den einzelnen Pseudozentren die Gewichte der Kanten darstellen. Aus den Graphen dieser beiden Objekte wird ein neuer Graph gebildet, indem die Knoten aus Paaren von Knoten der beiden Objekte erzeugt werden. Dann wird der größte gemeinsame Subgraph bestimmt. Eine Weiterentwicklung des Verfahrens von Weskamp et al. [2004] bringt durch einen Filterschritt und Indizierung eine signifikante Geschwindigkeitsverbesserung für den Vergleich großer Datenmengen.

In dem Verfahren von Jambon et al. [2003] werden Aminosäuren durch einen Satz funktioneller chemischer Gruppen repräsentiert, hydrophobe Aminosäuren werden nicht berücksichtigt. Ein Protein wird als Graph aus Dreiecken, die jeweils chemische Gruppen repräsentieren, dargestellt. Eine heuristische Suche erlaubt das Auffinden von ähnlichen Substrukturen in zwei Proteinen.

3D-Strukturvergleich mittels Geometric Hashing

Die zweite Gruppe an Verfahren verwendet *Geometric Hashing* zum Vergleich der 3D-Struktur von Proteinen. Der Algorithmus wird in zwei Stufen eingeteilt, die Repräsentationsphase und die Erkennungsphase. In der Repräsentationsphase werden drei Stellen im Protein dazu verwendet, ein neues Koordinatensystem zu definieren. Der Rest des Proteins wird dann in diesem neuen Koordinatensystem beschrieben. Gehören die das Koordinatensystem aufspannenden Punkte zu einer gemeinsamen Substruktur zweier Proteine, so sollten die äquivalenten Stellen im Protein ähnliche neue Koordinaten haben. Dabei müssen alle möglichen Koordinatensysteme miteinander verglichen werden. Die Repräsentation einer Proteinstruktur wird durch eine *Hash*-Tabelle realisiert. Diese Tabelle wird für jede Proteinstruktur eines Datensatzes erstellt, so dass relativ einfach

neue Strukturen hinzugefügt werden können. In der Erkennungsphase wird nun die zu vergleichende Proteinstruktur auf die gleiche Weise repräsentiert und mit den abgelegten Strukturen in der Datenbank verglichen.

In dem Verfahren PNAS wird die 3D-Struktur des Proteins vereinfacht durch die Koordinaten der C_α-Atome beschrieben [Nussinov and Wolfson, 1991]. Durch beliebige Wahl dreier C_α-Atome kann man nun ein neues Koordinatensystem aufspannen und alle verbliebenen C_α-Atome mit diesem neuen Koordinatensystem beschreiben. Eine Weiterentwicklung des Verfahrens von Leibowitz et al. erlaubt auch multiple Vergleiche der Proteinstrukturen [Leibowitz et al., 1999, 2001].

In dem Programm Tess von Wallace et al. [1997] werden die Aminosäuren der Proteine durch drei Punkte definiert, die dann in einer *Hash*-Tabelle abgelegt werden. Fischer et al. [1995] verwenden *Connolly*- und *Sphegen*-Oberflächenpunkte sowie C_α-Atome zur Beschreibung der Proteine und verwenden *Geometric Hashing* zum Vergleich.

Verwendung genetischer Algorithmen zur Ähnlichkeitserkennung

Lehtonen et al. [1999] benutzen in ihrem Programm GENFIT genetische Algorithmen zum Vergleich von Proteinstrukturen. Dabei wird zunächst eine Auswahl zufälliger Rotationsmatrizen und Translationsvektoren generiert, anhand derer zwei Proteinstrukturen überlagert werden. Jede Überlagerung wird analysiert und bewertet. Dafür werden Paare benachbarter C_α-Atome gesucht und so gefiltert, dass jedes C_α-Atom der einen Struktur maximal einem äquivalenten C_α-Atom der anderen Struktur entspricht. Es werden nur größere Teilstücke der jeweiligen zugrundeliegenden Aminosäuresequenz für die Überlagerung berücksichtigt. Die Bewertung der Überlagerung geschieht dann anhand der intermolekularen C_α-C_α-Abstände der Paare. Die gut bewerteten Überlagerungen überleben eine Generation im genetischen Algorithmus, so dass sukzessive die Qualität der Überlagerung verbessert wird.

Vergleich von 3D-Strukturen mit Hilfe von ESOM

Kupas et al. [2004a,b], Kupas and Ultsch [2005] haben ein weiteres Verfahren zur Klassifizierung von Substrukturen mit Hilfe von emergenten SOM entwickelt (Details siehe Appendix C). Der Algorithmus baut auf dem Pseudozentren-Konzept von Schmitt et al. [2002] auf. Jede Bindetasche wird systematisch in Substrukturen bestehend aus vier benachbarten Pseudozentren zerlegt. Die Verbindungslinien zwischen den vier Pseudozentren bilden eine Pyramide mit dreieckiger Grundfläche. Diese Substrukturen wer-

den sowohl durch die physikochemischen Eigenschaften der vier Pseudozentren als auch durch ihre geometrischen Eigenschaften charakterisiert. Mit den geometrischen Merkmalen Höhe, Schiefe und Maßzahl der Grundfläche der Pyramide wird eine emergente selbstorganisierende Merkmalskarte antrainiert. Mit dieser ESOM wird eine Distanz-und Dichte-basierte Clusterung der geometrischen Eigenschaften der Substrukturen durchgeführt. Alle Cluster werden anschließend in Subcluster gemäß der physikochemischen Eigenschaften der vier Pseudozentren der Substrukturen unterteilt, so dass nur solche mit passenden chemischen Eigenschaften in einem Cluster liegen.

Alle Substrukturen in einem Cluster haben ähnliche phaysikochemische und geometrische Eigenschaften und stellen ein *Match* für die zugehörigen Bindetaschen dar. Jeweils zwei solcher ähnlichen Substrukturen desselben Clusters werden überlagert, um die Ähnlichkeit der Bindetaschen zu bewerten. Aus dieser Überlagerung ergibt sich eine Koordinatentransformation, die auf die beiden zugehörigen Bindetaschen angewandt wird. Dann werden alle Paare von Pseudozentren bzw. Oberflächenpunkten gezählt, die die gleiche physikochemische Eigenschaft besitzen und einen Abstand kleiner als 1 Å haben. Die Anzahl der Paare ist der *Score* für diese beiden Substrukturen, es werden nur die Paare von Bindetaschen weiter betrachtet, die mindestens neun gemeinsame Pseudozentren aufweisen.

Durch die Clusterung der Substrukturen muss die aufwändige Überlagerung nur noch für solche Taschen durchgeführt werden, die mindestens eine gemeinsame Substruktur besitzen. Die entsprechende Koordinatentransformation ist durch die Substrukturen vorgegeben und muss nur für vier Pseudozentren dieser ermittelt werden. Alle unbedeutenden Matches werden durch den Filterschritt, der alle die *Matches* mit einem Score kleiner als neun Pseudozentren verwirft, nicht berücksichtigt.

Die Methode ist auf einen Datensatz von 109 Bindetaschen zu 13 verschiedenen EC-Nummern angewandt worden, für den bekannt ist, dass Bindetaschen von Enzymen zu gleicher EC-Nummer sehr ähnlich sind, während solche von Enzymen zu unterschiedlichen EC-Nummern keine Gemeinsamkeiten aufweisen. Durch die Clusterung und die anschließende Überlagerung sind nur solche Paare von Bindetaschen übrig geblieben, die zur gleichen EC-Nummer gehören und damit sehr große fast identische Bereiche haben. Ähnliche Substrukturen aus Bindetaschen von Enzymen zu unterschiedlichen EC-Nummern sind durch den Filterschritt eliminiert worden, da sie außer der zufälligen Ähnlichkeit der Substrukturen keine weiteren Gemeinsamkeiten besitzen.

Allerdings ist die Substruktur aus vier Pseudozentren sehr klein, so dass viele solcher zufälligen Ähnlichkeiten auftreten. Alleine aufgrund der Clusterung ist es nicht möglich,

eine Aussage über die Ähnlichkeit der zugehörigen Bindetaschen zu machen. Die rechenintensive Überlagerung der Bindetaschen muss also in jedem Fall durchgeführt werden, was bei sehr großen Datensätzen problematisch ist.

3.2. Modellierung der Lösungsmittel-zugänglichen Oberfläche des Proteins bzw. der Bindetasche

Ausserdem gibt es Verfahren, die die Lösungsmittel-zugängliche Oberfläche der Proteine modellieren und mit diesen Deskriptoren nach Ähnlichkeiten in den Rezeptorstrukturen suchen.

Rosen et al. [1998] verwenden als Deskriptoren so genannte *Sparse Critical Points*, die die *Connolly*-Oberfläche der Bindetasche eines Proteins beschreiben. Jede *Connolly*-Fläche, bestehend aus einer Menge von Oberflächenpunkten, wird durch ein Tripel bestehend aus einem einzelnen Oberflächenpunkt, der zugehörigen Oberflächennormalen und der Größe der *Connolly*-Fläche ersetzt. Damit wird die Anzahl an Oberflächenpunkten erheblich reduziert, um einen schnellen Vergleich zu ermöglichen. Als Algorithmus zum Vergleich der Proteine wird *Geometric Hashing* verwendet.

Pickering et al. [2001] benutzen einen *Clique* Algorithmus, um Proteinoberflächen miteinander zu vergleichen. Als Bindestellen werden Bereiche in einem bestimmten Radius um den Liganden definiert. *Connolly*-Oberflächenpunkte werden zur Beschreibung der Proteinoberfläche verwendet und werden mit ihrer Kurvigkeit und einem Maß für ihre Gestalt ausgestattet.

Kinoshita et al. haben eine Datenbank von Proteinoberflächen (eF-site) aufgebaut, die zur Ähnlichkeitssuche benutzt werden kann [Kinoshita and Nakamura, 2001, 2003]. Auch hier werden die *Connolly*-Oberflächenpunkte verwendet. Für jeden Punkt auf der Oberfläche eines Proteins werden das elektrostatische Potential, ein Hydrophobizitätswert und Attribute für die Flexibilität abgespeichert. Ein *Clique*-Algorithmus sucht nach ähnlichen Bereichen von Oberflächen, an denen die Oberflächenpunkte gleiche Eigenschaften aufweisen. Bindetaschen werden als Bereich im Radius von 5 Å um einen Liganden definiert.

Poirrette et al. [1997] verwenden genetische Algorithmen zum Vergleich der *Connolly*-Oberflächen zweier Proteinstrukturen. Grundlage des Vergleichs bilden Informationen über die Form der Oberfläche, Oberflächennormalen und Deskriptoren für die Wasserstoffbrückenbindungen.

Oshiro and Kuntz [1998] charakterisieren Bindetaschen durch einen Negativdruck der

Taschenoberfläche. Dazu wird die Oberfläche mit einem Satz von Kugeln aufgefüllt. Das Zentrum einer jeden Kugel repräsentiert dabei die mögliche Lage eines Ligandenatoms. Die Kugeln werden mit chemischen und energetischen Informationen versehen und können dazu verwendet werden, Unterschiede in den Bindetaschen einer Familie von Enzymen zu detektieren.

In dem Verfahren von Mason et al. [2004] wird die Oberfläche durch das Auffüllen der Bindetasche durch Kugeln mit Radius 1.3 Å modelliert. Diese Kugeln werden in innere Kugeln ohne Kontakt zur Proteinberfläche und in Oberflächenkugeln unterschieden. Um die Kugeln werden Würfel beschrieben, so dass eine zusammenhängende Gitteroberfläche entsteht. Jeder Würfel wird mit den physikochemischen Eigenschaften der umliegenden Atome versehen. Dieses Bindevolumen wird dann mit einem Basis-Set, generiert aus bekannten, typischen Bindestellen, verglichen und das ähnlichste ausgewählt. Zum Vergleich zweier Bindetaschen werden dann die zugehörigen *Bestmatches* aus dem Basis-Set miteinander verglichen.

Stahl et al. [2000] berichteten über die automatische Gruppierung von 176 Zinkmetalloproteinasen mit Hilfe eines selbstorganisierenden Neuronalen Netzes. Die Detektion von Taschen erfolgt mit einem dem LIGSITE-Algorithmus [Hendlich et al., 1997] sehr ähnlichen Verfahren. Für jede Tasche wird die lösungsmittelzugängliche Oberfläche berechnet. Jedem dieser Oberflächenpunkte wird die auf diesen Oberflächenpunkt gerichtete physikochemische Eigenschaft der die Tasche begrenzenden Aminosäuren zugewiesen. Die Deskriptoren der Taschen sind so genannte Kreuzkorrelationsvektoren. Dazu werden alle Paare von Punkten, die dichter als 15 Å liegen, betrachtet. Die Entfernung von 15 Å wird in zehn gleichgroße *Bins* unterteilt. Diese zehn *Bins* werden dann nochmal in fünfzehn *Bins* gemäß der möglichen Paare von physikochemischen Eigenschaften der Oberflächenpunkte unterteilt. Diese Unterteilung liefert einen 150-dimensionalen Vektor mit Elementen CV_d^T, jedes dieser Vektorelemente ist die Anzahl von Paaren von Oberflächenpunkten, die in das *Bin d* fallen und die physikochemischen Eigenschaften spezifiziert durch T haben. Für jede detektierte Tasche wird solch ein Kreuzkorrelationsvektor berechnet. Mit diesen 150-dimensionalen Vektoren wird dann eine toroide SOM mit 20x20 Neuronen und euklidischen Distanzen trainiert. Der Ansatz separiert die katalytischen Zentren dieser Enzyme von anderen Kavitäten auf deren Oberfläche und Enzyme mit der gleichen EC-Nummer sind zueinander gruppiert. Der Ansatz vergleicht aber nur komplette Taschen auf der Proteinoberfläche. Er dient vornehmlich dazu, aktive Zentren zu detektieren und von nicht funktionalen Taschen zu trennen. Zudem werden keine direkten geometrischen Deskriptoren verwendet, sondern nur die verschiedenen

Abstände zwischen Paaren von Oberflächenpunkten betrachtet und die Anzahl derer ermittelt, die in einem gewissen Intervall liegen.

Die Gruppe um R. Nussinov und H. Wolfson hat das Verfahren *SiteEngine* entwickelt [Shulman-Peleg et al., 2004]. Sie verwenden das Pseudozentrum-Konzept von CAVBASE, in dem sie die physikochemischen Eigenschaften durch 3D-Deskriptoren repräsentieren. *SiteEngine* vergleicht sowohl Proteinbindetaschen als auch komplette Proteinstrukturen. Für die Repräsentation der Proteineigenschaften werden die Aminosäuren zuerst in Pseudozentren übersetzt und die Eigenschaft jedes Pseudozentrums auf die umliegenden Oberflächenpunkte projiziert. Dabei wird jedoch die Direktionalität der Wasserstoffbindung nicht berücksichtigt. Für jeden Oberflächenbereich wird der Schwerpunkt („Oberflächenpseudozentrum“) bestimmt und ein Deskriptor für die Krümmung der Oberfläche berechnet. Während des Vergleichs zweier Bindetaschen/ Proteinstrukturen werden alle Triplets von drei Pseudozentren überlagert. In einem sehr schnellen Bewertungsschritt werden nur die Überlagerungen berücksichtigt, bei denen die Oberflächenschwerpunkte in der Überlagerung nahe zusammen fallen, eine ähnliche physikochemische Umgebung und eine vergleichbare Oberflächenkrümmung aufweisen. Die 5000 besten Überlagerungen werden anschließend nach ihren RMSD-Abweichungen geclustert und die beste Überlagerung jedes Clusters in einem zweiten aufwändigeren Bewertungsschritt genauer untersucht. Dazu muss nicht nur der Oberflächenschwerpunkt, sondern jeder einzelne Oberflächenpunkt den Anforderungen des ersten Bewertungsschrittes entsprechen. Für die 100 besten Überlagerungen wird mit Hilfe eines Graphalgorithmus die maximale eins-zu-eins Zuordnung von Pseudozentren zweier Bindetaschen gefunden, die deren RMSD-Abweichung und den Überlappungsgrad von hydrophoben und aromatischen Oberflächenbereichen berücksichtigt. Ein Vorteil der Methode ist, dass man sowohl mit kompletten Proteinstrukturen als auch mit Bindetaschen Datenbanksuchen durchführen kann. *SiteEngine* benutzt aber keine Methode zur automatischen Detektion von Bindestellen, diese werden über die Präsenz eines zuvor definierten Liganden zugewiesen (Proteinatome im 4 Å Radius um die Ligandenoberflächen).

Ein weiteres Verfahren sind so genannte *Shape Histograms* zur Beschreibung der Gestalt von Proteinstrukturen [Kastenmüller et al., 1998, Ankerst et al., 1999a,b]. Die Proteinstruktur wird dazu in ein kanonisches Koordinatensystem rotiert. Dann wird eine Kugel um das Protein gelegt und diese spinnennetzartig in Sektoren aufgeteilt. Für jeden Sektor wird die Anzahl an Oberflächenpunkten des Proteins ermittelt und diese Anzahlen werden in einem Histogramm aufgetragen. Ähnlichkeiten zwischen Proteinen werden über ein gewichtetes Distanzmaß der *Shape Histograms* ermittelt. Das Distanz-

maß bezieht benachbarte Sektoren in die Distanzberechnung mit ein, so dass die Methode robuster gegenüber Flexibilität der Proteine ist.

In einem Ansatz von Morris et al. werden Kugelflächenfunktionen zur Beschreibung der geometrischen Gestalt von Proteinbindetaschen verwendet [Morris et al., 2005]. Die von ihnen verwendeten Bindetaschen sind mit dem Programm SURFNET [Laskowski, 1995] detektiert worden. Jede Bindetasche wird vorher in ein kanonisches Koordinatensystem transformiert, um eine Vergleichbarkeit zu gewährleisten. Die Oberfläche der Bindetasche wird als eine Funktion der Kugelkoordinaten θ und ρ betrachtet. Für diese Funktion wird dann eine Darstellung als Linearkombination von Kugelflächenfunktionen berechnet. Die Deskriptoren sind die Expansionskoeffizienten der Kugelflächenfunktionen. Zum Vergleich der Bindetaschen werden die Vektoren mit den Expansionskoeffizienten verglichen, sind diese ähnlich, so ist auch die Gestalt der Bindetaschen ähnlich. Der Ansatz ist in der Lage, effizient große Datenmengen zu analysieren, da nur ein Vergleich der Koeffizientenvektoren durchgeführt werden muss und keine rechenintensive Überlagerung der Bindetaschen notwendig ist. Er hat jedoch auch einige entscheidende Nachteile:

- Alle Koeffizienten sind globale Koeffizienten für die gesamte Bindetasche, sie beinhalten keinerlei Ortsinformation. Somit ist es auch nicht möglich, nur lokale Ähnlichkeiten zwischen zwei Taschen zu detektieren.

- In den Vergleich geht nur die Form der Bindetasche ein. Physikochemische Eigenschaften werden überhaupt nicht berücksichtigt, obwohl sie essentiell für die Bindung eines Liganden sind.

- Durch die Betrachtung von globalen Koeffizienten ohne die Möglichkeit, diese auf lokale Oberflächeneigenschaften der Tasche zurückzuführen, ist es auch nicht möglich, Ähnlichkeiten bei konformativer Flexibilität korrekt zu detektieren. Dieses Problem wird umgangen, indem Koeffizienten für alle möglichen Konformationen berechnet werden. Angesichts des großen Wachstums der Datenmenge wird aber auch hier bald eine Grenze erreicht sein, so dass sich das Problem nicht mehr durch die Berechnung eines großen Satzes von Koeffizienten pro Tasche lösen lässt.

- Eine Voraussetzung für die Verwendung der Kugelflächenfunktionen zur Beschreibung der Oberfläche ist, dass diese sternförmig ist. Morris et al. behaupten, dass diese Eigenschaft für Bindetaschen meist gegeben sei, es ist aber fraglich, ob dies gesichert ist.

Der in dieser Arbeit beschriebene Ansatz verfolgt, im Gegensatz zu schon bestehenden Methoden zur Klassifizierung von Proteinen nach der Funktion, die Idee, kleine Teilbereiche in den Bindetaschen zu vergleichen. Dazu werden die in der Datenbank CAVBASE gespeicherten Bindetaschen in Substrukturen zerlegt. Diese Substrukturen sollen zum einen über geometrische und zum anderen über physikochemische Eigenschaften charakterisiert werden. Auf einer emergenten selbstorganisierenden Merkmalskarte (ESOM) werden die Deskriptionsvektoren dieser Substrukturen, generiert aus dem gesamten in CAVBASE gespeicherten Datenbestand, entsprechend ihrer Ähnlichkeit bzw. Verschiedenheit angeordnet. Eine sowohl Distanz- als auch Dichte-basierte Clusterung dieser Substrukturen wird durchgeführt. Zwei Substrukturen innerhalb desselben Clusters haben ähnliche geometrische und physikochemische Eigenschaften und weisen auf eine Gemeinsamkeit der beiden zugehörigen Bindetaschen hin. Mit dieser Methode ist es möglich, einen großen Datenbestand von Bindetaschen in einem Schritt auf Ähnlichkeiten sowohl in ihrer Geometrie als auch in ihren exponierten physikochemischen Bindungseigenschaften zu untersuchen. Durch die Zerlegung in Substrukturen können auch nur lokale Gemeinsamkeiten der Bindetaschen detektiert werden.

3.3. Zusammenfassung der Ansätze zum Vergleich von Proteinstrukturen

Es existieren einige Ansätze, Proteine über diverse Modellierungen ihrer lösungsmittelzugänglichen Oberflächen zu charakterisieren und zu vergleichen. Meist beziehen sich diese jedoch auf gesamte Proteinstrukturen. Der für das Wirkstoffdesign interessante Bereich der Bindestelle eines Liganden ist jedoch nur ein kleiner Teil der Rezeptoroberfläche. Es gibt nur wenige Methoden, die die Bindetaschen der Proteine miteinander vergleichen. Diese erfordern entweder eine rechenintensive Überlagerung der Bindetaschen [Shulman-Peleg et al., 2004] oder sie dienen ausschließlich zur Trennung der aktiven Zentren von unbedeutenden Vertiefungen auf der Oberfläche [Stahl et al., 2000]. Andere Ansätze sind nur in der Lage, globale Ähnlichkeiten der Taschen zu detektieren und berücksichtigen keine physikochemischen Deskriptoren [Morris et al., 2005]. Zudem ist es bei den meisten Methoden nicht möglich, die Detailgenauigkeit der Anwendung der Größe des Datensatzes anzupassen und so die Rechenzeit zu steuern.

Die Tabelle 3.1 fasst die Eigenschaften der in diesem Kapitel vorgestellten Methoden zum Vergleich der dreidimensionalen Struktur von Proteinen zusammen. Für jedes Verfahren ist aufgeführt, ob geometrische und physikochemische Deskriptoren verwendet

werden und ob die Methode skalierbar ist, d.h. die Detailgenauigkeit anpassbar ist.

Tabelle 3.1.: Zusammenfassung der Eigenschaften der Ansätze zur Klassifizierung von Proteinen

Referenz	**Name**	**geometrische Deskriptoren**	**physikochemische Deskriptoren**	**Skalierbarkeit**
Modellierung der Proteinstrukturen anhand der Aminosäuren				
Holm and Sander [1994, 1996a,b], Holm [1998]	DALI	ja	nein	nein
Lessel and Schomburg [1994]	—	ja	nein	nein
Grindley et al. [1993]	—	ja	nein	nein
Artymiuk et al. [1994], Spriggs et al. [2003]	ASSAM	ja	ja	nein
Russell [1998], Wangikar et al. [2003]	DRESPAT	ja	nein	nein
Schmitt et al. [2002], Weskamp et al. [2004]	CAVBASE	ja	ja	nein
[Jambon et al., 2003]	SUMO	ja	ja	nein
Nussinov and Wolfson [1991]	PNAS	ja	nein	nein
Leibowitz et al. [1999, 2001]	MUSTA	ja	nein	nein
Wallace et al. [1997]	TESS	ja	nein	nein
Fischer et al. [1995]	—	ja	nein	nein
Lehtonen et al. [1999]	GENFIT	ja	nein	nein
Fortsetzung nächste Seite				

Fortsetzung				
Referenz	**Name**	**geometrische Deskriptoren**	**physikochemische Deskriptoren**	**Skalierbarkeit**
Kupas et al. [2004b]	—	ja	ja	nein
Modellierung der lösungsmittelzugänglichen Oberfläche des Proteins				
Rosen et al. [1998]	—	ja	nein	nein
Pickering et al. [2001]	—	ja	nein	nein
Kinoshita and Nakamura [2001, 2003]	eF-Site	ja	ja	nein
Poirrette et al. [1997]	—	ja	ja	nein
Oshiro and Kuntz [1998]	—	ja	ja	nein
Mason et al. [2004]	QSCD	ja	ja	nein
Stahl et al. [2000]	—	ja	ja	nein
Shulman-Peleg et al. [2004]	Site Engine	ja	ja	nein
Kastenmüller et al. [1998], Ankerst et al. [1999a,b]	Shape Histograms	ja	nein	ja
Morris et al. [2005]	—	ja	nein	ja
eigener Ansatz	WaveGeoMap	**ja**	**ja**	**ja**

4. Datensätze

Der in dieser Arbeit entwickelte Algorithmus zum Vergleich von Substrukturen in Proteinbindetaschen soll in drei Schritten auf die Funktionsfähigkeit und die Qualität der Ergebnisse hin überprüft werden. Der erste Schritt ist dabei ein Vergleich von wenigen Bindetaschen mit bekannten Ähnlichkeiten. Bei guten Ergebnissen auf dieser kleinen Auswahl aus der Fülle der Bindetaschen in der Datenbank CAVBASE wird im zweiten Schritt ein größerer Datensatz von Enzymstrukturen verwendet, um den Algorithmus zu validieren. Zur Überprüfung der Funktionsfähigkeit der Methode auch für andere Enzymklassen wird als Testdatensatz ein weiterer Datensatz von Enzymbindetaschen zu anderen EC-Nummern als derer des Validierungsdatensatzes verwendet. Auch in diesem Datensatz sind alle Ähnlichkeiten und Unterschiede zwischen den Bindetaschen bekannt. Der Algorithmus sollte diese richtig detektieren.

Im Folgenden sollen die in dieser Arbeit verwendeten Datensätze vorgestellt werden. Bei allen Datensätzen sind für jede EC-Nummer Bindetaschen von verschiedenen in der Proteindatenbank (PDB) abgelegten Strukturen verwendet worden.

4.1. Datensatz aus 109 Bindetaschen zu 12 ausgewählten EC-Nummern ohne Ähnlichkeiten untereinander (DS V1)

Der für den Validierungsschritt verwendete Datensatz *DS V1* besteht aus ausgewählten Bindetaschen von 109 Enzymen, die in 2290 Patches zerlegt werden. Die Bindetaschen sind nach unterschiedlichen EC-Nummern (siehe Kapitel 2.3) klassifiziert. Er enthält zu den EC-Hauptklassen eins bis vier und sechs verschiedene Unterklassen, zur Hauptklasse eins drei, zur Hauptklasse fünf eine und zu allen anderen zwei Unterklassen. Der Datensatz dient zum *Proof-of-Concept* und zur Validierung der Methode. Alle in diesem Datensatz verwendeten Enzyme mit unterschiedlichen EC-Nummern weisen keine über den Reaktionsmechanismus oder Ähnlichkeiten der Substrate begründbaren Ge-

meinsamkeiten in ihren aktiven Zentren auf. Der Algorithmus sollte diese vorgegebenen Unterschiede auch richtig detektieren. Die Tabelle 4.1 enthält die PDB-Codes der zugehörigen Enzyme der Bindetaschen dieses Datensatzes.

4.2. Datensatz aus 708 Bindetaschen zu 13 ausgewählten EC-Nummern ohne Ähnlichkeiten (DS V2)

Dieser Datensatz ist eine Erweiterung des Validierungsdatensatzes *DS V1* aus 109 Bindetaschen. Er enthält alle in der CAVBASE enthaltenen Enzyme mit den in Tabelle 4.2 aufgeführten EC-Nummern. Zu vielen der Strukturen sind mehrere Taschen in der CAVBASE abgelegt. Es ist jeweils die größte Tasche verwendet worden, da diese in den meisten Fällen die tatsächliche Ligandenbindestelle ist. Der Datensatz enthält 708 Bindetaschen aus 13 verschiedenen EC-Klassen, die in 12435 Patches zerlegt werden.

4.3. Datensatz aus 1494 Bindetaschen zu den 13 EC-Nummern mit den meisten Einträgen in CAVBASE (DS T1)

Dieser Datensatz enthält Enzyme zu den 13 EC-Nummern mit den meisten Einträgen in der Datenbank CAVBASE. Die enthaltenen Enzyme sind mit ihren EC-Nummern in Tabelle 4.3 aufgeführt. Zu vielen Enzymen sind mehrere Taschen in der CAVBASE abgelegt. Es ist jeweils die größte Tasche verwendet worden, da diese in den meisten Fällen die tatsächliche Ligandenbindestelle ist. Der Datensatz enthält 1494 Bindetaschen, die in 29684 Patches zerlegt werden. Die Enzyme, deren erste drei Stellen der EC-Nummern unterschiedlich sind, weisen keine Gemeinsamkeiten in ihren Bindestellen auf. Die Polymerasen, bei denen sich nur die vierte Stelle der EC-Nummer unterscheidet, haben Gemeinsamkeiten in den Bindetaschen.

Tabelle 4.1.: Tabelle mit den EC-Nummern, den Namen und der Anzahl der Strukturen des Datensatzes *DS V1* (109 Einträge)

EC-Nummer	**Name des Enzyms**	**Anzahl**	**PDB-Codes**
EC 1.1.1.21	Aldehyde reductase	10	1ads, 1ah0, 1ah3, 1ah4, 1az1, 1az2, 1ef3, 2acr, 2acs, 2acu
EC 1.1.1.42	Isocitrate dehydrogenase	8	1ai2, 1ai3, 1bl5, 1cw1, 1gro, 1grp, 8icd, 9icd
EC 1.14.13.2	4-hydroxybenzoate-3-monooxygenase	7	1bf3, 1cj2, 1cj3, 1d7l, 1ius, 1pxa, 1pxb
EC 2.7.1.37	Protein kinase	16	1bkx, 1atp, 1cdk, 1ydr, 1hck, 1ckp, 1b38, 1gol, 1phk, 1csn, 1ir3, 2src, 1fin, 1erk, 3lck, 1p38
EC 2.7.4.9	dTMP kinase	11	1e9a, 1e9b, 1e9c, 1e9d, 1e9e, 1gtv, 1tmk, 2tmk, 3tmk, 4tmk, 5tmp
EC 3.4.21.62	Subtilisin	9	1au9, 1bfu, 1bh6, 1c3l, 1c9j, 1sua, 1sud, 1sue, 5sic
EC 3.4.23.20	Penicillopepsin	8	1apt, 1apu, 1apv, 1apw, 1bxo, 1bxq, 2wed, 3app
EC 4.2.1.1	Carbonate dehydratase	23	1cil, 1a42, 1ca2, 1g52, 1g54, 1i8z, 1i90, 1if4, 1if8, 1bcd, 1azm, 1bzm, 1czm, 1flj, 1znc, 2znc, 3znc, 1dmx, 1dmy, 1keq, 1urt, 1jcz, 1jd0
EC 4.4.1.11	Methionine gamma-lyase	4	1e5e, 1e5f, 1gc0, 1gc2
EC 5.4.2.1	Phosphoglycerate mutase	7	1bq3, 1bq4, 1e58, 1e59 1ejj, 1eqj, 1qhf
EC 6.3.2.3	Glutathione synthase	5	1glv, 1gsa, 1gsh, 2glt, 2hgs
EC 6.3.2.9	UDP-N-acetylmuramoylalanine–D-glutamate ligase	1	1e0d

Tabelle 4.2.: Tabelle mit den EC-Nummern und den Namen der Strukturen des Datensatzes *DS V2* (708 Einträge)

EC-Nummer	**Name des Enzyms**	**Anzahl**
EC 1.1.1.21	Aldehyde reductase	30
EC 1.1.1.42	Isocitrate dehydrogenase (NADP(+))	35
EC 1.14.13.2	4-hydroxybenzoate-3-monooxygenase	35
EC 2.7.1.37	Protein kinase	220
EC 2.7.4.9	dTMP kinase	27
EC 3.4.21.62	Subtilisin	76
EC 3.4.23.20	Penicillopepsin	14
EC 4.2.1.1	Carbonate dehydratase	209
EC 4.4.1.11	Methionine gamma-lyase	8
EC 5.4.2.1	Phosphoglycerate mutase	14
EC 5.3.3.1	Steroid delta-isomerase	27
EC 6.3.2.3	Glutathione synthase	7
EC 6.3.2.9	UDP-N-acetylmuramoylalanine–D-glutamate ligase	6

Tabelle 4.3.: Tabelle mit den EC-Nummern und den Namen der Strukturen des Datensatzes *DS T1* (1494 Einträge)

EC-Nummer	**Name des Enzyms**	**Anzahl**
EC 1.1.1.1	Alcohol dehydrogenase	71
EC 2.1.1.45	Thymidylate synthase	100
EC 2.5.1.18	Glutathione transferase	140
EC 2.6.1.1	Aspartate transaminase	88
EC 2.7.7.6	DNA-directed RNA polymerase	20
EC 2.7.7.48	RNA-directed RNA polymerase	53
EC 2.7.7.49	RNA-directed DNA polymerase	43
EC 3.2.1.1	Alpha-amylase	91
EC 3.4.21.4	Trypsin	308
EC 3.4.23.16	HIV-1 retropepsin	246
EC 3.4.24.27	Thermolysin	57
EC 4.2.1.1	Carbonate dehydratase	209
EC 5.3.1.5	Xylose isomerase	68

4.4. Datensatz aus 283 Bindetaschen von Endopeptidasen (DS T2)

Dieser Datensatz enthält eine Auswahl von 283 Taschen aus der Familie der Endopeptidasen. Diese werden in 2930 Oberflächenpatches zerlegt. An diesem Datensatz soll gezeigt werden, dass sowohl Gemeinsamkeiten als auch Unterschiede zwischen den einzelnen Unterfamilien gefunden werden können. Die Tabelle 4.4 zeigt die verschiedenen EC-Nummern und die Namen der zugehörigen Enzyme dieses Datensatzes.

Tabelle 4.4.: Tabelle mit den EC-Nummern und den Namen der Strukturen des Datensatzes *DS T2* (283 Einträge) aus Endopeptidasen verschiedener Familien

EC-Nummer	**Name des Enzyms**	**Anzahl**
Serinproteasen		
EC 3.4.21.4	Trypsin	30
EC 3.4.21.5	Thrombin	30
EC 3.4.21.22	Koagulationsfaktor IXa	6
EC 3.4.21.36	Pankreas Elastase	30
EC 3.4.21.62	Subtilisin	30
Cysteinproteasen		
EC 3.4.22.1	Cathepsin B	11
EC 3.4.22.36	Caspase 1	14
EC 3.4.22.38	Cathepsin K	22
EC 3.4.22.51	Cruzipain	13
Asparthylproteasen		
EC 3.4.23.16	HIV-1 Retropepsin	30
EC 3.4.23.20	Penicillopepsin	14
EC 3.4.23.22	Endothiapepsin	30
EC 3.4.23.25	Saccharopepsin	11
EC 3.4.23.39	Plasmepsin II	12

4.5. Erweiterung des Datensatzes aus Bindetaschen von Endopeptidasen (DS P)

Dieser Datensatz enthält eine Auswahl von 118 Taschen aus der Familie der Endopeptidasen. Diese werden in 1376 Oberflächenpatches zerlegt. Der Datensatz stellt eine Erweiterung des Datensatzes *DS T2* dar. An diesem Datensatz soll verifiziert werden, dass die entsprechenden Gemeinsamkeiten und Unterschiede zwischen den einzelnen Unterfamilien gefunden werden, wenn die Daten auf die mit dem Datensatz *DS T2* antrainierte ESOM projiziert werden. Die Tabelle 4.4 zeigt die verschiedenen EC-Nummern und die Namen der zugehörigen Enzyme dieses Datensatzes.

Tabelle 4.5.: Tabelle mit den EC-Nummern und den Namen der Strukturen des Datensatzes *DS P* (118 Einträge) aus Endopeptidasen verschiedener Familien

EC-Nummer	**Name des Enzyms**	**Anzahl**
Serinproteasen		
EC 3.4.21.1	Chymotrypsin	40
EC 3.4.21.6	Koagulationsfaktor Xa	37
EC 3.4.21.21	Koagulationsfaktor VIIa	13
Cysteinproteasen		
EC 2.7.7.48	SARS Proteinase	4
EC 3.4.22.52	Calpain-1	5
EC 3.4.22.53	Calpain-2	4
Asparthylproteasen		
EC 3.4.23.5	Cathepsin D	3
EC 3.4.23.24	SAP	3
EC 3.4.23.46	Beta-Sekretase	9

4.6. Zusammenfassung der vorgestellten Datensätze

In diesem Kapitel sind die verwendeten Datensätze vorgestellt worden. Es sind zwei Validierungsatensätze zur Ermittlung der Parametereinstellungen sowie drei Testdatensätze zum Überprüfen der Generalisierung der Methode ausgewählt worden. Der erste Validierungsdatensatz *DS V1* besteht aus 109 Bindetaschen zu 13 verschiedenen Enzymklassen. Diese weisen zwischen den Enzymklassen keine funktionell begründeten Ähnlichkeiten auf. Der zweite Validierungsdatensatz *DS V2* ist eine Erweiterung des ersten Datensatzes auf 708 Bindetaschen. Er enthält die größten Bindetaschen aller in der CAVBASE enthaltenen Enzyme zu den in *DS V1* verwendeten EC-Nummern. Bei beiden Datensätzen sollen die Substrukturen der Taschen nach den zugehörigen EC-Nummern voneinander getrennt werden.

Der erste Testdatensatz *DS T1* enthält Enzyme zu den 13 EC-Nummern mit den meisten Einträgen in der Datenbank CAVBASE. Er besteht aus 1494 Bindetaschen. Die Enzyme, deren erste drei Stellen der EC-Nummern unterschiedlich sind, weisen keine Gemeinsamkeiten in ihren Bindestellen auf. Die Polymerasen, bei denen sich nur die vierte Stelle der EC-Nummer unterscheidet, haben Gemeinsamkeiten in den Bindetaschen.

Der zweite Testdatensatz *DS T2* besteht aus 283 Endopeptidasen. Diese gehören den drei Subfamilien der Serinproteasen, Cysteinproteasen und Asparthylproteasen an. Innerhalb einer Subfamilie haben die Bindetaschen sehr ähnliche Teilbereiche. Diese Gemeinsamkeiten und Unterschiede sollen mit Hilfe der entsprechenden Substrukturen gefunden werden.

Der dritte Testdatensatz besteht aus weiteren 118 Enzymen der drei Subfamilien der Endopeptidasen des Datensatzes *DS T2*. Für die Patches der Enzyme soll für die mit dem Datensatz *DS T2* berechneten Cluster die Clusterzugehörigkeit bestimmt werden. Mit diesem Datensatz soll überprüft werden, ob Gemeinsamkeiten und Unterschiede zwischen den Taschen auch ohne Neuberechnung der Clusterung korrekt gefunden werden.

5. Klassifizierung von Oberflächenpatches in Protein-Bindetaschen

In einem Vorversuch (siehe Anhang C) wurden die Proteinbindetaschen in lokale Strukturen aus vier Pseudozentren [Schmitt et al., 2002] zerlegt. Diese lokalen Strukturen haben die geometrische Gestalt einer Pyramide mit dreieckiger Grundfläche. Es wurde deshalb die Anzahl von vier Pseudozentren gewählt, da die Dreieckspyramide die kleinste räumliche Struktur darstellt, die aus einer Menge von Pseudozentren gebildet werden kann. Diese minimalen Substrukturen wurden über die drei geometrischen Eigenschaften Höhe, Grundfläche und Schiefe der Pyramide sowie über die physikochemischen Eigenschaften der vier zugehörigen Pseudozentren charakterisiert. Die Deskriptoren sind geclustert worden, um gemeinsame Substrukturen der Bindetaschen zu finden [Kupas et al., 2004a,b, Kupas and Ultsch, 2005].

Der Minimalansatz, lokale Strukturen aus vier Pseudozentren zu verwenden, hat gezeigt, dass es über die Clusterung von Substrukturen in den Bindetaschen möglich ist, funktionelle Ähnlichkeiten von Enzymen zu detektieren. Jedoch ist eine solche minimale Substruktur zu klein, um auch sicher Unterschiede bestimmen zu können. Viele nicht verwandte Bindetaschen haben solche lokalen Regionen gemeinsam. Zudem ist sie nicht in der Lage, Ähnlichkeiten bei Bindetaschen, die in ihrer Konformation variabel sind, zu detektieren, da sich die Nachbarschaftsbeziehungen der Pseudozentren bei Variabilität der Taschen zu stark verändern.

Die in dieser Arbeit verfolgte Strategie ist, den zu untersuchenden Teilbereich der Bindetasche so genau wie möglich wiederzugeben und sowohl geometrische als auch physikochemische Eigenschaften so gut wie möglich zu approximieren. Grundlage für diese „Maximalstrategie“ ist die Menge der in CAVBASE abgelegten Oberflächenpunkte. Diese stellen eine gute Approximierung der räumlichen Gestalt der Bindetasche dar und sind mit den physikochemischen Eigenschaften der zugehörigen Pseudozentren ver-

sehen. Zudem ist die Klassifizierung durch die Größe der Substrukturen und die Unabhängigkeit von den Nachbarschaftsbeziehungen der Pseudozentren robuster bezüglich möglicher Flexibilität in den Bindetaschen. Nachteile solch „maximaler“ Substrukturen ist die Tatsache, dass ihre Charakterisierung wesentlich komplexer und aufwendiger ist und damit mehr Rechenzeit für eine Klassifizierung gebraucht wird.

5.1. Algorithmus zur Klassifizierung der Substrukturen

Der Algorithmus zur Klassifizierung solch „maximaler“ Substrukturen gliedert sich in mehrere Schritte:

- Zerlegung der Bindetaschen in Teilmengen von Oberflächenpunkten,
- Transformation in ein kanonisches Koordinatensystem und Entfaltung,
- Filtern von unvollständigen Patches,
- Ausrichtung der Patches,
- Berechnung der physikochemischen und geometrischen Deskriptoren,
- Aufbereitung der Daten,
- ESOM-Clusterung der Daten.

Im Folgenden werden diese Schritte näher erläutert. Die einzelnen Algorithmen im Pseudocode finden sich in Appendix B.

Generierung der Oberflächenpatches

Sei S eine diskrete Menge von Punkten einer Oberfläche im $\mathbb{R}^3$. Der Floyd-Abstand d_F zwischen zwei Oberflächenpunkten wird mit dem Algorithmus der kürzesten Wege (Floyd-Algorithmus Cormen et al. [1990]) berechnet. Dabei wird jeder Punkt der Oberfläche als Knoten eines Graphen aufgefasst. Jeder Knoten hat Kanten zu den Oberflächenpunkten, die einen euklidschen Abstand ¡ 1 haben. Die Gewichte der Kanten sind die euklidschen Abstände der Knoten. Der Abstand zwischen zwei Punkten der Oberfläche ist dann der kürzeste Weg in diesem Graphen. Bildlich gesprochen wird der Abstand „entlang der Oberfläche“ gemessen.

Definition 5.1.1 (Oberflächenpatch) *Sei S eine diskrete Menge von Punkten einer Oberfläche im $\mathbb{R}^3$, $M \in S$ ein Punkt dieser Menge, $r > 0$ der Radius und $d_F(x, M)$ der Floyd-Abstand von $x \in S$ zu M. Das Oberflächenpatch P ist dann definiert als*

$$P = \{x \in S |\ d_F(x, M) \leq r\}.$$

Die Mittelpunkte M der Patches sind die Oberflächenpunkte, die die kleinste euklidsche Distanz zu den Pseudozentren haben. Damit ist eine einheitliche Zerlegung der Bindetasche in Oberflächenpatches gewährleistet. Die Randpunkte KR der Patches seien alle diejenigen Oberflächenpunkte, für die gilt: $d_F(x, M) \geq r - \frac{r}{10}$.

Die Bindetaschen werden anhand der Pseudozentren systematisch in ihre Oberflächenpatches zerlegt. Ein jedes dieser Patches stellt eine planare kreisförmige Oberfläche dar, die durch ihre verschiedenen Höhen beschrieben werden kann.

Transformation in ein kanonisches Koordinatensystem

Um eine vergleichbare Darstellung der Patches zu erhalten, müssen diese in ein kanonisches Koordinatensystem transformiert werden. Dazu wird zunächst der Mittelpunkt des betrachteten Patches in den Ursprung geschoben. Anschließend wird das Patch so rotiert, dass es mit seiner kreisförmigen Grundfläche parallel zur xy-Ebene liegt. Die unterschiedliche Strukturierung der Patches wird dann durch die z-Koordinaten definiert, die die Höhen der Patchoberfläche angeben. Da nicht immer die Kreisebene, die die „Grundfläche“ der Patches darstellt, die beiden Achsen mit der größten Varianz enthält, ist die standardmäßig verwendete PCA zur Rotation der Patches nicht geeignet. Deswegen wird alternativ für jedes Patch der Schwerpunkt bestimmt und der Vektor zwischen Ursprung = Mittelpunkt und Schwerpunkt des Patches wird dann auf die positive z-Achse rotiert.

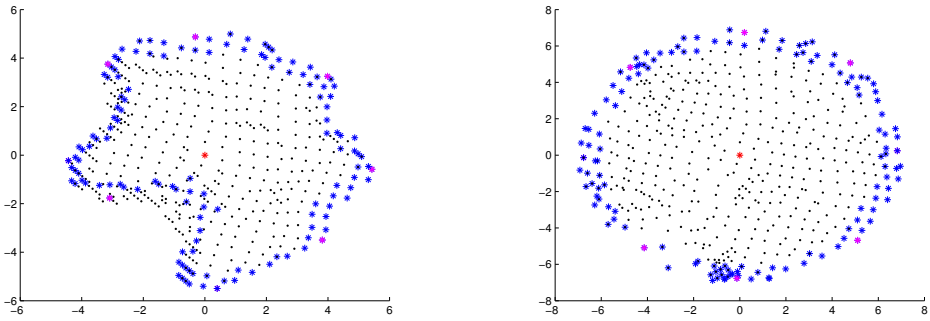
Da die Bindetaschen unterschiedliche Subtaschen besitzen und unterschiedlich stark vergraben [Hendlich et al., 1997] sind, kommt es vor, dass die Patches nun nicht kreisförmig sind, sondern dass z.B. der Rand des Patches an einer Seite „umgeklappt“ ist (siehe Abb. 5.1(a)). Deswegen werden alle Patches nun noch entfaltet.

Sei $p \in P$ ein beliebiger Punkt eines Oberflächenpatches, $m \in P$ der Mittelpunkt des entsprechenden Patches, $d_F(p, M)$ die Floyd-Distanz zwischen p und M, $p\vec{M}$ die Projektion des Vektors zwischen p und M in die xy-Ebene und $|p\vec{M}|$ die Länge des

Vektors. Dann ist

$$p_{neu} = \frac{p\vec{M} * d_F(p, M)}{|p\vec{M}|}.$$

Die Vektoren vom Mittelpunkt zu den Oberflächenpunkten werden auf die Floyd-Distanz entlang der Oberfläche zu diesem Punkt gestreckt und der Oberflächenpunkt dann dementsprechend verschoben (siehe Abb. 5.1(b)). Betrachtet werden dabei nur die x- und y-Koordinaten der Vektoren, die z-Koordinaten bleiben unverändert, da diese die Höhe des Patches an diesem Punkt darstellen.



(a) Patch mit eingeklapptem Rand (b) Patch entfaltet

Abbildung 5.1.: Entfaltung von Oberflächenpatches. Der Mittelpunkt des Patches ist rot, die Randpunkte sind blau gefärbt.

Filtern von unvollständigen Patches

Patches am Rand der Bindetasche bilden zumeist keine vollständige kreisförmige Fläche und sollen nicht betrachtet werden. Deswegen werden alle die Patches herausgefiltert, deren xy-Koordinaten der Randpunkte keinen vollständigen Kreis bilden. Dazu wird der Vektor [r,0] in acht 45°-Schritten um die z-Achse rotiert und der nächste Nachbar aus der Menge der Randpunkte des Patches bestimmt. Ist die Differenz zwischen dem Paar von Punkten größer als eins, so wird das Patch verworfen. Damit ist gesichert, dass nur solche Patches betrachtet werden, die einen nahezu vollständigen Rand haben.

Ausrichten der Patches

Als letzter Schritt werden die Patches nun noch so ausgerichtet, dass das Teilstück mit der niedrigsten mittleren Höhe im ersten Quadranten der xy-Ebene liegt. Dazu werden

die Patches in acht Sektoren unterteilt. Diese Unterteilung ist durch die Koordinatenachsen und durch zwei Ringe R_i mit den Radien $r_i = \frac{1}{i} \cdot max\{d(M,x)|\ x \in P\},\ i = 1,2$ gegeben. Um den Fehler bei der Approximierung der z-Werte durch den Mittelwert so gering wie möglich zu halten, wird die Sektoreinteilung so gewählt, dass jeder Sektor möglichst gut durch diesen Mittelwert beschrieben wird. Dazu wird das Patch in 10°-Schritten um die z-Achse rotiert und dann in die acht Sektoren aufgeteilt. Für jeden Sektor wird die Summe der quadrierten Abweichungen der z-Koordinaten vom Mittelwert berechnet. Der Drehwinkel für die optimale Sektoreinteilung wird so gewählt, dass die Summe der Fehlermaße für die einzelnen Sektoren den kleinsten Wert annimmt. Damit wird das Patch so um die z-Achse gedreht, dass der Fehler bei der Beschreibung der Höhenwerte jedes Sektors durch den Mittelwert minimal ist. Anschließend werden die so zerlegten Patches dann in einer Weise um die z-Achse rotiert, dass der Sektor mit dem kleinsten Mittelwert der z-Koordinaten im 1. Quadranten der xy-Ebene liegt.

Berechnen der Deskriptoren für die Patches

Für die Patches werden eine Reihe von Deskriptoren berechnet. Dazu gehören Werte, die die geometrische Gestalt der Patches beschreiben, sowie physikochemische Eigenschaften der Patches wiedergeben. Erstere werden in den nachfolgenden Kapiteln 5.2 und 5.3, letztere in Kapitel 5.4 näher erläutert.

Aufbereitung der Deskriptoren

Für das ESOM-Training ist eine Aufbereitung der einzelnen Deskriptoren eines Datensatzes nötig (siehe Kapitel 2.1). Bei allen Paaren von Merkmalen mit einem Pearson-Korrelationskoeffizienten größer oder gleich 0.8 wird eines der Merkmale eliminiert. Von den übrigen Merkmalen werden die Verteilungen mit Hilfe von Histogrammen, *PDE-* und *QQ-Plots* betrachtet. Bei schiefen Verteilungen werden die Werte des Merkmals so transformiert, dass die Verteilung der transformierten Werte symmetrisch ist. Um Ausreisser zu entdecken, werden noch *Boxplots* der Merkmale betrachtet. Offensichtliche Ausreisser, die mit einem visuell signifikanten Abstand von den anderen Daten liegen, werden entfernt.

Da die Deskriptoren der physikochemischen Eigenschaften Werte im Intervall $[0,1]$ annehmen, werden auch die geometrischen Merkmale in das Intervall $[0,1]$ transformiert, damit die Merkmale bei der Distanzberechnung gleich gewichtet werden.

ESOM-Clusterung der Substrukturen

Nach der Aufbereitung wird mit den Deskriptoren der Patches aller Bindetaschen als Eingabewerte eine toroide ESOM antrainiert, um Strukturen in den Daten entdecken zu können. Die Anzahl der Deskriptoren ist je nach verwendeter Methode zur Beschreibung der Patches unterschiedlich (siehe Kapitel 5.2, 5.3 und 5.4). Die Anzahl der Neuronen der Karte richtet sich nach der Größe des Datensatzes, sie beträgt mindestens das Vierfache der Anzahl der Datenpunkte. Diese Kartengröße ist empirisch ermittelt worden. Bei kleineren Karten sind die Strukturen der Datensätze nicht sichtbar. Das Ergebnis des ESOM-Trainings wird mit der U-Matrix visualisiert. Für eine Vorklassifizierung wird der U*C-Cluster-Algorithmus (siehe Kap. 2.1) verwendet. Das Ergebnis dieser U*C-Clusterung wird anschließend auf der U-Matrix überprüft. Zu große Cluster werden weiter in kleinere Cluster unterteilt. Solche Datenpunkte, die aufgrund der Dichtestruktur der Daten einem Cluster zugeordnet werden, aber auf der U-Matrix schon außerhalb des Clusters liegen, werden aus dem Cluster entfernt. Als Ergebnis erhält man eine Distanz- und Dichte-basierte Clusterung der Substrukturen. Jedes dieser Cluster enthält Substrukturen mit sehr ähnlichen Deskriptionsvektoren, während sich solche aus verschiedenen Clustern stark voneinander unterscheiden. Die Clusterung mit dem U*C-Cluster-Algorithmus und der anschließenden Nachbearbeitung der Clusterung wird im Folgenden als ESOM-Clusterung bezeichnet.

5.2. Beschreibung der Oberfläche durch Mittelwert und Varianz von Patchsektoren (SectorGeoMap)

In dieser Arbeit werden zwei verschiedene Strategien zur Charakterisierung der Oberfläche eines solchen Patches verfolgt. Die Strategie der *SectorGeoMap* ist eine relativ grobe Beschreibung der Patchstruktur. Ein Vorteil ist die kurze Berechnungszeit der Deskriptoren. Ein Nachteil dieser Methode ist, dass durch die recht grobe Beschreibung nicht alle Ähnlichkeiten und Unähnlichkeiten richtig detektiert werden (siehe Kapitel 6.1). Die Strategie beruht auf dem Konzept der Deskriptorberechnung für die *Shape Histograms* von Ankerst et al. [1999a]. Die *Shape Histograms* werden zum Vergleich kompletter Rezeptorstrukturen definiert. Nach vorhergehender Koordinatentransformation auf eine kanonische Darstellung wird die dreidimensionale Struktur in die xy-Ebene projiziert. Um diese Projektion wird ein Kreis beschrieben und dieser spinnennetzartig in Sektoren aufgeteilt. Dazu wird der Kreis vom Mittelpunkt aus in verschiedene Ringe

und diese dann in verschiedene Stücke aufgeteilt. Für jeden Sektor wird die Anzahl der Atome in diesem Sektor ermittelt. Die so entstehenden Vektoren werden dann mit einem gewichteten Distanzmaß, welches benachbarte Sektoren berücksichtigt, verglichen.

Das Prinzip der Aufteilung in Sektoren zur Beschreibung der Oberflächenpatches sowie die gewichtete Distanzberechnung zur Berücksichtigung der Nachbarschaftsbeziehungen der einzelnen Sektoren ist auf die Oberflächenpatches angewandt worden. Jedes Patch wird in acht verschiedene Sektoren aufgeteilt (siehe Kapitel 5.1). Jeder dieser Sektoren ist durch die z-Koordinaten der Oberflächenpunkte innerhalb des Sektors charakterisiert. Als geometrische Deskriptoren werden die Mittelwerte und Varianzen der z-Koordinaten der einzelnen Sektoren berechnet.

Die acht Sektoren werden nach folgendem Schema durchnummeriert (siehe Abbildung 5.2): Die Nummer 1 ist der Sektor im inneren Ring im 1. Quadranten der xy-Ebene. Dann wird im Uhrzeigersinn auf dem inneren Ring weiter gezählt. Danach wird der äussere Ring durchnummeriert, auch hier wird im 1. Quadranten der xy-Ebene begonnen. Für jedes Patch wird ein Deskriptionsvektor berechnet, der für die acht Sektoren in

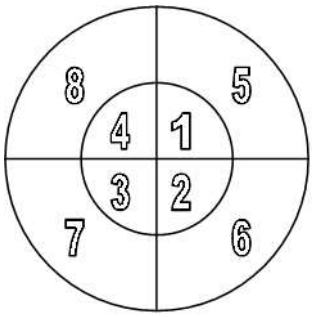


Abbildung 5.2.: Schema der Nummerierung der Sektoren der Oberflächenpatches

der oben beschriebenen Reihenfolge die Mittelwerte und Varianzen der z-Koordinaten enthält.

Mit den transformierten Deskriptoren der Oberflächenpatches wird eine emergente SOM antrainiert. Als Distanzmaß wird jedoch nicht die euklidische Distanz verwendet, sondern ein gewichtetes Distanzmaß, bei dem nebeneinanderliegende Sektoren miteinander verrechnet werden [Ankerst et al., 1999a]. Das gewichtete Distanzmaß erlaubt Toleranz gegenüber Flexibilität in den Bindetaschen. Der Abstand dR zwischen zwei Ringen des Patches ist die Differenz der Radien, der Abstand dS zwischen zwei Stücken des Patches ist der Winkel zwischen deren Mittelpunkten. Der kombinierte Abstand

zweier Sektoren ist dann definiert als

$$d(i,j) = \sqrt{((dR(i,j)^2 + dS(i,j)^2))}$$

Die Einträge der Matrizen WM_k, $k = 1 \ldots 7$ sind bestimmt durch $w_{ij} = e^{(-d(i,j))}$. Die Gewichtsmatrix WM für die gewichtete Abstandsbestimmung ist die Diagonalmatrix dieser WM_k mit zusätzlichen WM_k an der Stelle des Deskriptionsvektors für übereinstimmende physikochemische Eigenschaften, also Donor und Donor/Akzeptor und Akzeptor und Donor/Akzeptor. Das gewichtete Distanzmaß ist dann definiert als

$$d_{WM}(x,y) = (x-y) \cdot WM \cdot (x-y)^T = \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij}(x_i - y_i)(x_j - y_j)$$

mit x und y als zwei Deskriptionsvektoren und $N = 48$ die Länge dieser Vektoren.

5.3. Beschreibung der Oberfläche durch Wavelets (WaveGeoMap)

Die zweite Strategie benutzt zur Charakterisierung der geometrischen Gestalt der Patches eine Wavelet-Transformation (siehe Kapitel 2.2). Dieses Verfahren ist zwar rechenintensiver, erlaubt aber eine wesentlich genauere Beschreibung der Oberfläche, als es durch die *SectorGeomaps* möglich ist. Ein Vorteil ist, dass es möglich ist, über die Anzahl der Koeffizienten nur die Grobstruktur oder aber auch die Feinheiten in der Oberfläche zu vergleichen.

Als Startgitter V_0 werden neun Oberflächenpunkte verwendet, der Mittelpunkt und acht Punkte auf dem Rand des Patches mit jeweils gleichem Abstand zueinander. Da die Patches kanonisch ausgerichtet sind, enthalten nur die z-Koordinaten dieser Punkte Information. Folglich dienen dann auch nur diese als Approximationskoeffizienten. Dann werden die Iterationsschritte der Wavelet-Transformation durchlaufen bis zum Gitter V_i je nach gewünschter Detailgenauigkeit (siehe Kapitel 2.2). Dabei muss immer $\|\{v \mid v \in V_i\}\| \leq \|\{x \mid x \in P\}\|$ gelten. Für eine erste Klassifizierung einer großen Menge von Patches wird dabei ein Iterationschritt zum Gitter V_1 ausreichen, da dies schon eine grobe Beschreibung der Oberflächeneigenschaften der Patches liefert.

An den Zwischenpunkten, an denen die Oberfläche glatt ist, gibt es zusätzlich zu den Approximationskoeffizienten keine neue Information bezüglich der Gestalt der Ober-

fläche. Dort sind auch die Absolutbeträge der Wavelet-Koeffizienten nahe bei Null. Alle die Wavelet-Koeffizienten, deren Absolutbetrag groß ist, enthalten dagegen neue Information bezüglich der Geometrie der Patchoberfläche. Nur diese Koeffizienten sind für die Clusterung interessant. Um nicht die Ortsinformation der Koeffizienten speichern zu müssen, werden die n Koeffizienten als Deskriptoren verwendet, deren Absolutbetrag im Mittel bei allen Patches am grössten ist [Mörchen, 2003]. Dadurch werden für alle Patches die gleichen Koeffizienten verwendet und die Notwendigkeit einer Speicherung des Orts entfällt.

Der für das ESOM-Training verwendete Deskriptionsvektor für jedes Patch setzt sich somit zusammen aus den Deskriptoren für die geometrische Gestalt der Patchoberfläche, das sind die neun Approximationskoeffizienten und die n Waveletkoeffizienten der i Resolutionsstufen, und den physikochemischen Deskriptoren, also der prozentualen Verteilung der fünf Eigenschaften in vier Sektoren eines Patches. Der Deskriptorvektor hat daher die Dimension $d = 9 + n * i + 5 * 4 = 29 + n * i$.

5.4. Charakterisierung der physikochemischen Eigenschaften der Patches

Für die Beschreibung der physikochemischen Eigenschaften werden die Patches in Viertelkreissektoren, wie im Kapitel 5.1 beschrieben, aufgeteilt und für jeden dieser vier Sektoren wird die prozentuale Verteilung der fünf Eigenschaften (Donor, Akzeptor, kombinierte Donor-Akzeptor-, aliphatische und aromatische Eigenschaft) berechnet. Diese Beschreibung ist ein guter Kompromiss zwischen einer sehr detaillierten Beschreibung der Chemie und wenigen Eigenschaften, um noch kurze Rechenzeiten beim ESOM-Training zu gewährleisten. Eine Verteilung der Eigenschaften auf dem gesamten Patch liefert zwar weniger Dimensionen des Deskriptionsvektors, bietet aber auch nur eine sehr grobe Beschreibung der Bindungseigenschaften dieser Stelle der Bindetasche. Eine Aufteilung in mehr Sektoren oder aber die Berücksichtigung der physikochemischen Eigenschaft eines jeden Oberflächenpunktes liefert zwar eine sehr detaillierte Beschreibung des Patches, führt aber auch zu vielen Dimensionen des Deskriptionsvektors und somit zu langen Rechenzeiten. Aufgrund der geringen Größe der Oberflächenpatches erscheint es praktikabel und ausreichend, eine Unterteilung in vier Sektoren und für jeden Sektor eine Berechnung der Anteile der fünf verschiedenen molekularen Erkennungs- und Bindungseigenschaften vorzunehmen.

5.5. Parameterstudien zur Patchgröße

Die korrekte Funktionalität des Algorithmus zum Auffinden von ähnlichen Oberflächenpatches hängt maßgeblich von deren Größe ab. Wird der Radius zu klein gewählt, so umfasst das Patch einen zu kleinen Bereich der Bindetasche, und es werden viele zufällige Ähnlichkeiten zwischen Bindetaschen gefunden, die funktionell keine Ähnlichkeiten aufweisen. Ist der Radius zu groß, so umfasst das Patch mehr als den Bereich einer funktionellen Subtasche, und dadurch wird zu stark zwischen den Bindetaschen diskriminiert. Es können dann nur noch fast identische Bindetaschen als ähnlich erkannt werden. Solche Gemeinsamkeiten, die sich auf einzelne Subtaschen beschränken, können nicht mehr detektiert werden.

Um die optimale Größe der Patches zu bestimmen, sind umfangreiche Parameterstudien durchgeführt worden. Anhand einer Zerlegung der Bindetaschen in konvexe Teilbereiche mit dem Algorithmus CavDecomp durch Nils Weskamp ist eine mittlere Größe einer solchen Subtasche erstellt worden. Diese beläuft sich auf neun bis zehn Pseudozentren. Da die am häufigsten auftretende Distanz zweier benachbarter Pseudozentren 1.3 Å beträgt, ergibt sich eine ungefähre Größe einer solchen Subtasche von 13 Å. Dieser Wert diente zur groben Orientierung bei der Auswahl des Radius für die Parameterstudien. Es sind für den Radius der Patches Werte aus $[4, 12] \cap \mathbb{N}$ für die Ermittlung des optimalen Radius gesetzt worden. Die Bestimmung des optimalen Radius erfolgte aber nur anhand der Clusterergebnisse. Für die Studien ist der Validierungdatensatz *DS V1* verwendet worden, da für diesen Datensatz bekannt ist, dass zwischen Enzymen zu unterschiedlichen EC-Nummern keine direkt aufgrund des Reaktionsmechanismus oder der gebundenen Substrate anzunehmenden Ähnlichkeiten bestehen. Die Ergebnisse der Clusterung können daher leicht bestimmt werden. Eine korrekte Diskriminierung zwischen ähnlichen und unterschiedlichen Patches ist äquivalent zu Clustern, die ausschließlich Patches aus Bindetaschen von Enzymen zu derselben EC-Nummer enthalten. Zudem sollte mindestens ein Cluster zu jeder der dreizehn verschiedenen EC-Klassen gefunden werden. Ein Maß für die Qualität der Clusterung ist deshalb die Reinheit der Cluster sowie die Vollständigkeit der Clusterung, d.h. für jede EC-Nummer existiert mindestens ein reiner Cluster.

Die Abbildungen 5.3 und 5.4 zeigen die Ergebnisse der Parameterstudie. Auf der x-Achse ist der Radius in Å aufgetragen. Auf der y-Achse ist das Verhältnis zwischen reinen Clustern und nicht reinen Clustern aufgetragen. Für Radii $r \leq 10$ ist die Clusterung vollständig, für Radii $r \geq 11$ ist sie unvollständig. Bei der Beschreibung der

Patchoberfläche durch Aufteilung in Sektoren liegt das Maximum der Kurve und somit der optimale Radius bei 9 Å, bei der Beschreibung durch Wavelets bei 10 Å.

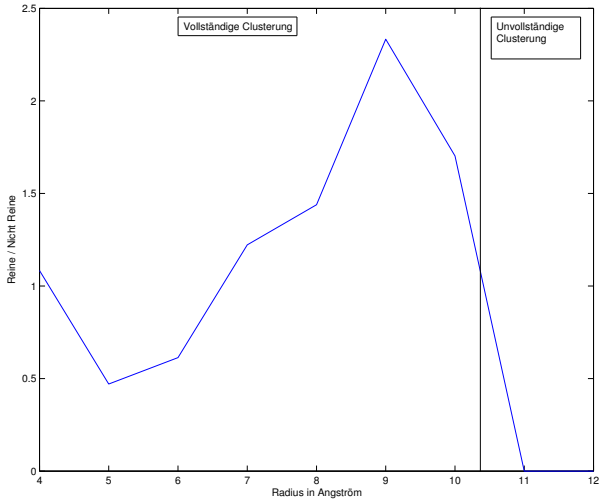


Abbildung 5.3.: Parameterstudien zur Methode SectorGeoMap

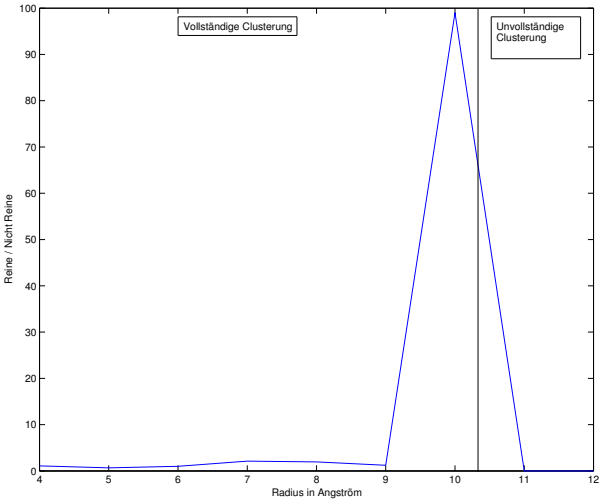


Abbildung 5.4.: Parameterstudien zur Methode WaveGeoMap

Die Tabelle 5.1 enthält zusammenfassend die Ergebnisse der Parameterstudien für beide Möglichkeiten zur Beschreibung der geometrischen Eigenschaften der Patches. Sie gibt die prozentuale Verteilung der Cluster für die verschiedenen Radii an. Reine Cluster enthalten nur Patches zu derselben EC-Nummer, quasi-reine Cluster enthalten mind. 90% zu einer und max. 10% zu einer zweiten EC-Nummer und Mischcluster sind alle übrigen Cluster.

5.6. Zusammenfassung der Klassifizierung von Oberflächenpatches

In diesem Kapitel werden die Zerlegung der Bindetaschen in Oberflächenpatches und die für die Clusterung verwendeten Deskriptoren definiert. Die Oberflächenpunkte der Bindetaschen werden systematisch in kreisrunde Patches zerlegt. Die Mittelpunkte dieser Patches werden anhand der zugehörigen Pseudozentren der Bindetasche bestimmt, so dass eine einheitliche Zerlegung der Taschen gewährleistet ist. Jedes dieser Patches wird in ein kanonisches Koordinatensystem transformiert, so dass die Gestalt der Oberfläche durch die z-Koordinaten der Patchpunkte bestimmt ist, und ausgerichtet, damit die Patches untereinander vergleichbar sind. Anschließend werden die Patches entfaltet, so dass jedem Punkt (x,y) genau ein z-Wert zugeordnet ist.

Zwei verschiedene Methoden werden vorgestellt, die geometrische Gestalt der Patches zu beschreiben. Eine Möglichkeit ist die Strategie der *SectorGeoMap*. Für jeden Sektor werden Mittelwert und Varianz der z-Koordinaten berechnet. Diese Charakterisierung der Patches erfordert wenig Rechenzeit, ist aber auch nicht sehr detailliert. Die zweite Möglichkeit ist die Strategie der *WaveGeoMap*, eine Beschreibung der Patches durch die Koeffizienten sogenannter Oberflächenwavelets. Diese Methode bietet den Vorteil, dass die Anzahl der Koeffizienten je nach Anwendung und Grösse des Datensatzes angepasst werden kann und damit eine gröbere oder eine detailliertere Beschreibung der Oberfläche möglich ist, was mit einer kürzeren oder längeren Rechenzeit einhergeht.

Für die physikochemischen Deskriptoren werden die Patches in vier Sektoren aufgeteilt und die Verteilungen der fünf essentiellen Eigenschaften Wasserstoffbrücken-Donor, -Akzeptor, gemischt -Donor-Akzeptor, aliphatische und aromatische Eigenschaften berechnet. Diese Beschreibung ist ein guter Konmpromiss zwischen sehr vielen Werten und damit großer Genauigkeit und sehr wenigen Deskriptoren; wie die Verteilung der Eigenschaften auf dem gesamten Patch, und damit einer groben Beschreibung der Wechselwirkungseigenschaften dieses Teils der Bindetasche.

Tabelle 5.1.: Tabelle mit den Ergebnissen der ESOM-Clusterung für den Validierungdatensatz *DS V1* bei verschiedenen Patchgrößen. Angegeben ist die prozentuale Verteilung der Cluster. Reine Cluster sind solche, die nur Patches von Enzymen zu der gleichen EC-Nummer enthalten, quasi-reine Cluster enthalten mind. 90% zu einer und max. 10% zu einer zweiten EC-Nummer, und Misch-Cluster sind alle übrigen Cluster. Für Radii $r \leq 10$ ist die Clusterung vollständig, für Radii $r \geq 11$ ist sie unvollständig, d.h. es existiert nicht zu jeder EC-Nummer mindestens ein reiner Cluster.

SectorGeoMap			
Radius in Å	**Reine Cluster**	**Quasi-reine Cluster**	**Misch-Cluster**
4	52%	28%	20%
5	32%	32%	36%
6	38%	29%	33%
7	55%	28%	17%
8	59%	22%	19%
9	70%	15%	15%
10	63%	15%	22%
11	97%	3%	0%
12	64%	19%	17%
WaveGeoMap			
Radius in Å	**Reine Cluster**	**Quasi-reine Cluster**	**Misch-Cluster**
4	52%	30%	18%
5	40%	29%	31%
6	50%	33%	17%
7	68%	24%	8%
8	66%	17%	17%
9	55%	32%	13%
10	99%	1%	0%
11	100%	0%	0%
12	55%	20%	25%

Für beide Möglichkeiten der geometrischen Beschreibung sind Parameterstudien durchgeführt worden. Bei dem Ansatz der *SectorGeoMap* liefert ein Radius von 9 Å die besten Clusterergebnisse für den Validierungdatensatz *DS V1*. Ist der Radius kleiner, werden die Enzyme nicht mehr so gut nach ihren EC-Nummern getrennt. Ist der Radius größer, existiert nicht zu jeder EC-Nummer mindestens ein Cluster. Für den Ansatz der *WaveGeoMap* liegt der optimale Radius bei 10 Å.

6. Ergebnisse der Klassifizierung von Oberflächenpatches in Protein-Bindetaschen

Im Folgenden werden die Ergebnisse der in Kapitel 5 beschriebenen Methoden zur Klassifizierung von Oberflächenpatches vorgestellt. Die Oberflächen der Bindetaschen werden in Patches zerlegt und für jedes Patch werden geometrische und physikochemische Deskriptoren berechnet. Mit den Deskriptoren wird eine ESOM mit ca. der vierfachen Menge der Datensätze an Neuronen trainiert. Dieser Wert ist ein Erfahrungswert, kleinere Karten haben wesentlich schlechtere Ergebnisse bei der Clusterung geliefert. Die Daten werden mit der ESOM-Clusterung in Klassen aufgeteilt.

Für die verwendeten Datensätze existiert durch die EC-Nummern der Enzyme eine Vorklassifizierung der Bindetaschen gemäß ihrer Funktion (siehe Kapitel 2.3). Die Ergebnisse der Clusterung werden mit der EC-Klassifizierung der zugehörigen Enzymstrukturen verglichen, um die Güte der Clusterung zu bewerten. Jede Bindetasche hat mehrere Subtaschen mit unterschiedlichen Eigenschaften, in denen bestimmte Bereiche des Liganden binden. Dazu gehören unter anderem solche Bereiche, die für die spezifische Erkennung des Liganden wichtig sind sowie die katalytischen Bereiche, in denen die Umsetzung des Substrats dieses Enzyms stattfindet. Die verschiedenen Subtaschen unterscheiden sich sowohl in ihrer Geometrie als auch in ihren physikochemischen Eigenschaften.

Bei einer Zerlegung der Bindetaschen in Patches werden deshalb mehrere kleine Cluster zu einer EC-Nummer entsprechend der verschiedenen Substrukturen in der Bindetasche erwartet. Zudem weisen einige der verwendeten Enzyme Variabilität in ihren Taschen auf. Diese Taschen liegen in unterschiedlichen Konformationen vor, die zu unterschiedlichen Geometrien der Patches führen. Für diese Bereiche entstehen weitere zusätzliche kleine Cluster, die den verschiedenen Konformationen dieses Teilbereichs der Bindetasche entsprechen. Alle in diesem Kapitel abgebildeten U-Matrizen sind als In-

seln wie in Kapitel 2.1 beschrieben dargestellt. Die einzelnen Cluster entsprechen den blauen und grünen Bereichen der U-Matrix, da diese geringe Distanzen zwischen den Datenpunkten bedeuten. Sie sind die Täler auf der U-Matrix. Die braunen und weißen Bereiche sind die Clustergrenzen, da sie große Distanzen zwischen den Datenpunkten bedeuten. Sie sind die Bergzüge der Distanzlandschaft. Die Einfärbung der U-Matrix gemäß der Distanzen zwischen den Datenpunkten ist in Abbildung 2.2 dargestellt.

Je nach Datensatz lieferte die ESOM-Clusterung zwischen 50 und 300 kleinen Clustern von Patches. Dabei sollten innerhalb eines Clusters nur ähnliche Teilbereiche der Bindetasche liegen. Für die Validierungsdatensätze *DS V1*, *DS V2* sowie den Testdatensatz *DS T1* entspricht das der Reinheit der Cluster, da die Bindetaschen zu verschiedenen EC-Nummern untereinander keine durch den Reaktionsmechanismus oder die gebundenen Substrate begründbaren Ähnlichkeiten aufweisen. Eine perfekte Clusterung der Patches würde nur Cluster mit Patches zu derselben EC-Nummer hervorbringen. Für den Datensatz *DS T2*, der drei verschiedene Unterfamilien aus der Enzymfamilie der Endopeptidasen enthält, werden für jede der drei Unterfamilien sowohl gemischte als auch reine Cluster erwartet. Die gemischten Cluster sollten die Teilbereiche der Bindetasche enthalten, die alle Enzyme einer Unterfamilie gemeinsam haben. Für die reinen Cluster werden solche Bereiche in den Bindetaschen erwartet, die sich von denen der anderen Enzyme der Subfamilie unterscheiden und die Spezifität des Enzyms ausmachen.

Das Kapitel ist chronologisch zu der Entwicklung der Methode aufgebaut. Zunächst ist die *SectorGeoMap* zum Vergleich der Patches verwendet worden. Die Ergebnisse für einen ersten Datensatz aus drei Bindetaschen sind vielversprechend. Die U-Matrix zum Validierungsdatensatz *DS V1* zeigt jedoch noch viele Regionen mit Bestmatches zu unterschiedlichen EC-Nummern, die biologisch nicht begründbar sind. Um solche großen Bereiche mit falsch klassifizierten Patches zu vermeiden, ist die Methode der *WaveGeoMap* entwickelt worden. Diese Methode liefert sehr gute Ergebnisse für den Validierungsdatensatz *DS V1*. Sie ist dann auch auf die anderen Datensätze angewendet worden und liefert auch hier die erwarteten Ergebnisse.

6.1. Ergebnisse der SectorGeoMap zum Vergleich der Patches

Erster Test der Methode mit drei Bindetaschen

Um die grundsätzliche Funktionalität des Algorithmus zu zeigen, ist für einen ersten Test ein Datensatz von drei Bindetaschen herangezogen worden. Dieser enthält die Bindetaschen zweier Strukturen einer Carboanhydrase II mit den PDB-Codes 1cil und 1cin und die einer HIV-I-Protease mit dem PDB-Code 1hiv. Davon besitzen 1cil und 1cin fast identische Bindetaschen, während 1hiv eine völlig unterschiedliche Bindetasche aufweist. Mit dem Datensatz soll gezeigt werden, dass gemeinsame Bereiche der Bindetaschen der Carboanhydrasen gefunden werden und die Patches korrekt von denen der HIV-I-Protease getrennt werden. Erwartet werden viele Täler auf der U-Matrix, die nur einen *Bestmatch* aus einer Bindetasche enthalten. Die zugehörigen Patches entsprechen den Teilbereichen der Bindetaschen, die unterschiedlich sind. Einige wenige Täler sollten entstehen, die mit jeweils einem Bestmatch der beiden Carboanhydrasen populiert sind. Diese entsprechen den gemeinsamen Subtaschen der Enzyme. *Bestmatches* zu Patches der HIV-I-Protease sollten alle in einem eigenen Tal liegen, da sie zu den Patches der anderen Bindetaschen keine funktionalen Ähnlichkeiten haben. Durch die Überlappung der Patches kann es jedoch vorkommen, dass Patches einer Bindetasche gemeinsam ein Tal auf der U-Matrix populieren.

Die drei Bindetaschen sind in 80 verschiedene Oberflächenpatches zerlegt worden. Die Patches sind in vier Sektoren aufgeteilt worden und für jeden Sektor sind Mittelwert und Varianz der z-Koordinaten sowie die prozentuale Verteilung der physikochemischen Eigenschaften berechnet worden. Mit den aufbereiteten 28 Deskriptoren dieser 80 Patches ist eine ESOM mit 50x82 Neuronen antrainiert worden. Die Abbildung 6.1 zeigt die dazugehörige U-Matrix. Die *Bestmatches* der Oberflächenpatches sind entsprechend der Zugehörigkeit zu einer der drei Bindetaschen eingefärbt. Alle zu 1cil gehörigen haben hellblaue Farbe, die zu 1cin gehörigen sind dunkelblau und die zu 1hiv gehörigen sind magenta gefärbt. Die U-Matrix zeigt vier Cluster mit jeweils einem hell- und einem dunkelblauen *Bestmatch*, d.h. *Bestmatches* von 1cil und 1cin liegen in einem Cluster. Es existieren keine Cluster, die sowohl Bestmatches von 1cil oder 1cin und 1hiv enthalten. Allerdings gibt es Cluster, die *Bestmatches* von Oberflächenpatches einer Bindetasche enthalten. Dies sind benachbarte Patches und die Ähnlichkeit ist auf die Überlappung der verschiedenen Patches zurückzuführen.

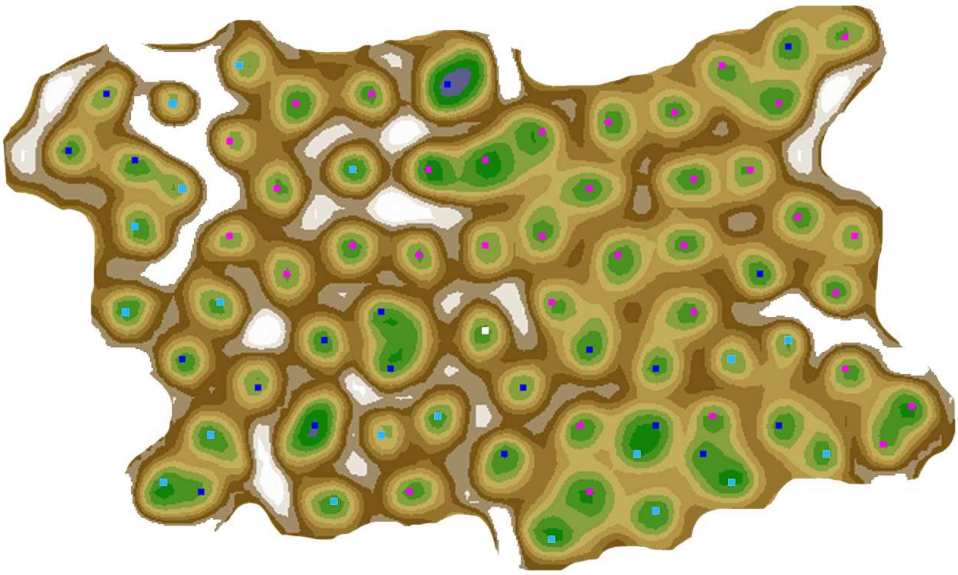


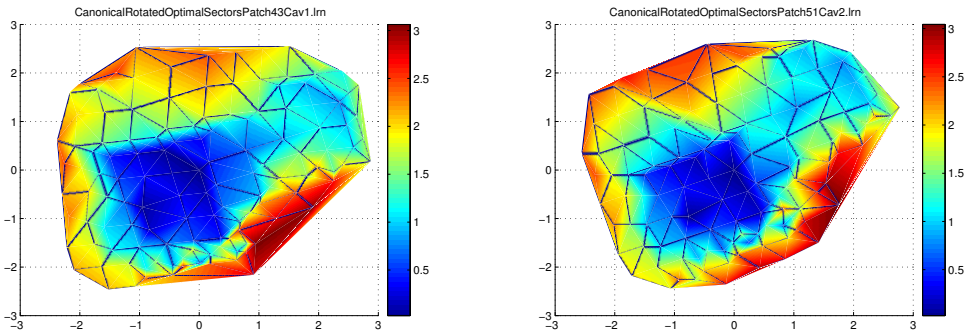
Abbildung 6.1.: Die Abbildung zeigt die U-Matrix zur SectorGeomap der Oberflächenpatches der drei Bindetaschen von 1cil, 1cin und 1hiv. Die Bestmatches der Taschen 1cil und 1cin sind ähnlich (hell- und dunkelblau) und die der Bindetasche von 1hiv sind unterschiedlich (magenta) gefärbt.

Die Abbildung 6.2 zeigt als Beispiel die Patches aus einer Klasse mit je einem *Bestmatch* aus 1cil und 1cin. Die beiden Patches haben eine fast identische geometrische Gestalt, was sich an der Färbung der Oberfläche festmachen lässt. Dabei bedeutet blau einen sehr kleiner Höhen-Wert und rot einen sehr großer Höhenwert. Die Verteilung der physikochemischen Eigenschaften dieser beiden Patches ist ebenso wie die geometrische Gestalt fast identisch.

Dieses Ergebnis unterstreicht, dass die Beschreibung von Oberflächenpatches durch Mittelwert und Varianz der Höhe in Sektoren zu korrekten Ergebnissen führt. Da man anhand dieser drei Bindetaschen noch keinerlei Aussage über die Güte der Clusterung mit diesen Deskriptoren treffen kann, wurde im Weiteren der Validierungsdatensatz *DS V1* aus 109 Enzym-Bindetaschen heran gezogen, um die Methode zu testen und zu validieren.

Ergebnisse für den Validierungsdatensatz DS V1

Die 109 Enzym-Bindetaschen sind in Oberflächenpatches mit einem Radius von 9 Å zerlegt worden. Dieser Radius hat sich bei Parameterstudien als optimaler Radius herausgestellt (siehe Kapitel 5.5). Alle Patches sind wie in Kapitel 5.1 beschrieben ausgerichtet



(a) Patch aus der Tasche von 1cil (b) Patch aus der Tasche von 1cin

Abbildung 6.2.: Zwei Patches der Taschen 1cil und 1cin aus derselben U-Matrix-Klasse in Aufsicht. Die Farben codieren die entsprechenden z-Werte. Sowohl die geometrische Gestalt als auch die zugewiesenen physikochemischen Eigenschaften der beiden Patches sind fast identisch.

worden und solche mit unvollständigem Rand sind herausgefiltert worden. Die Menge der Patches mit unvollständigem Rand hängt vom Radius ab, bei kleinem Radius sind es nur die äußersten Patches, bei größerem Radius auch Patches zu Pseudozentren weiter im Inneren der Tasche. Bei diesem Radius sind ca. 20 % der Patches vorher herausgefiltert worden. Es sind die physikochemischen Deskriptoren wie in Kapitel 5.4 beschrieben bestimmt worden. Für die geometrischen Deskriptoren sind alle 2290 Patches in die vier Sektoren aufgeteilt worden und für jeden Sektor sind Mittelwert und Varianz der Höhe berechnet worden (siehe Kapitel 5.2).

Zunächst ist eine ESOM mit 100x164 Neuronen alleine mit den physikochemischen Deskriptoren, also den prozentualen Anteilen der fünf physikochemischen Eigenschaften pro Sektor, trainiert worden. Eine Aufbereitung ist nicht notwendig, da die Eigenschaften alle Prozentwerte sind und damit im Intervall $[0, 1]$ liegen und die Verteilungen nicht schief sind.

Die U*C-Clusterung und anschließende Nachbearbeitung (ESOM-Clusterung) (siehe Kapitel 5.1) ergibt 84 verschiedene Cluster. Darunter sind einige Cluster mit Patches einer einzelnen EC-Nummer. Aber es existieren große Bereiche auf der U-Matrix, die Patches von Bindetaschen zu unterschiedlichen EC-Nummern enthalten. Ein direkter Vergleich der Clusterung mit der bestehenden EC-Klassifizierung ist nicht möglich, da jede EC-Klasse durch die Zerlegung der Bindetaschen in Patches in n verschiedene Cluster zerfällt. Betrachtet man die Reinheit der Cluster als Maß für die Güte der Clusterung, so enthalten 60% der Cluster nur Patches derselben EC-Nummer. 25% der Cluster

sind quasi-reine Cluster, sie enthalten mind. 90% Patches einer EC-Nummer und max. 10% Patches mit einer anderen EC-Nummer, und 15% sind gemischte Cluster. Die gemischten Cluster sind jedoch sehr groß und enthalten einen Großteil der Patches des Datensatzes. Zudem findet sich kein Cluster, welches nur Patches aus den Bindetaschen der Phosphoglycerat-Mutasen enthält.

Alleine durch die Verteilung der physikochemischen Eigenschaften in den vier Sektoren der Patches werden also die verschiedenen EC-Nummern nicht ausreichend getrennt. Daher ist es unerlässlich, auch die geometrische Gestalt der Patches in die Clusterung mit einzubeziehen. Die geometrischen Deskriptoren müssen vor dem ESOM-Training aufbereitet werden. Ausreißer wurden entfernt und anschließend die Daten in das Intervall $[0, 1]$ transformiert, um sie dem Wertebereich der physikochemischen Deskriptoren anzupassen. Mit den aufbereiteten Deskriptoren ist eine ESOM mit 100x164 Neuronen in 30 Lernepochen trainiert worden. Die ESOM-Clusterung ergibt 101 verschiedene Cluster. Die Zahl der nach der EC-Klassifikation eindeutig zugewiesenen und populierten Cluster ist größer als bei der ausschließlichen Verwendung der chemischen Eigenschaften der Patchoberfläche. Jedoch treten auch hier einige große koherente Bereiche mit Patches aus Taschen von Enzymen aus unterschiedlichen EC-Klassen auf. 70% der Cluster enthalten nur Patches derselben EC-Nummer, 15% der Cluster sind quasi-reine Cluster und 15% sind gemischte Cluster.

Eine Zerlegung der Bindetaschen in Oberflächenpatches liefert eine deutliche bessere Clusterung der Patches nach funktionellen Ähnlichkeiten der Bindetaschen als der in Anhang C beschriebene Minimalansatz, bei dem die Bindetaschen in lokale Regionen bestehend aus vier Pseudozentren zerlegt werden. Die Beschreibung der geometrischen Eigenschaften durch Mittelwert und Varianz der Höhenwerte in Sektoren eines Patches ist jedoch nicht hinreichend diskriminierend, um die Patches gemäß ihrer vorgegebenen EC-Klassifizierung voneinander zu trennen. Deswegen ist eine detailliertere Beschreibung der geometrischen Gestalt der Oberflächenpatches notwendig.

6.2. Ergebnisse der Methode der WaveGeoMap

Um eine genauere Beschreibung der Gestalt der Patches zu realisieren, sind zur Berechnung der geometrischen Deskriptoren Oberflächenwavelets verwendet worden (siehe Kapitel 5.3). Bei allen Datensätzen sind die Bindetaschen zunächst in Patches mit dem empirisch ermittelten optimalen Radius von 10 Å zerlegt worden (siehe Kapitel 5.5). Alle Patches sind wie in Kapitel 5.1 beschrieben ausgerichtet worden und alle Patches

mit unvollständigem Rand (ca. 20 %) sind herausgefiltert worden.

Aufbereitung der Deskriptoren

Für alle verbleibenden Patches sind die neun Approximationskoeffizienten, also die z-Koordinaten der acht Randpunkte und des Mittelpunkts (V_0), und die ersten 16 Waveletkoeffizienten, die zu einer ersten Unterteilung des Gitters V_0 in V_1 gehören, sowie die physikochemischen Deskriptoren (siehe Kapitel 5.4) verwendet worden. Die physikochemischen Deskriptoren sind symmetrisch zum Mittelwert verteilt, enthalten keine Ausreißer und liegen als Prozentwerte alle im Intervall [0,1]. Eine Aufbereitung ist also nicht notwendig. Bei den geometrischen Deskriptoren sind jedoch Aufbereitungsschritte nötig. Die neun Approximationskoeffizienten sind rechtsschief verteilt. Sie sind mit einem Exponent von 3 transformiert worden, um eine symmetrische Verteilung zu erhalten. Ausreißer sind in den Approximationskoeffizienten nicht vorhanden. Die Verteilungen der 16 Waveletkoeffizienten sind symmetrisch, eine nichtlineare Transformation ist also nicht nötig. Es sind aber Ausreißer vorhanden, so dass für die ersten acht Koeffizienten nur diejenigen im Bereich [-5,5] und für die restlichen Koeffizienten diejenigen im Bereich [-3,3] betrachtet wurden. Anschließend sind alle geometrischen Deskriptoren auf das Intervall [0,1] normiert worden, damit die Varianzen der Merkmale mit denen der physikochemischen Deskriptoren übereinstimmen.

Ergebnisse für den Validierungdatensatz DS V1

Auch hier ist zur Validierung zunächst der Datensatz von 2290 Patches zu 109 Enzym-Bindetaschen verwendet worden (siehe Kapitel 4.1). Mit den aufbereiteten 45 Deskriptoren ist eine ESOM mit 100x164 Neuronen trainiert worden. Die Abbildung 6.3 zeigt die zugehörige U-Matrix als Insel, die *Bestmatches* sind entsprechend der EC-Nummern der zugehörigen Enzyme gefärbt worden.

In Abbildung 6.4 ist eine politische Karte dieser U-Matrix zu sehen. Zusammenhängende Bereiche auf der U-Matrix mit niedrigen U-Höhen sind entsprechend der EC-Nummern der Bestmatches eingefärbt worden. Die Farbcodierung ist dieselbe wie in Abbildung 6.3. Der Hintergrund der U-Matrix ist ausgeblendet worden.

Die U-Matrix zeigt eine gute Trennung der verschiedenen Patches nach EC-Nummern. Es existieren deutlich voneinander abgegrenzte Regionen mit niedrigen U-Höhen, die meist nur Patches von Taschen der gleichen EC-Nummer enthalten. Die Bereiche der U-Matrix, in denen *Bestmatches* mit unterschiedlichen EC-Nummern liegen, sind Berg-

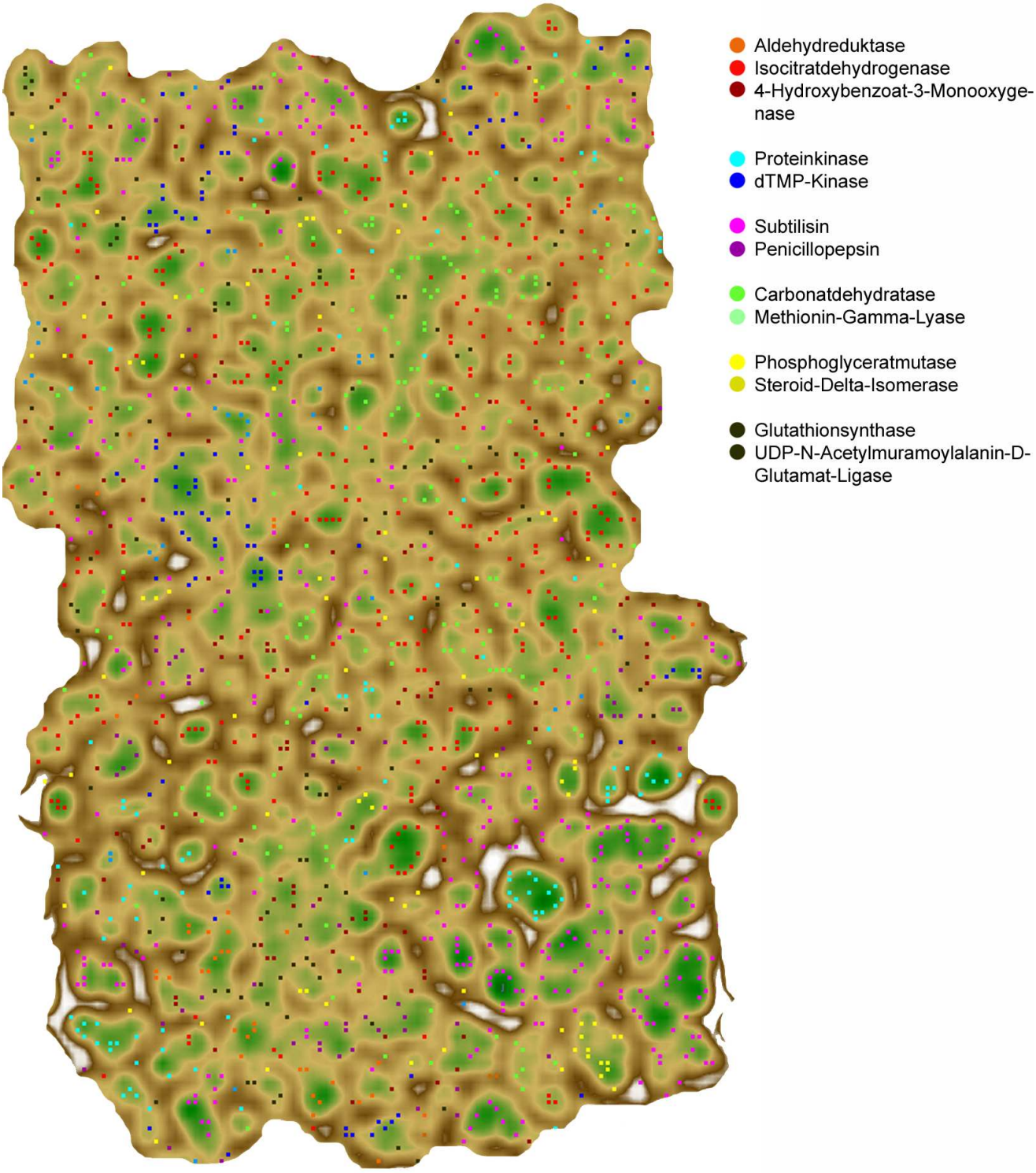


Abbildung 6.3.: U-Matrix zu den durch Waveletkoeffizienten charakterisierten 2290 Oberflächenpatches des Validierungsdatensatzes *DS V1*. Die Bestmatches sind entsprechend der verschiedenen EC-Nummern eingefärbt.

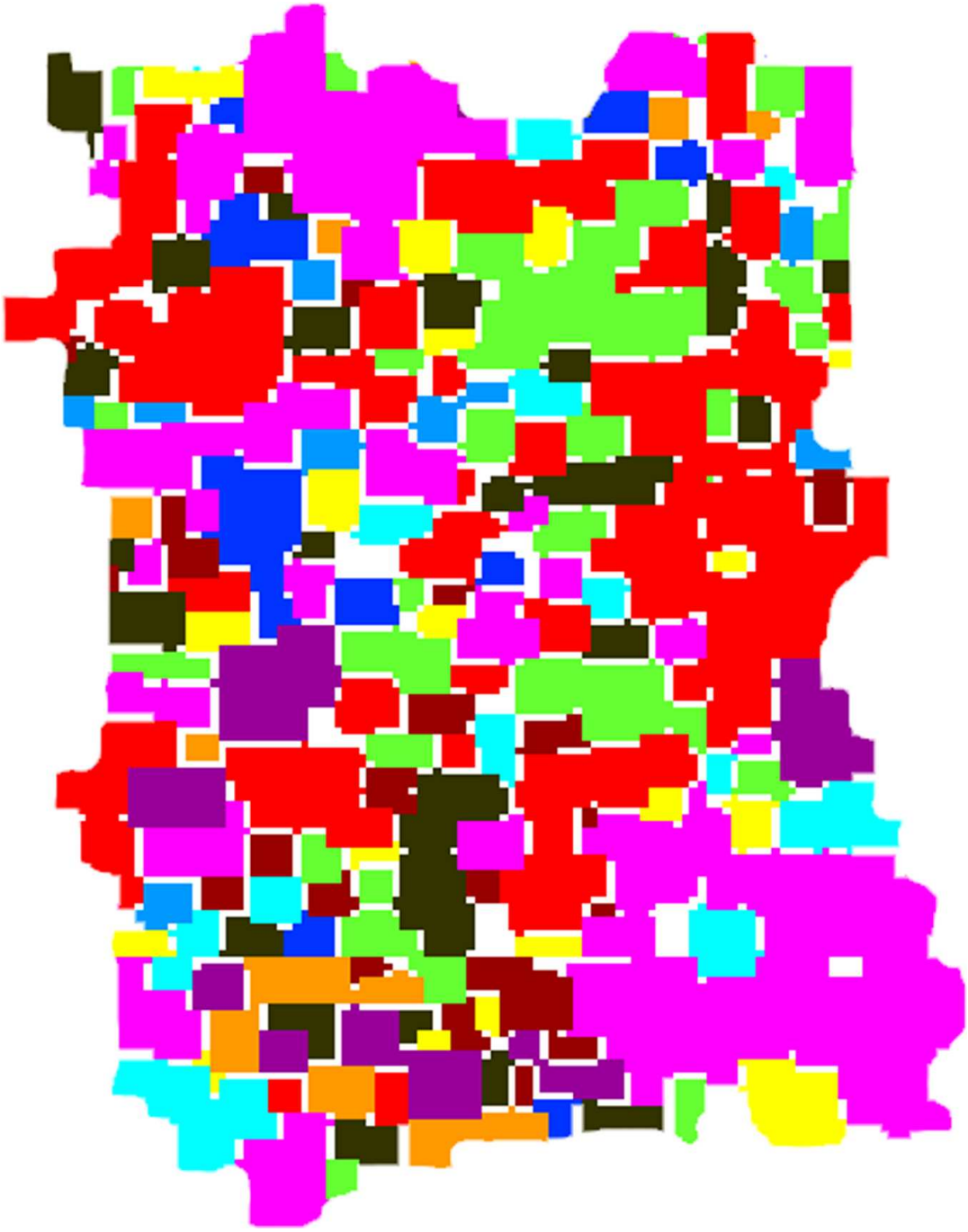
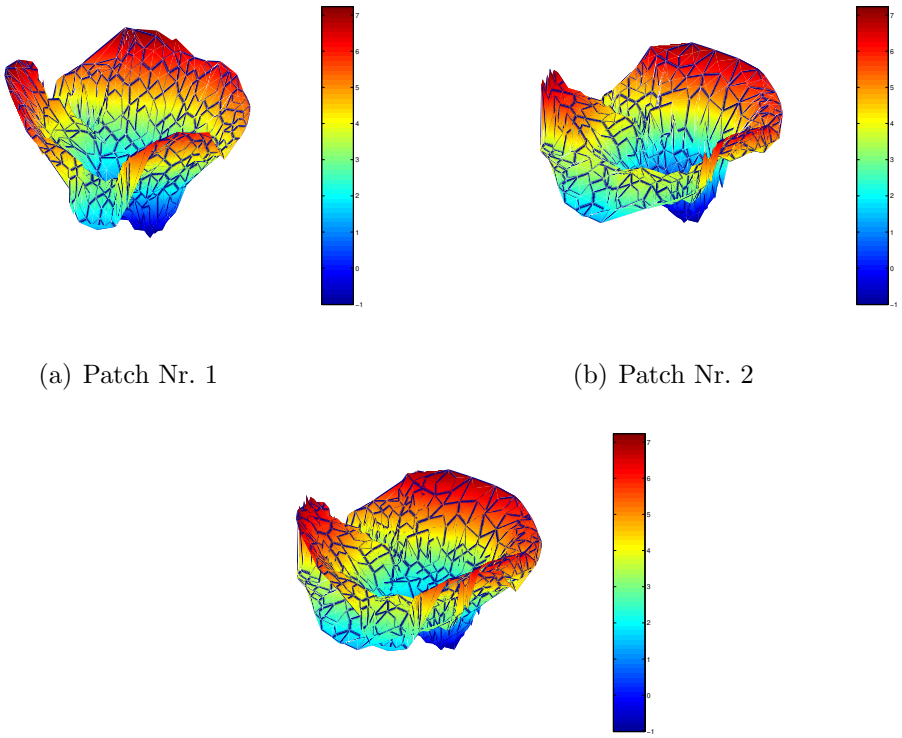


Abbildung 6.4.: Politische Karte zu den durch Waveletkoeffizienten charakterisierten 2290 Oberflächenpatches des Validierungsdatensatzes *DS V1*. Die Regionen sind entsprechend der verschiedenen EC-Nummern der zugehörigen Bestmatches eingefärbt.

züge, d.h. Regionen mit großen U-Höhen. Diese werden in die Clusterung nicht mit einbezogen, da sie große Abstände und damit Unähnlichkeit zwischen den *Bestmatches* bedeuten. Die U-Matrix zeigt, dass die verwendeten Deskriptoren und damit die Metrik und die verwendeten Parameter die Oberflächenpatches nach ihren EC-Nummern voneinander trennen. Sie sind also geeignet, um Ähnlichkeiten und Differenzen von Teilbereichen der Protein-Bindetaschen korrekt zu detektieren. Denn Bindetaschen der Enzyme zu derselben EC-Nummer sind sehr ähnlich, und die zugehörigen Patches liegen auch dicht zusammen im Datenraum. Patches von unterschiedlichen Bindetaschen liegen weit auseinander im Datenraum. Die ESOM-Clusterung mit der Vorclusterung durch den U*C-Clusteralgorithmus (siehe Kapitel 5.1) ergibt 52 verschiedene Cluster. Betrachtet man die Reinheit der Clusterung in Bezug auf die EC-Klassifizierung, so enthalten 51 dieser Cluster nur Patches derselben EC-Nummer, und nur eines der Cluster enthält zwei Patches von Enzymen mit unterschiedlicher EC-Nummer, eines der menschlichen Aldose-Reduktase und eines der Para-Hydroxybenzoat-Hydroxylase. Diese Patches sind sich aber sehr ähnlich. Die einzelnen Patches und die Überlagerung der beiden ist in Abbildung 6.5 zu sehen.



(a) Patch Nr. 1

(b) Patch Nr. 2

(c) Überlagerung der beiden Patches

Abbildung 6.5.: Überlagerung der Patches des Cluster Nr. 32 des Validierungdatensatzes *DS V1*.

Ergebnisse für den erweiterten Validierungdatensatz DS V2

Die mit dem Validierungdatensatz *DS V1* hinsichtlich ihrer Diskriminierung otpimierten Deskriptoren zur Beschreibung der Patches sind für den größeren, erweiterten Validierungsdatensatz *DS V2* (siehe Kapitel 4.2) berechnet worden und aufbereitet worden. Mit den Deskriptionsvektoren der 12435 Patches ist eine ESOM mit 300x492 Neuronen antrainiert worden. Die zugehörige U-Matrix in der Inseldarstellung ist in Abbildung 6.6 zu sehen. Auch hier sind die *Bestmatches* wieder nach den EC-Nummern der zugehörigen Enzymstrukturen eingefärbt worden.

In Abbildung 6.7 ist eine politische Karte dieser U-Matrix zu sehen. Zusammenhängende Bereiche auf der U-Matrix mit niedrigen U-Höhen sind entsprechend der EC-Nummern der Bestmatches eingefärbt worden. Die Farbcodierung ist dieselbe wie in Abbildung 6.6. Der Hintergrund der U-Matrix ist ausgeblendet worden.

Die Clusterung dieser ESOM (siehe Kapitel 5.1) ergibt 310 verschiedene Cluster. Betrachtet man die Reinheit der Clusterung in Bezug auf die EC-Klassifizierung, so enthalten 304 Cluster, das sind 98% aller Cluster, nur Patches derselben EC-Nummer, 6 Cluster, das sind 2% aller Cluster, sind quasi-reine Cluster, d.h. mehrfach populierte Cluster, enthalten aber mind. 90% Patches von Strukturen zu einer EC-Nummer und max. 10% Patches, die einem Protein einer anderen EC-Nummer zuzuweisen sind, und keine Cluster sind gemischte Cluster. Die Cluster Nr. 16, 206, 212, 262, 279 und 286 sind diese quasi-reinen Cluster, in welchen weniger als 10% der Patches eine andere EC-Nummer haben. Die Tabelle 6.1 zeigt den Inhalt dieser sechs Cluster. Die Ausreißer in den Clustern mit mehr als einer EC-Nummer haben jedoch eine sehr ähnliche Gestalt. Die Abbildung 6.8 zeigt die Patches und ihre Überlagerung eines dieser Cluster. Die Überlagerung der Patches einer Klasse ist sehr gut. Es existieren nur wenig Variationen in der geometrischen Gestalt der Patches.

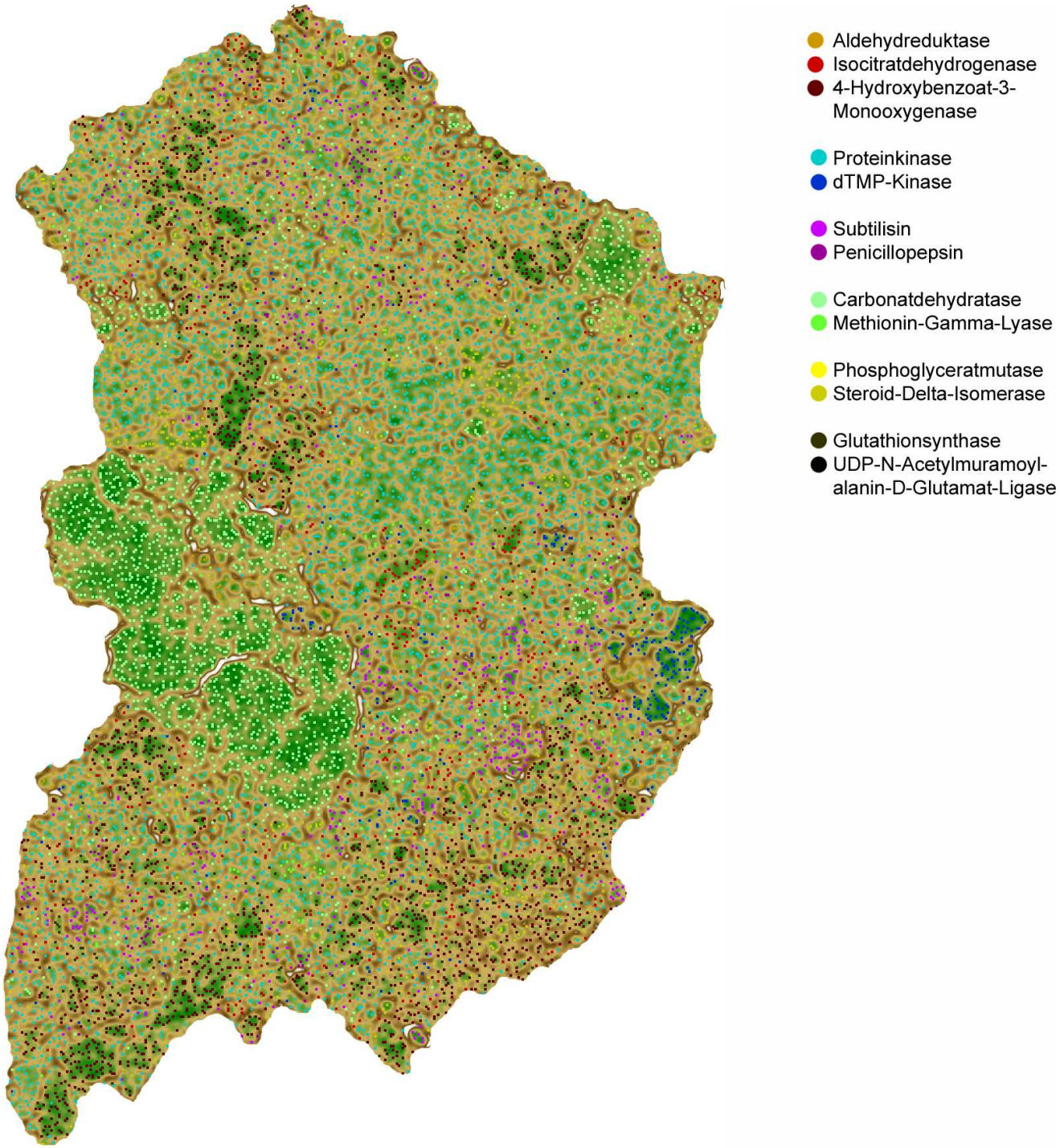


Abbildung 6.6.: U-Matrix zu den durch Waveletkoeffizienten charakterisierten 12435 Oberflächenpatches des erweiterten Validierungsdatensatzes *DS V2*. Die *Bestmatches* sind entsprechend der verschiedenen EC-Nummern eingefärbt. Die ESOM-Clusterung ergibt 310 Cluster.

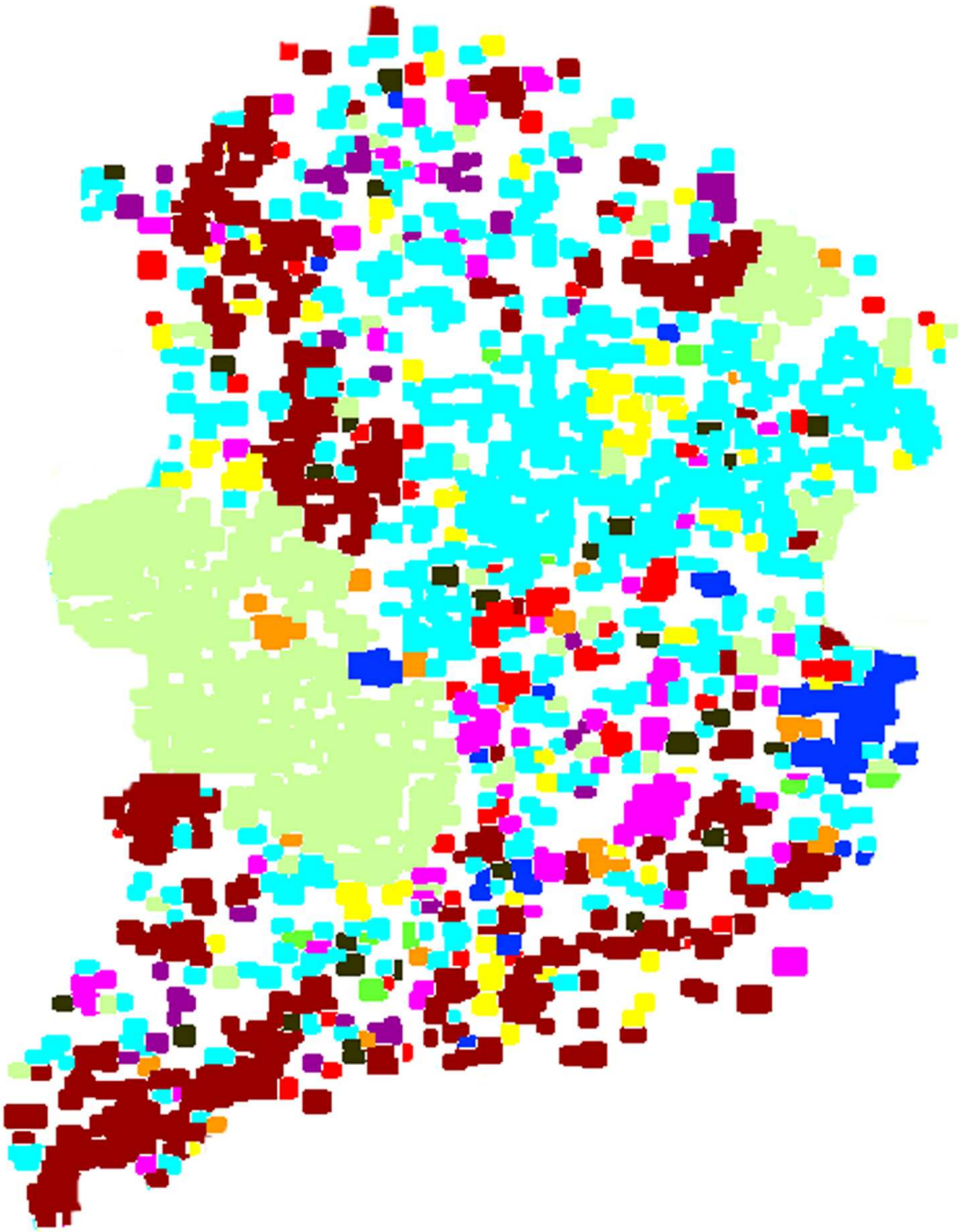


Abbildung 6.7.: Politische Karte zu den durch Waveletkoeffizienten charakterisierten 12435 Oberflächenpatches des Validierungsdatensatzes *DS V2*. Die Regionen sind entsprechend der verschiedenen EC-Nummern der zugehörigen Bestmatches eingefärbt.

Tabelle 6.1.: Clusternummern, PDB-Codes, EC-Nummern und Namen der Enzyme für die quasi-reinen Cluster des erweiterten Validierungdatensatzes *DS V2*

PDB-Codes	**EC-Nummern**	**Namen der Enzyme**
	Cluster Nr. 16	
1BF3	1.14.13.2	p-Hydroxybenzoate hydroxylase
1BGJ	"	"
1BKW	"	"
1CC6	"	"
1CJ2	"	"
1CJ4	"	"
1DOC	"	"
1IUS	"	"
1IUU	"	"
1IUW	"	"
1PBC	"	"
1PBD	"	"
1PBE	"	"
1PBF	"	"
1PHH	"	"
1PXB	"	"
1PXC	"	"
1J1B	2.7.1.37	Glycogen synthase kinase-3 beta
	Cluster Nr. 206	
1BL7	2.7.1.37	Glycogen synthase kinase-3 beta
1W83	2.7.1.37	"

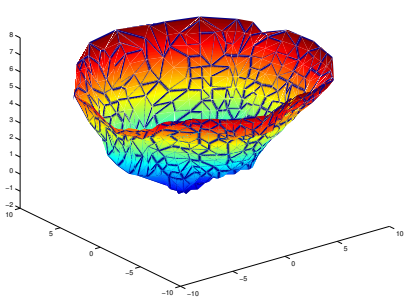
Fortsetzung nächste Seite

Fortsetzung		
PDB-Codes	**EC-Nummern**	**Namen der Enzyme**
1W83	2.7.1.37	"
1DDZ	4.2.1.1	Carbonate dehydratase
Cluster Nr. 212		
1O98	5.4.2.1	Phosphoglycerate mutase
1O99	5.4.2.1	"
1YQJ	2.7.1.37	Glycogen synthase kinase-3 beta
Cluster Nr. 262		
1I09	2.7.1.37	Glycogen synthase kinase-3 beta
1PBC	1.14.13.2	p-Hydroxybenzoate hydroxylase
Cluster Nr. 279		
1Y4A	3.4.21.62	Subtilisin
1ZXE	2.7.1.37	Glycogen synthase kinase-3 beta
Cluster Nr. 286		
1K0J	1.14.13.2	p-Hydroxybenzoate hydroxylase
1O99	5.4.2.1	Phosphoglycerate mutase

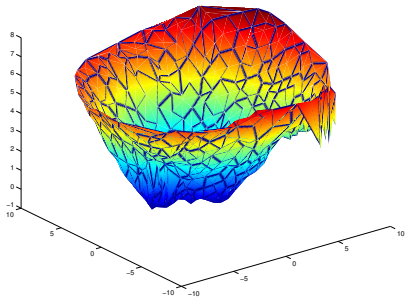
Ergebnisse für den Testdatensatz DS T1

Nachdem für die beiden Validierungsdatensätze vielversprechende Ergebnisse erzielt werden konnten, ist der Algorithmus auf den Testdatensatz *DS T1* (siehe Kapitel 4.3) angewendet worden. Dieser Datensatz besteht aus 1494 Enzym-Bindetaschen, welche in 29684 verschiedene Oberflächenpatches mit einem Radius von 10 Å zerlegt worden sind. Für die knapp 30000 Patches sind die physikochemischen und geometrischen Deskriptoren berechnet und aufbereitet worden. Auch für diesen Datensatz ist eine Vorklassifizierung durch EC-Nummern gegeben, jedoch enthält er zum größten Teil andere Enzymfamilien als die beiden Validierungsdatensätze.

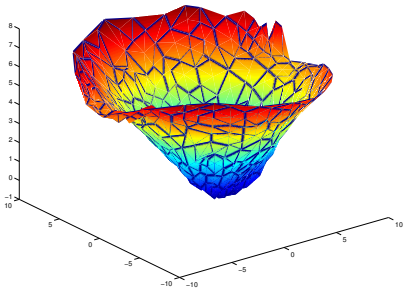
Mit den aufbereiteten Daten ist eine ESOM mit 300x492 Neuronen in 100 Epochen antrainiert worden. Die Abbildung 6.9 zeigt die zugehörige U-Matrix in der Inseldarstel-



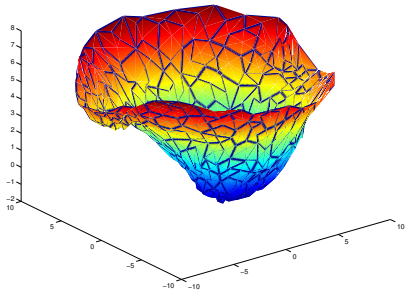
(a) Patch Nr. 1



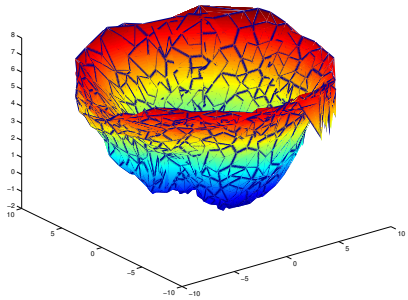
(b) Patch Nr. 2



(c) Patch Nr. 3



(d) Patch Nr. 4



(e) Überlagerung der Patches

Abbildung 6.8.: Patches des Cluster Nr. 206 des erweiterten Validierungdatensatzes *DS V2* und ihre Überlagerung.

lung. Alle *Bestmatches* sind wieder entsprechend der vorgegebenen EC-Klassifikation eingefärbt worden. Auch auf dieser Karte liegen *Bestmatches*, die zu Patches derselben Enzymfamilie gehören, dicht zusammen in Regionen mit niedrigen U-Höhen, während diejenigen unterschiedlicher Enzymfamilien einen weiten Abstand im Datenraum voneinander haben, was durch grosse U-Höhen sichtbar wird.

Da die Karte sehr groß und unübersichtlich ist, sind in den Abbildungen 6.10 bis 6.15 Teilbereiche dieser U-Matrix vergrößert dargestellt.

In Abbildung 6.16 ist eine politische Karte dieser U-Matrix zu sehen. Zusammenhängende Bereiche auf der U-Matrix mit niedrigen U-Höhen sind entsprechend der EC-Nummern der Bestmatches eingefärbt worden. Die Farbcodierung ist dieselbe wie in Abbildung 6.9. Der Hintergrund der U-Matrix ist ausgeblendet worden.

Die Clusterung dieser ESOM (siehe Kapitel 5.1) ergibt 332 verschiedene Cluster. Betrachtet man die Reinheit der Clusterung in Bezug auf die EC-Klassifizierung, so enthalten 305, also 92% der Cluster nur Patches von Proteinen derselben EC-Nummer. Insgesamt 24, also 7% der Cluster sind quasi-reine Cluster und enthalten mind. 90% Patches zu einer EC-Nummer und max. 10% Patches zu einer anderen EC-Nummer, und 1% sind gemischte Cluster. Cluster, die Patches von Proteinen zu mehr als einer EC-Nummer enthalten, sind häufig große Cluster, die einzelne Ausreißer enthalten. Cluster Nr. 97 enthält z.B. 180 Patches, davon sind 178 aus der Bindetasche des HIV-1-Retropepsin und 2 entstammen einer Xylose Isomerase. Cluster Nr. 88 enthält 68 Patches der Glutathion-Transferase und einen aus der Bindetasche des Trypsins. Die kleinen Cluster enthalten teilweise Patches, die von Enzymen stammen, die mehreren EC-Nummern zugeordnet sind. In Cluster Nr. 109 z.B. stammen alle Patches aus der Bindetasche des Enzyms HIV-1 Reverse Transkriptase, welches als RNA-gerichtete DNA Polymerase und als HIV-1 Retropepsin klassifiziert ist.

Die Abbildungen 6.17 und 6.18 zeigen die Patches und deren Überlagerung eines der drei gemischten Cluster, d.h. Cluster mit mehr als zwei EC-Nummern bzw. mehr als 10% einer zweiten EC-Nummer. Es ist deutlich zu sehen, dass die geometrische Gestalt der Patches in diesem Clustern sehr ähnlich ist. Auch die physikochemischen Eigenschaften der Patches desselben Clusters sind sich sehr ähnlich.

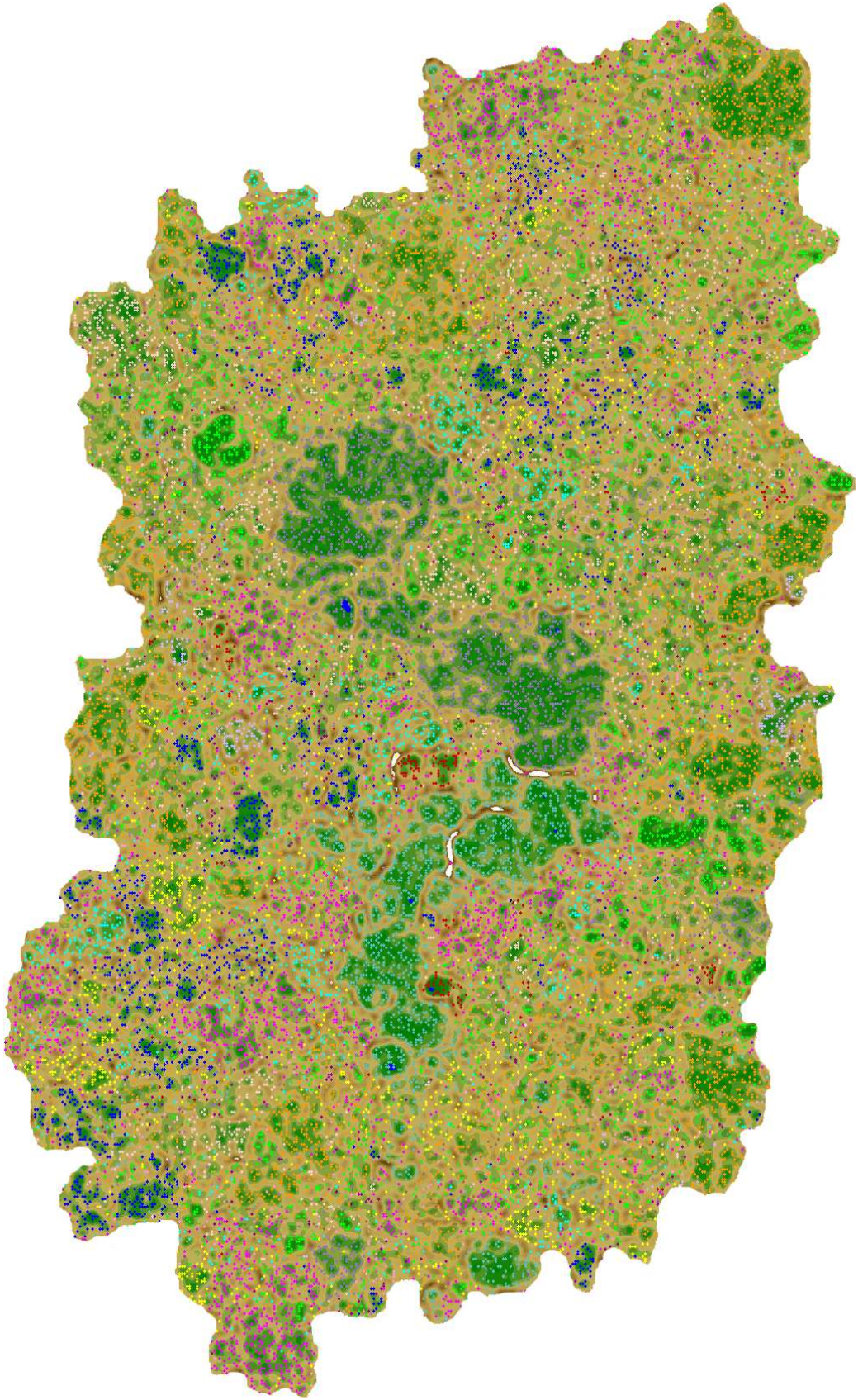


Abbildung 6.9.: U-Matrix zu den durch Waveletkoeffizienten charakterisierten 29684 Oberflächenpatches des Testdatensatzes *DS T1*. Die *Bestmatches* sind entsprechend der verschiedenen EC-Nummern eingefärbt. Die ESOM-Clusterung ergibt 332 Cluster.

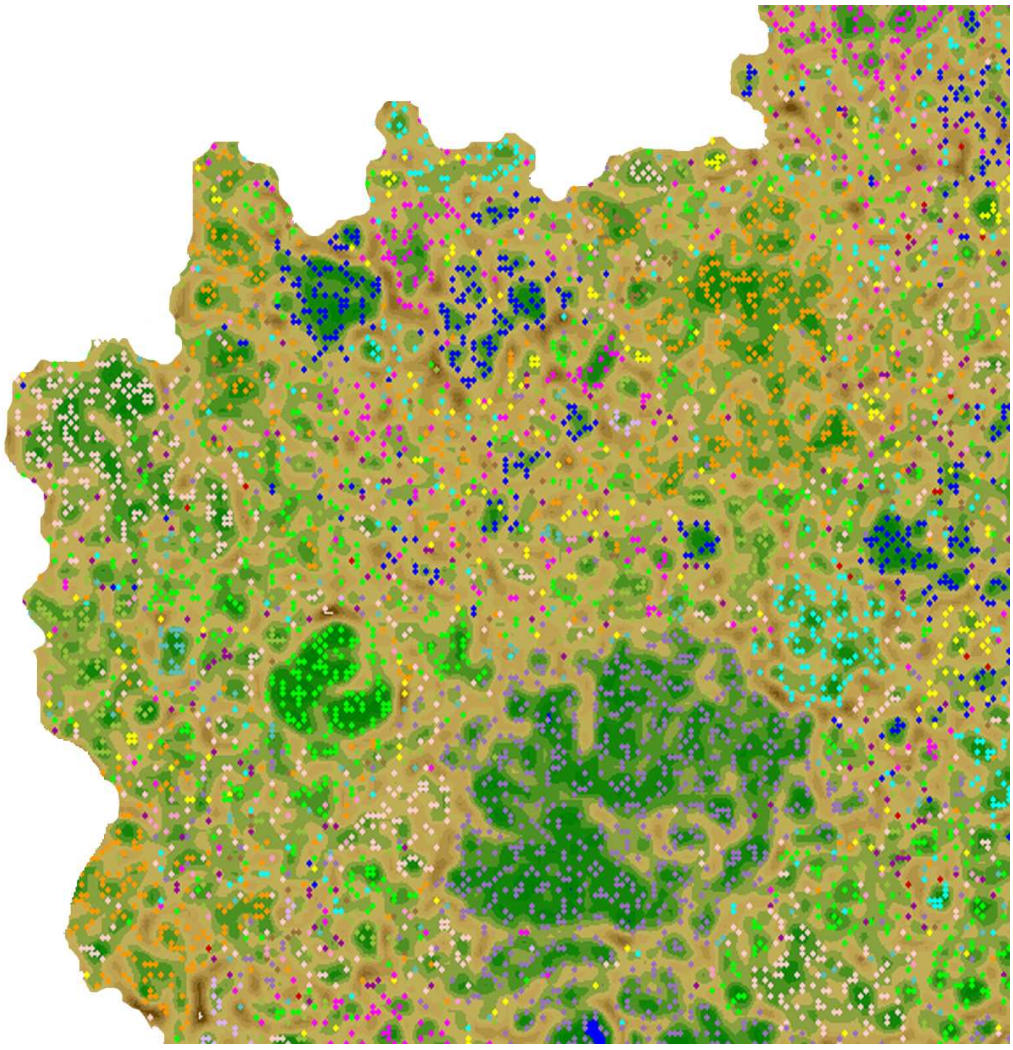


Abbildung 6.10.: Ausschnitt 1 der U-Matrix zu den durch Wavelets charakterisierten Oberflächenpatches. Die *Bestmatches* sind nach den verschiedenen EC-Nummern eingefärbt.

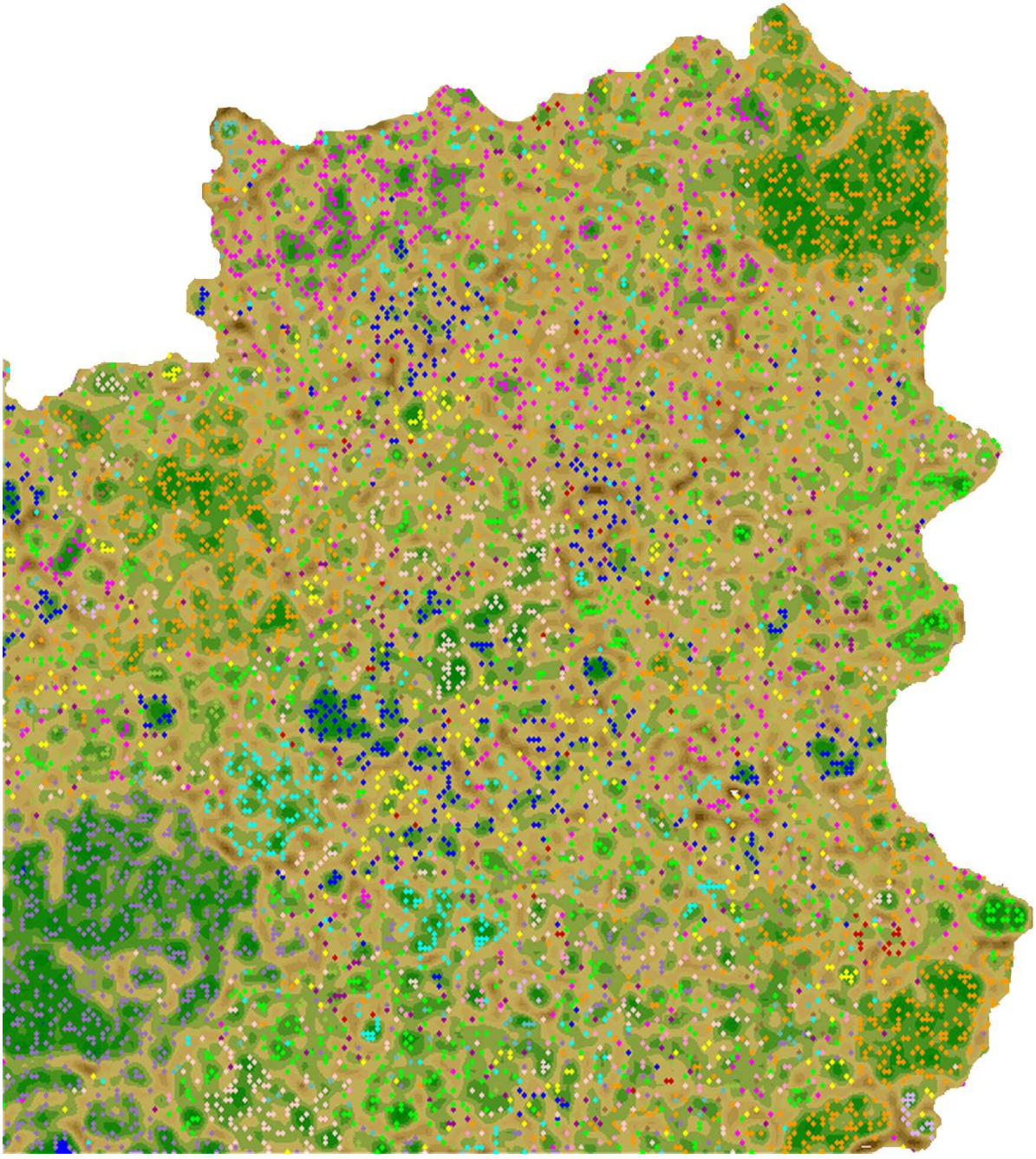


Abbildung 6.11.: Ausschnitt 2 der U-Matrix zu den durch Wavelets charakterisierten Oberflächenpatches. Die *Bestmatches* sind nach den verschiedenen EC-Nummern eingefärbt.

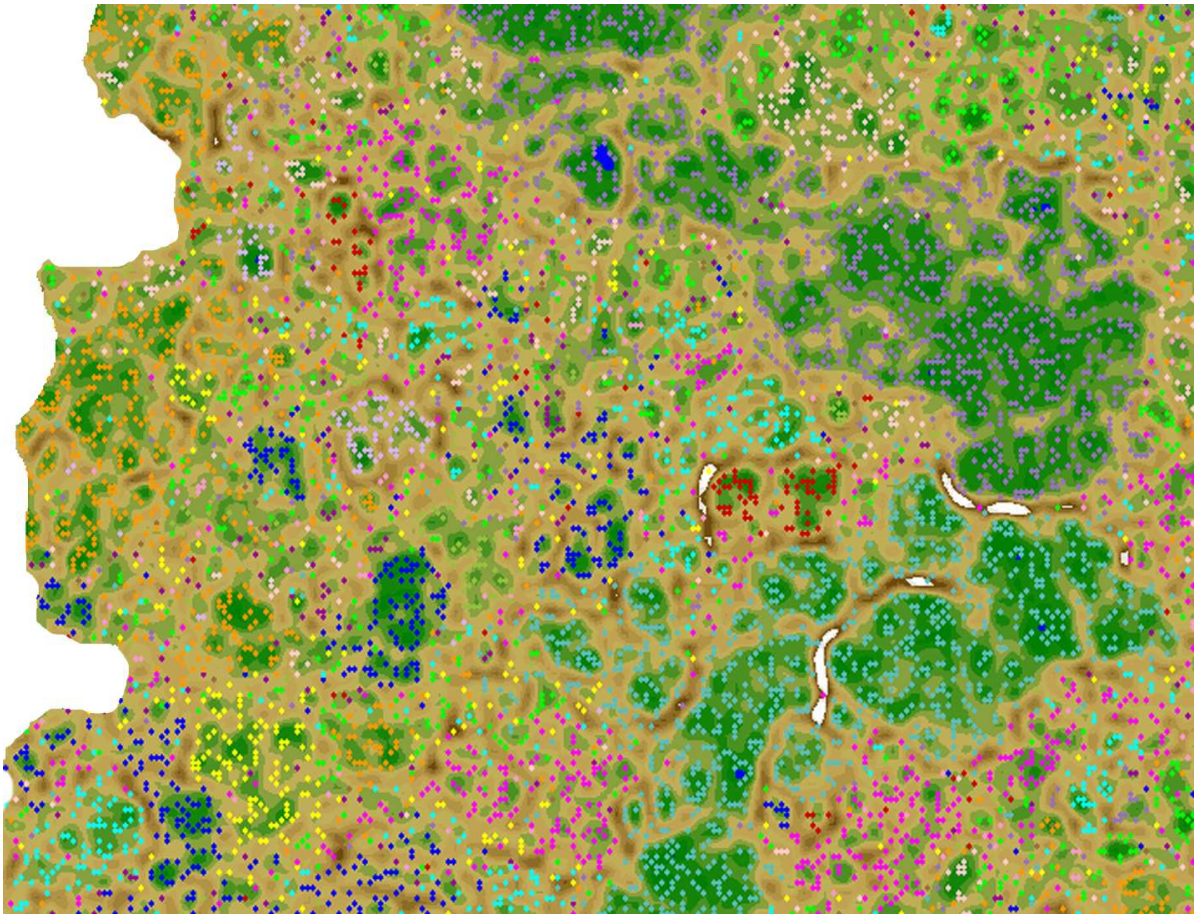


Abbildung 6.12.: Ausschnitt 3 der U-Matrix zu den durch Wavelets charakterisierten Oberflächenpatches. Die *Bestmatches* sind nach den verschiedenen EC-Nummern eingefärbt.

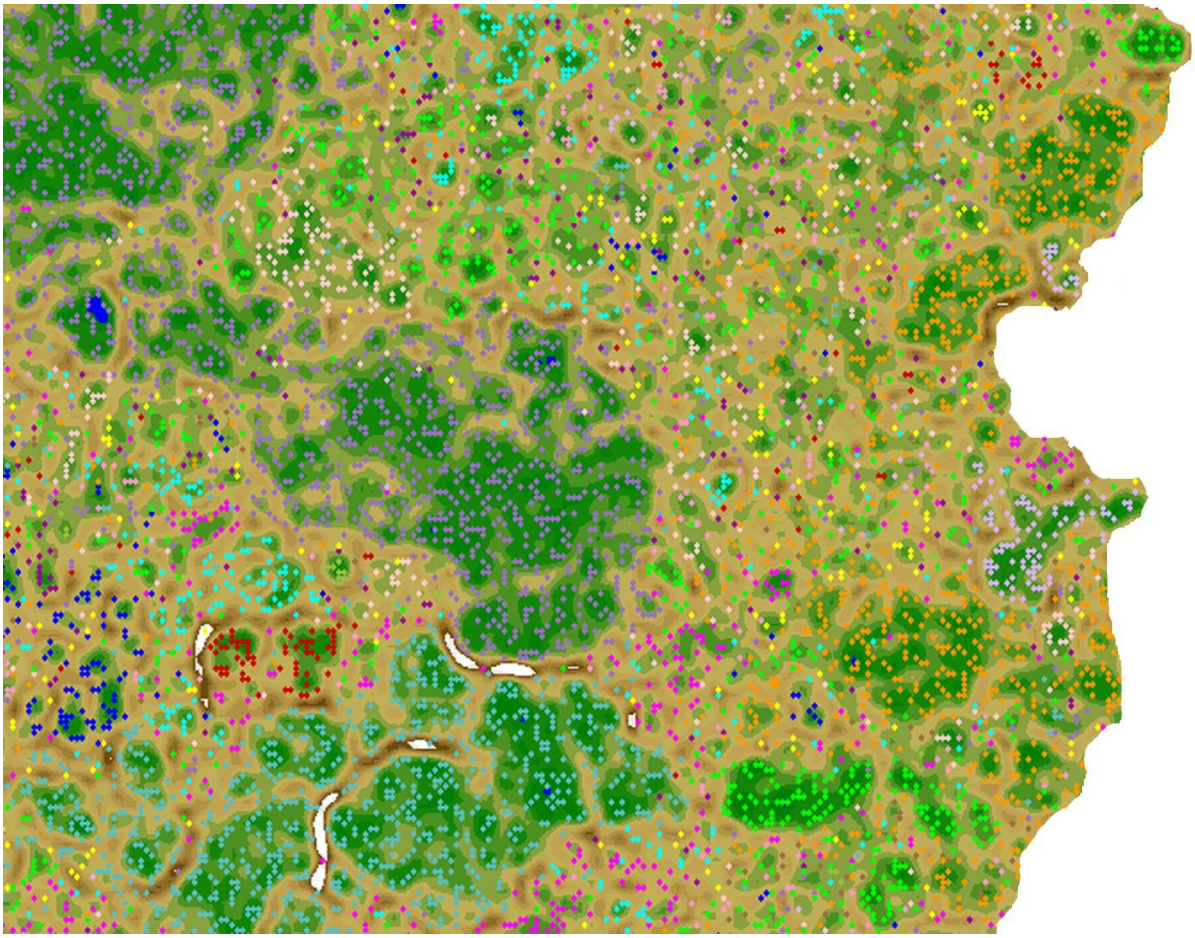


Abbildung 6.13.: Ausschnitt 4 der U-Matrix zu den durch Wavelets charakterisierten Oberflächenpatches. Die *Bestmatches* sind nach den verschiedenen EC-Nummern eingefärbt.

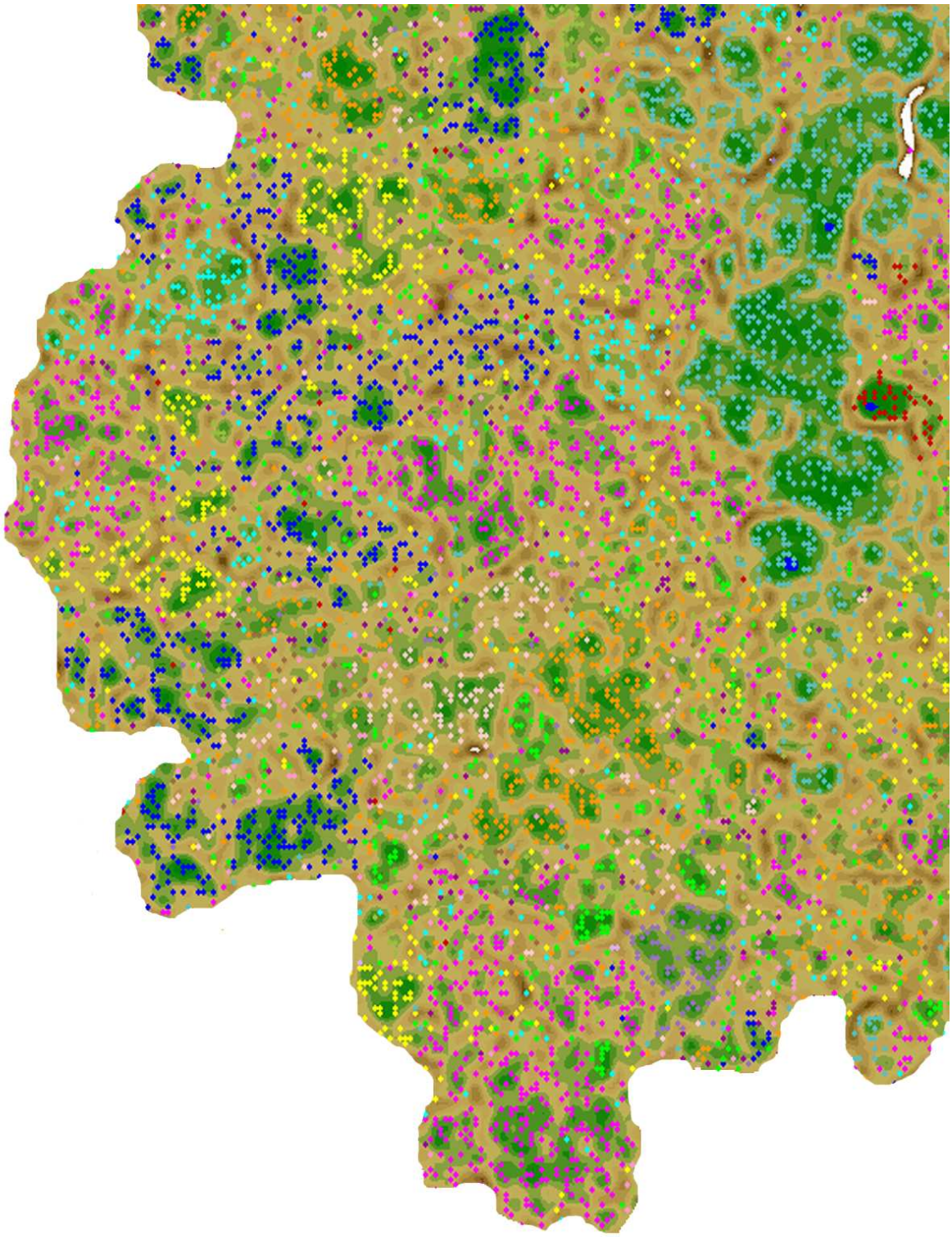


Abbildung 6.14.: Ausschnitt 5 der U-Matrix zu den durch Wavelets charakterisierten Oberflächenpatches. Die *Bestmatches* sind nach den verschiedenen EC-Nummern eingefärbt.

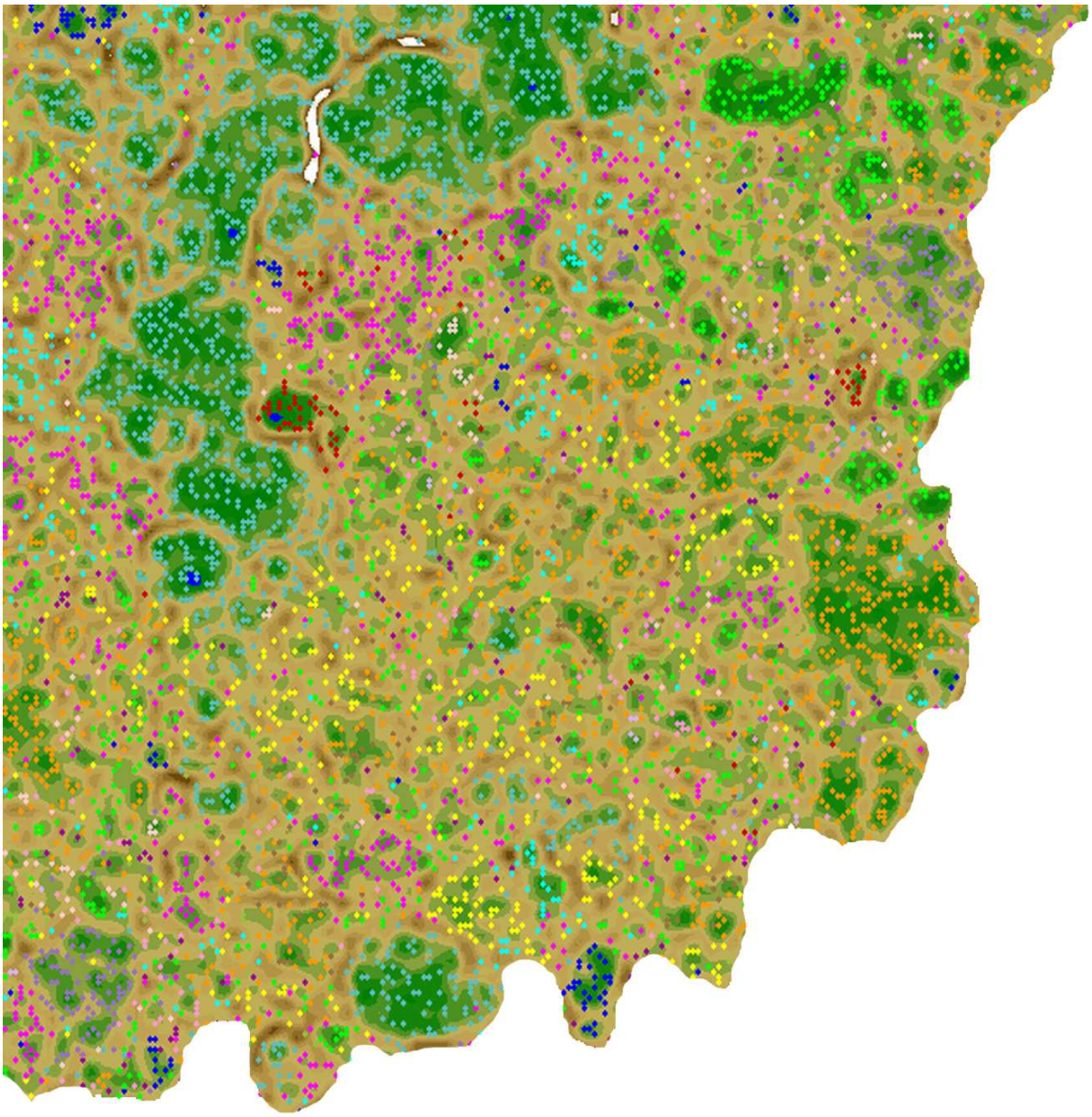


Abbildung 6.15.: Ausschnitt 6 der U-Matrix zu den durch Wavelets charakterisierten Oberflächenpatches. Die *Bestmatches* sind nach den verschiedenen EC-Nummern eingefärbt.

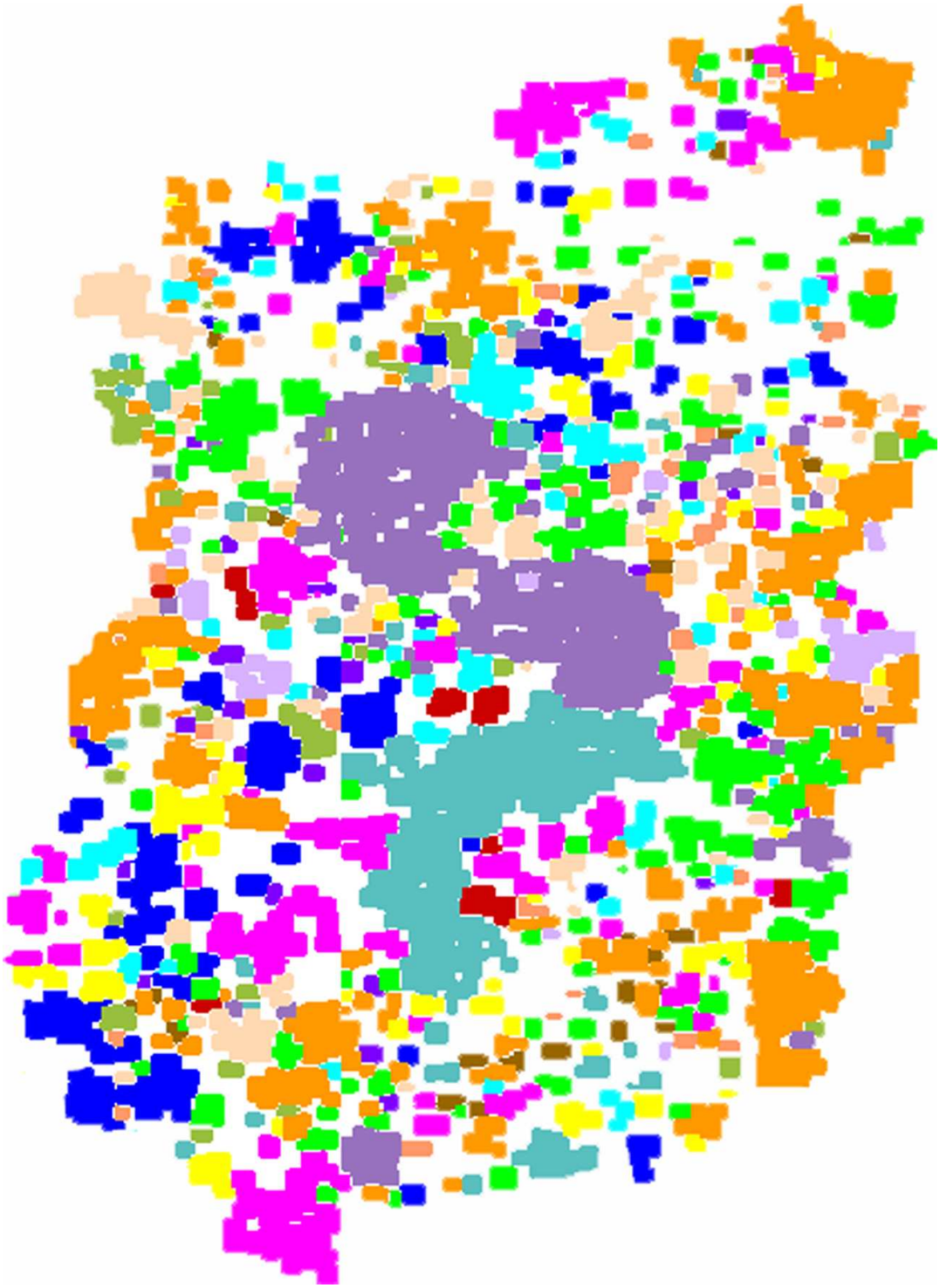
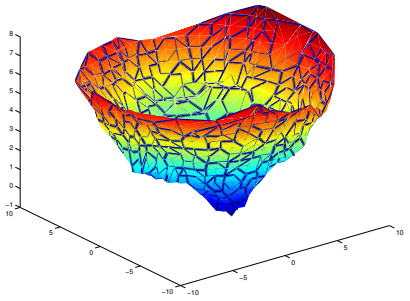
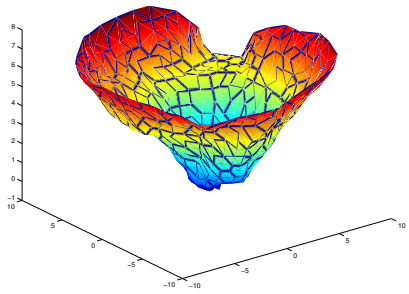


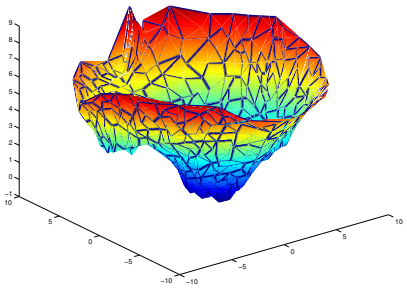
Abbildung 6.16.: Politische Karte zu den durch Waveletkoeffizienten charakterisierten 29684 Oberflächenpatches des Testdatensatzes *DS T1*. Die Regionen sind entsprechend der verschiedenen EC-Nummern der zugehörigen Bestmatches eingefärbt.



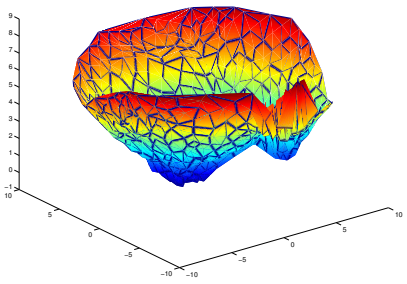
(a) Patch Nr. 1



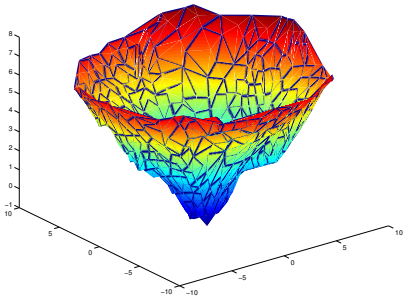
(b) Patch Nr. 2



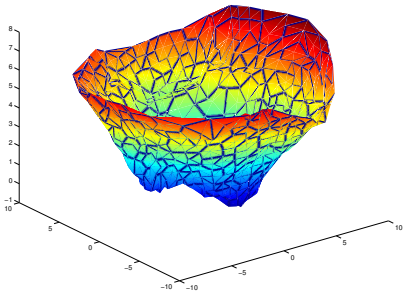
(c) Patch Nr. 3



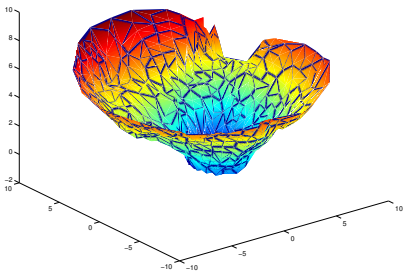
(d) Patch Nr. 4



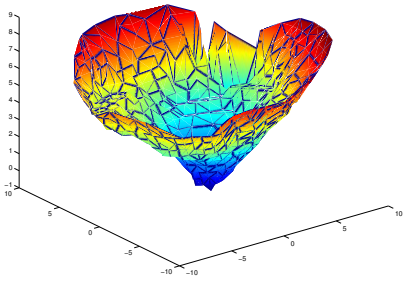
(e) Patch Nr. 5



(f) Patch Nr. 6



(g) Patch Nr. 7



(h) Patch Nr. 8

Abbildung 6.17.: Patches des Cluster Nr. 119 des Testdatensatzes *DS T1*.

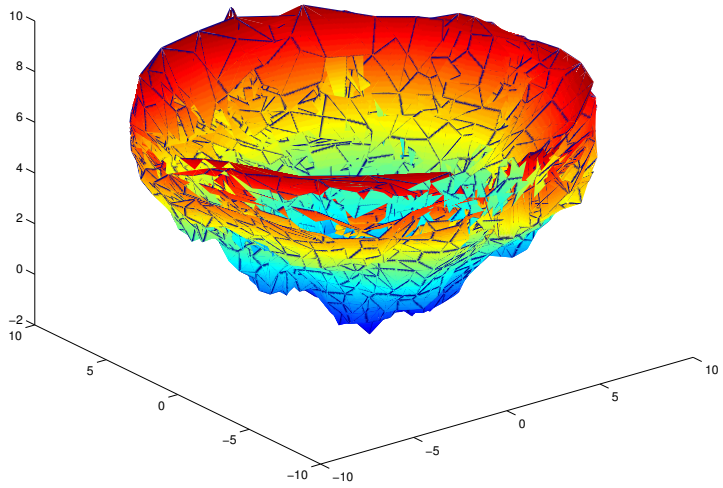


Abbildung 6.18.: Überlagerte Patches des Cluster Nr. 119 des Testdatensatzes *DS T1*.

Ergebnisse für den Datensatz aus Bindetaschen von Endopeptidasen DS T2

Mit diesem Datensatz aus Bindetaschen dreier Familien von Endopeptidasen sollte gezeigt werden, dass der Algorithmus sowohl Gemeinsamkeiten zwischen Bindetaschen unterschiedlicher EC-Nummern sowie auch deren Unterschiede richtig detektiert und clustert. Die 283 Bindetaschen sind in 2930 Patches zerlegt worden und für diese sind die geometrischen und physikochemischen Deskriptoren ermittelt worden. Zunächst ist eine ESOM nur mit den aufbereiteten Deskriptoren verschiedener Strukturen von Trypsin und Thrombin trainiert worden. Die zugehörige U-Matrix ist in Abbildung 6.19 dargestellt. Alle *Bestmatches*, die zu Trypsin gehören, sind in gelb dargestellt. Diejenigen von Thrombin sind blau eingefärbt. Die U-Matrix zeigt Gebiete mit niedrigen U-Höhen, in denen *Bestmatches* beider Strukturen liegen. Das sind Patches aus den Regionen der Bindetaschen beider Enzyme, die sehr ähnlich zueinander sind. Die U-Matrix zeigt aber auch solche Regionen mit Patches nur von einem der beiden Enzyme. Die zugehörigen Patches entsprechen den Teilbereichen der Bindetasche, die in beiden Enzymen verschieden sind.

Die U-Matrix für den gesamten Datensatz ist in Abbildung 6.20 abgebildet. Die *Best-*

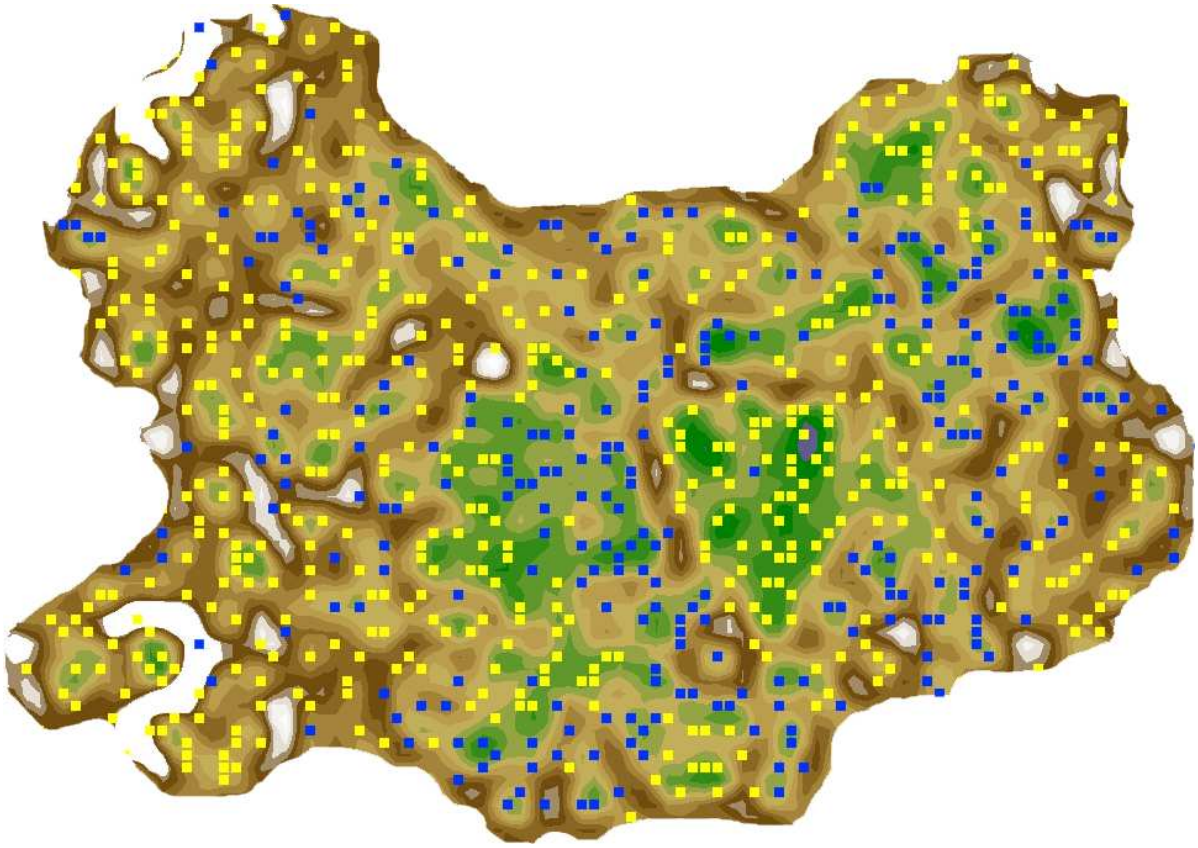


Abbildung 6.19.: U-Matrix nur mit den 45 Deskriptoren der 847 Patches von Trypsin (gelb) und Thrombin (blau).

matches sind entsprechend der EC-Nummern der zugehörigen Enzyme eingefärbt. Dabei sind solche einer Familie mit ähnlichen Farben eingefärbt. Alle Patches von Serinproteasen sind in Gelb-Rot-Orange-Tönen, die der Cysteinproteasen in verschiedenen Grün- und die Aspartyl-Proteasen in Blau-Abstufungen eingefärbt. Die ESOM-Clusterung ergibt 111 Cluster. Auch diese U-Matrix zeigt sowohl Cluster mit *Bestmatches* von Enzymen verschiedener, aber zur gleichen Familie gehörender Enzyme, die auch Ähnlichkeiten in ihren Taschen aufweisen, als auch Regionen mit Patches verschiedener Strukturen mit der gleichen EC-Nummer.

In Abbildung 6.21 ist eine politische Karte dieser U-Matrix zu sehen. Zusammenhängende Bereiche auf der U-Matrix mit niedrigen U-Höhen sind entsprechend der EC-Nummern der Bestmatches eingefärbt worden. Die Farbcodierung ist dieselbe wie in Abbildung 6.20. Der Hintergrund der U-Matrix ist ausgeblendet worden.

Alle in den folgenden Abbildungen dargestellten Überlagerungen der zu den Patches in einem Cluster gehörenden Aminosäuren und Pseudozentren sind mit dem *Cliq5*-Algorithmus von Schmitt et al. [2002] erstellt worden. Für eine Referenztasche sind die den Oberflächenpunkten des Patches zugeordneten Pseudozentren bestimmt worden. Anhand dieser Pseudozentren ist eine Überlagerung der Strukturen berechnet worden. Dargestellt ist der durch den *Cliq5*-Algorithmus gegebene zugehörige Oberflächenbereich der Referenztasche sowie die für die Überlagerung verwendeten Pseudozentren und zugehörigen Aminosäuren aller Taschen. Wegen der Klarheit der Darstellung sind die Oberflächenbereiche der anderen Taschen ausgeblendet worden. Sie sind dem Oberflächenbereich der Referenztasche aber sehr ähnlich. Die Anzahl der Pseudozentren kann von der vorgegebenen Anzahl abweichen, wenn für eine gute Überlagerung weniger Pseudozentren benötigt werden. Somit entspricht der dargestellte Oberflächenbereich nicht exakt den Oberflächenpatches in dem entsprechenden Cluster.

Die Cluster, die nur Strukturen zu derselben EC-Nummer enthalten, sind solche Bereiche in den Bindetaschen, die sich auch innerhalb der einzelnen Subfamilien unterscheiden. So enthalten z.B. Cluster, die nur Patches aus den Thrombinstrukturen enthalten, die Regionen um den 60er-Loop des Thrombin, der in den anderen Serinproteasen fehlt. Die Abbildung 6.22 zeigt als Beispiel die Überlagerung zweier Thrombinstrukturen und die zugehörigen Oberflächenpunkte.

Die Cluster, die Patches zu verschiedenen Strukturen einer Subfamilie enthalten, entsprechen den katalytischen Bereichen der Bindetaschen. Diese sind innerhalb einer Subfamilie weitgehend konserviert und landen deshalb im gleichen Cluster. Zu jeder der im Datensatz *DS T2* verwendeten Subfamilien existieren solche Cluster. Im Folgenden wer-

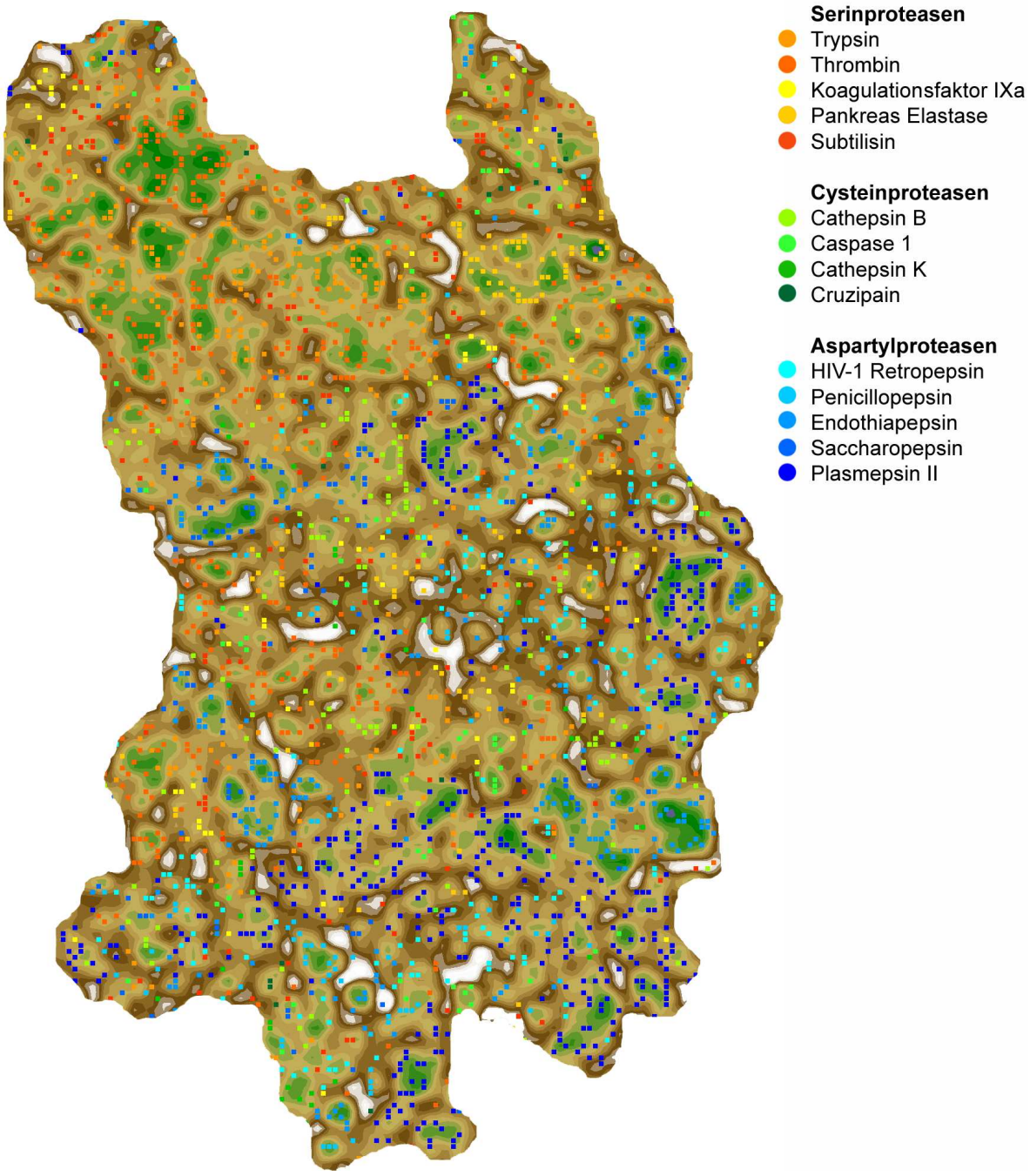


Abbildung 6.20.: U-Matrix der 2930 Oberflächenpatches der Endopeptidasen. Die *Best-matches* sind entsprechend der EC-Nummern und Familien gefärbt. Serinproteasen sind in Gelb-Rot-Orange-, Cysteinproteasen in Grün und Aspartylproteasen in Blau-Tönen gefärbt. Die ESOM-Clusterung ergibt 111 Cluster.

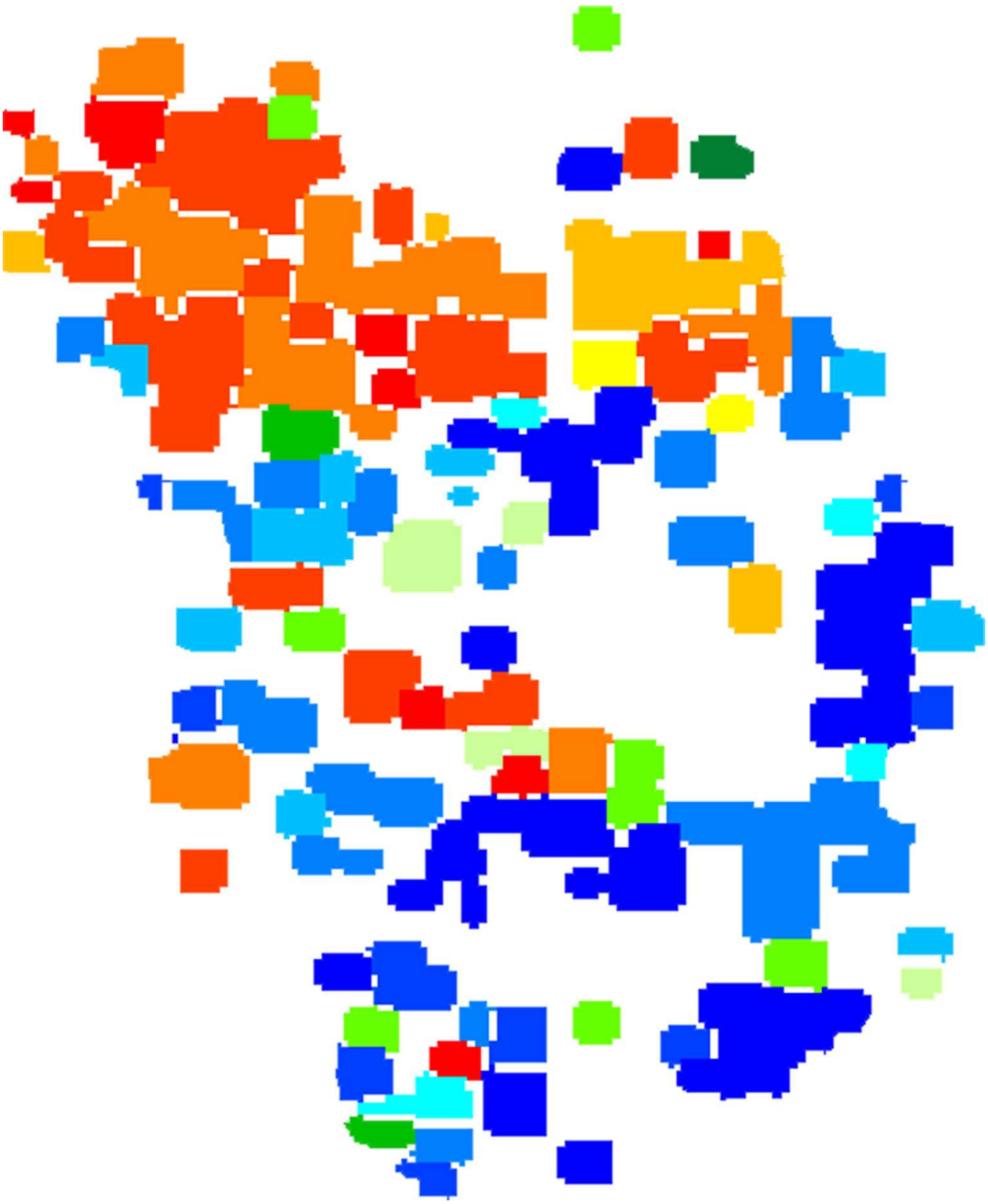


Abbildung 6.21.: Politische Karte zu den durch Waveletkoeffizienten charakterisierten 2930 Oberflächenpatches des Testdatensatzes *DS T2* aus Endopeptidasen. Die Regionen sind entsprechend der verschiedenen EC-Nummern der zugehörigen Bestmatches eingefärbt. Serinproteasen sind in Gelb-Rot-Orange-, Cysteinproteasen in Grün und Aspartylproteasen in Blau-Tönen gefärbt.

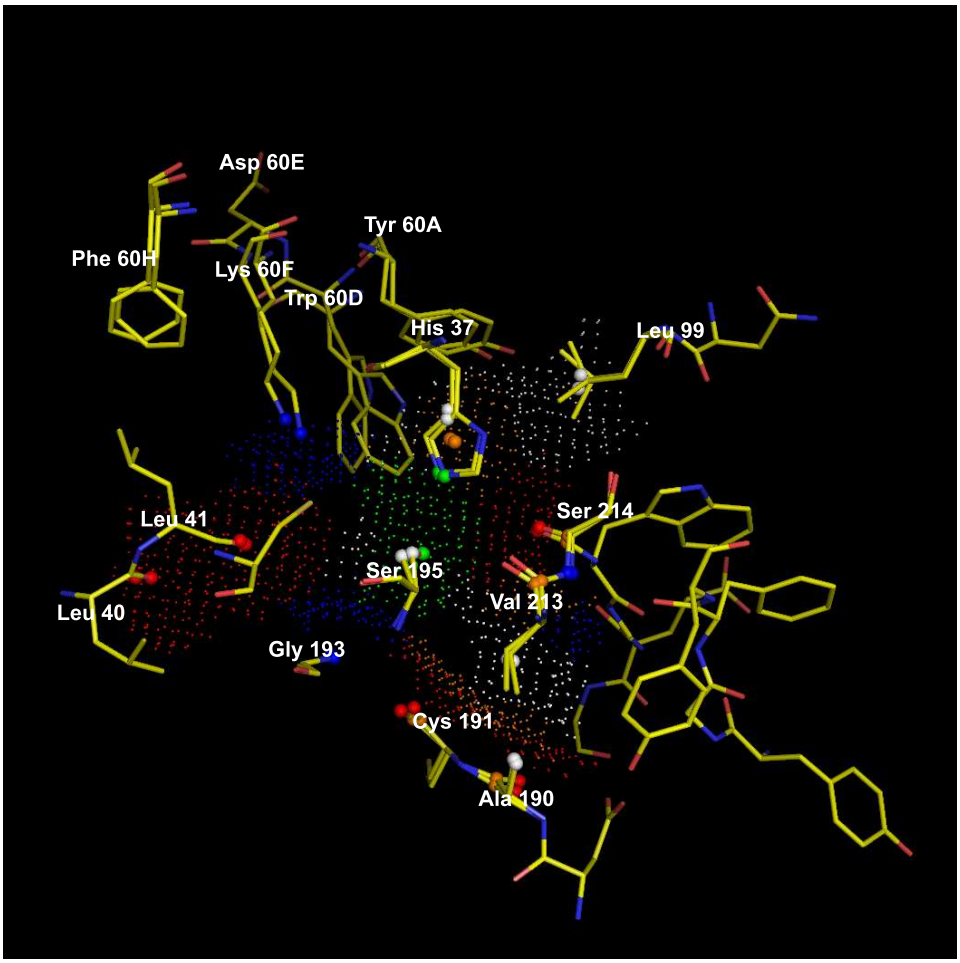


Abbildung 6.22.: Überlagerung zweier Strukturen eines Clusters, welches ausschließlich Patches verschiedener Thrombinstrukturen enthält. Die Patches in diesem Cluster gehören zu dem für Thrombin charakteristischen 60er-Loop. Für die Überlagerung ist durch den *Cliq5*-Algorithmus allerdings ein kleinerer Bereich verwendet worden.

den die Inhalte einiger Beispiele solcher Cluster vorgestellt, sie sind auf der U-Matrix in Abbildung 6.23 gekennzeichnet. Jede Abbildung zeigt a) das Cluster auf der U-Matrix und b) den entsprechenden Oberflächenbereich, die zugehörigen Aminosäuren und Pseudozentren sowie teilweise die gebundenen Liganden.

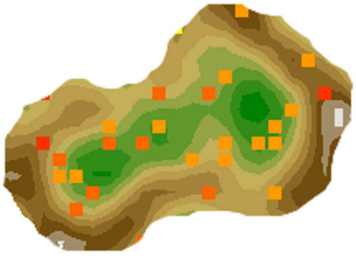
Die Abbildungen 6.24 und 6.25 zeigen beispielhaft die Inhalte zweier Cluster mit Patches aus Serinproteasen. Der dargestellte Bereich der Bindetaschen umfasst die S1-Tasche und Teile der katalytischen Triade. Diese Bereiche der Bindetasche sind bei den Serinproteasen weitgehend konserviert. Das in Abbildung 6.25 gezeigte Cluster beinhaltet ein Patch einer Subtilisin-Struktur. Subtilisin weist eine andere Faltung als die Serinproteasen der Trypsin-Familie auf, hat jedoch den Bereich der S1-Tasche und der katalytischen Triade mit den anderen Strukturen gemein. Die Aminosäuren des Subtilisin sowie deren Bezeichnungen sind in der Abbildung grün angegeben. Die Mitglieder der Trypsin-Familie sind in gelb mit weißen Benennungen der Aminosäuren dargestellt. Die *Bestmatches* zu Patches aus der Tasche von Subtilisin sind dunkelrot gefärbt. Der noch in dem Cluster liegende *Bestmatch* befindet sich ganz am Clusterrand, was sich durch die unterschiedliche Faltung der Strukturen ergibt.

Die Abbildungen 6.26 und 6.27 zeigen die Überlagerung der zu den Patches gehörenden Bereiche der Bindetaschen zweier Cluster, die verschiedene Strukturen der Unterfamilie der Aspartylproteasen enthalten. Zusätzlich zu den überlagerten Aminosäuren sind auch die verschiedenen gebundenen Liganden in den unterschiedlichen Aspartylproteasen dargestellt. Der Teil der Liganden, der nahe dem Bereich der überlagerten Oberflächenpatches des Clusters liegt, besitzt eine sehr ähnliche Struktur. Die Abbildungen zeigen, dass ähnliche Oberflächenpatches innerhalb eines Clusters auch Rückschlüsse auf ähnliche Ligandenbausteine in einem Teil der Bindetasche zulassen.

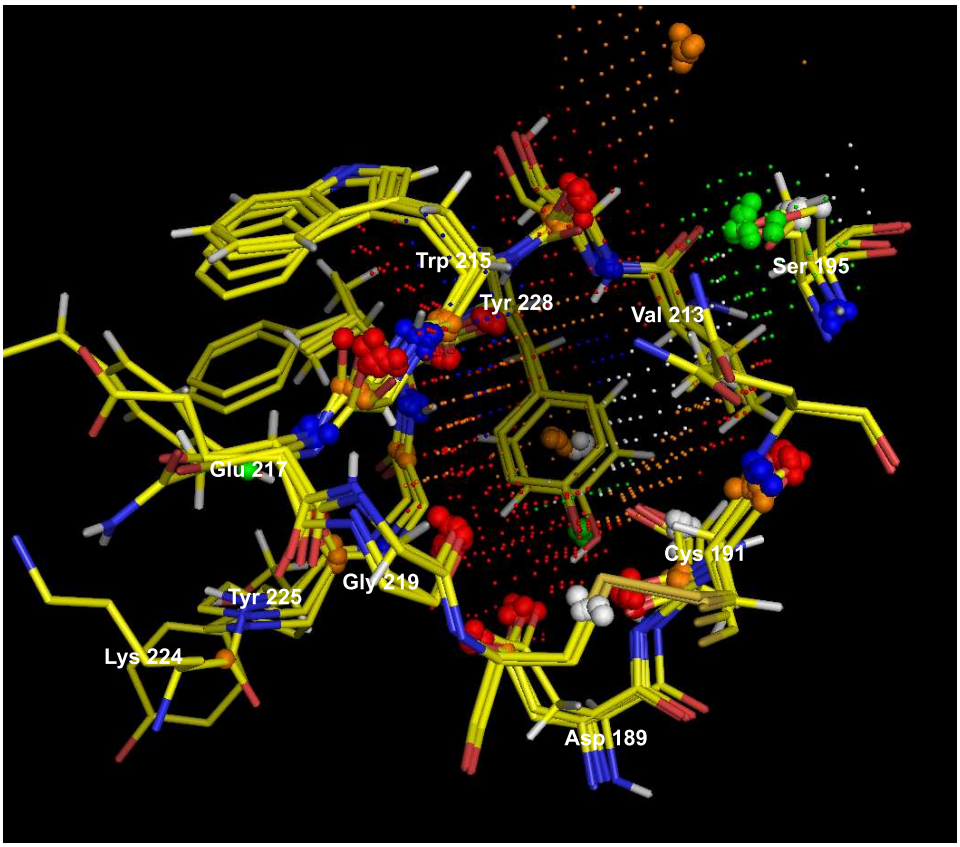
Solche Regionen der Karte, in denen *Bestmatches* zu Enzymen verschiedener Subfamilien zu liegen kommen, entsprechen zumeist Bergketten oder Hochtälern mit großen U-Höhen. Die zugehörigen Daten sind weiter voneinander entfernt als diejenigen innerhalb eines Bereich mit geringen U-Höhen. Visualisiert man sich die Patches aus Hochtälern, so weisen sie zwar einige Ähnlichkeiten auf, entsprechen aber keinen katalytischen Regionen der Bindetaschen. Es sind eher molekulare Erkennungsbereiche, die verwandte Bausteine in den Liganden über ihre Oberflächen binden. Da die U-Höhen in diesen Bereichen aber relativ hoch sind, bleiben diese bei der ESOM-Clusterung unberücksichtigt und somit unklassifiziert.

Für jede Struktur ist durch die Clusterzugehörigkeit der verschiedenen Patches eine Signatur gegeben, die eine eindeutige Funktionszuweisung der Bindetasche erlaubt. Für

Abbildung 6.23.: U-Matrix der 2930 Oberflächenpatches der Endopeptidasen (siehe Abb. 6.20). Die *Bestmatches* sind entsprechend der EC-Nummern und Familien gefärbt. Serinproteasen sind in Gelb-Rot-Orange-, Cysteinproteasen in Grün und Aspartylproteasen in Blau-Tönen gefärbt. Die Cluster, deren Inhalte als Beispiele gezeigt werden, sind schwarz umrandet.



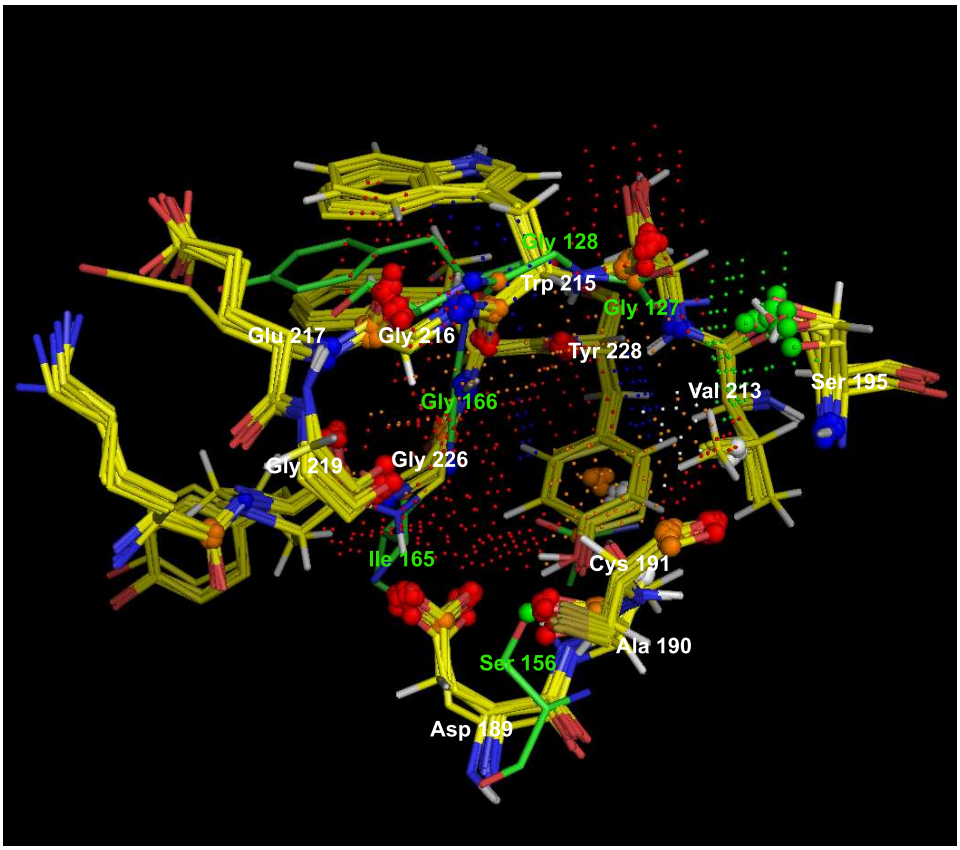
(a) Cluster mit Patches zu Serinproteasen



(b) Zugehörige Überlagerung der Aminosäuren

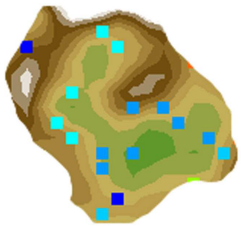
Abbildung 6.24.: Oberflächenpatches und zugehörige Aminosäuren verschiedener Strukturen Trypsin-ähnlicher Serinproteasen eines Clusters der U-Matrix zum Datensatz *DS T2*. Es ist ein Bereich des katalytischen Ser 195 zusammen mit der S1-Tasche zu erkennen.

(a) Cluster mit Patches zu Serinproteasen

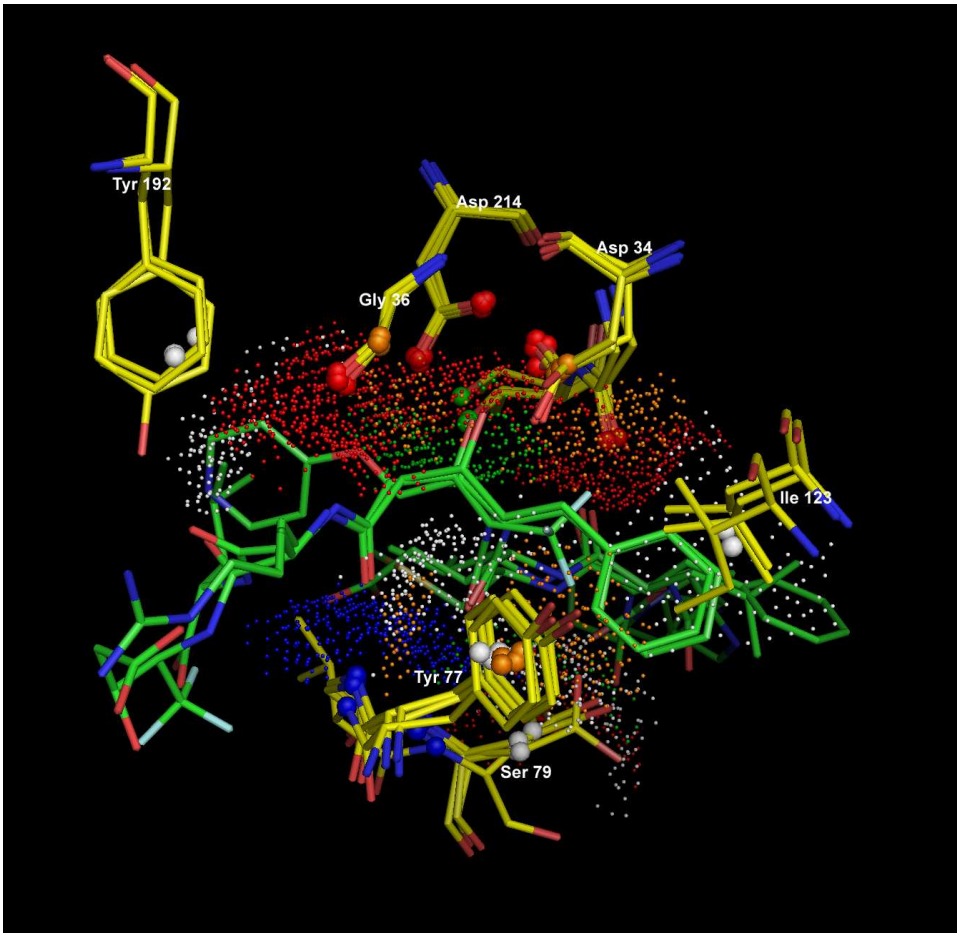


(b) Zugehörige Überlagerung der Aminosäuren

Abbildung 6.25.: Oberflächenpatches und zugehörige Aminosäuren verschiedener Strukturen Trypsin-ähnlicher Serinproteasen und Subtilisin eines Clusters der U-Matrix zum Datensatz *DS T2*. Es ist ein Bereich des katalytischen Ser 195 zusammen mit der S1-Tasche zu erkennen. Die Struktur von Subtilisin und die zugehörigen Aminosäurebezeichnungen sind grün, die der anderen Strukturen sind gelb mit weißen Benennungen dargestellt.

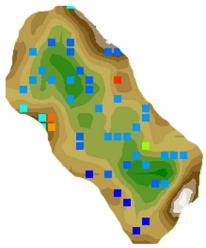


(a) Cluster mit Patches zu Aspartylproteasen



(b) Zugehörige Überlagerung der Aminosäuren

Abbildung 6.26.: Oberflächenpatches, zugehörige Aminosäuren und gebundene Liganden verschiedener Strukturen von Plasmepsin II und Endothiapepsin eines Clusters der U-Matrix zum Datensatz *DS T2*. Die Aminosäuren sind gelb, die Liganden sind grün gefärbt. Die Patches umfassen den Bereich der katalytischen Diade Asp 34 und Asp 134 sowie Tyr 77. Die in den Bindetaschen liegenden Liganden sind in diesem Bereich der katalytischen Zentren nahezu identisch.



(a) Cluster mit Patches zu Aspartylproteasen

(b) Zugehörige Überlagerung der Aminosäuren

Abbildung 6.27.: Oberflächenpatches, zugehörige Aminosäuren und gebundene Liganden verschiedener Strukturen von Penicillopepsin und Endothiapepsin eines Clusters der U-Matrix zum Datensatz *DS T2*. Die Aminosäuren sind gelb, die Liganden sind grün gefärbt. Die Patches umfassen den Bereich der katalytischen Diade Asp 33 und Asp 133 sowie Tyr 75. Die in den Bindetaschen liegenden Liganden sind in diesem Bereich der katalytischen Zentren nahezu identisch.

Thrombin z.B. gehört zu diesen Clustern eines, welches zur S1-Tasche gehörige Patches enthält. Diese Clusterzugehörigkeit weist Thrombin den Serinproteasen zu. Zusätzlich gehört zu dieser Signatur ein Cluster mit zum 60er-Loop gehörenden Patches, welches Thrombin unter den Serinproteasen eindeutig identifiziert.

Ergebnisse der Projektion des Datensatzes DS P auf die mit dem Datensatz DS T2 trainierte ESOM

Die 118 Bindetaschen des Datensatzes *DS P* sind in 1376 Patches zerlegt worden. Diese sind ausgerichtet worden und die Deskriptoren sind berechnet und aufbereitet worden. Zu den Deskriptionsvektoren der Patches sind die Bestmatches auf der mit dem Datensatz *DS T2* antrainierten ESOM bestimmt worden. Es ist keine neue Karte berechnet worden. Die Abbildung 6.28 zeigt die U-Matrix mit den Bestmatches des Datensatzes *DS T2* und *DS P*.

Die *Bestmatches* sind entsprechend der EC-Nummern der zugehörigen Enzyme eingefärbt. Dabei sind solche einer Familie mit ähnlichen Farben eingefärbt. Alle Patches von Serinproteasen sind in Rot-Tönen, die der Cysteinproteasen in Grün- und die Aspartylproteasen in Blau-Tönen eingefärbt. Dabei sind die *Bestmatches* des Datensatzes *DS T2* in hellen Tönen, die des Datensatzes *DS P* in dunklen Tönen gefärbt.

Die U-Matrix zeigt, dass die *Bestmatches* einiger Patches in den entsprechenden Clustern mit *Bestmatches* derselben Familie liegen, während aber solche von verschiedenen Familien wenig Ähnlichkeit haben. Nicht alle der projizierten Datenpunkte liegen in einem Cluster, viele liegen in Regionen mit großen U-Höhen und werden keinem Cluster zugeordnet. Diese Patches gehören zu den Teilbereichen der Bindetasche, die sich von den anderen Bindetaschen der Enzymfamilie unterscheidet. Da die Karte nicht neu antrainiert worden ist, liegen diese *Bestmatches* auf den Gebirgszügen der U-Matrix, was eine große Distanz zu den benachbarten Datenpunkten bedeutet.

Die Abbildung 6.29 zeigt einen Ausschnitt dieser U-Matrix, der ein Cluster für jede der drei Familien zeigt. Es existieren für jede der Familien Patches des Datensatzes *DS P*, deren *Bestmatches* in diesen Clustern liegen.

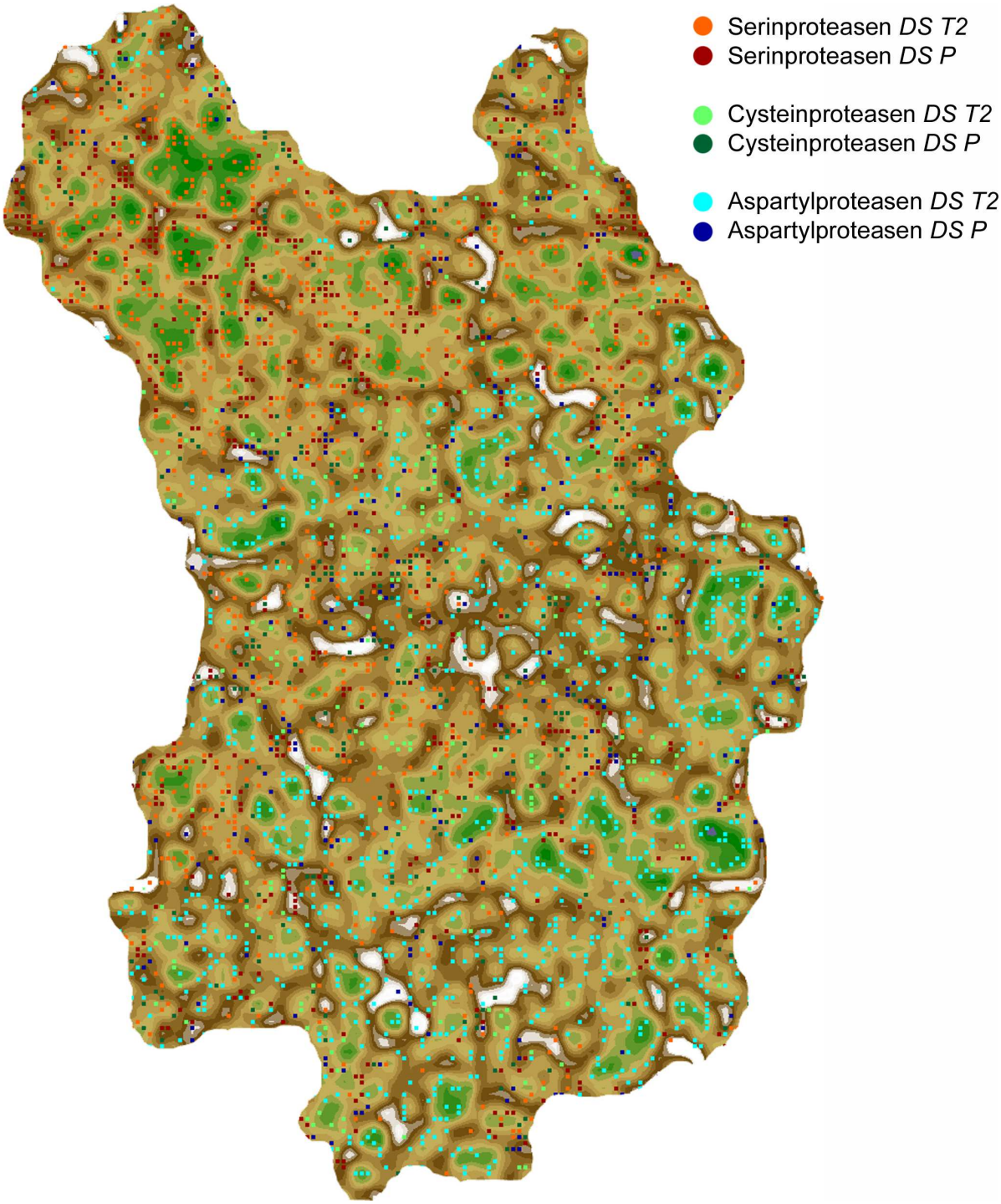


Abbildung 6.28.: U-Matrix trainiert mit den Oberflächenpatches der Endopeptidasen des Datensatzes *DS T2* und den darauf projizierten Patches des Datensatzes *DS P*. Die *Bestmatches* sind entsprechend der EC-Nummern und Familien gefärbt. Serinproteasen sind in Rot-, Cysteinproteasen in Grün und Aspartylproteasen in Blau-Tönen gefärbt. Die *Bestmatches* des Datensatzes *DS T2* sind in hellen, die des Datensatzes *DS P* in dunklen Tönen gefärbt.

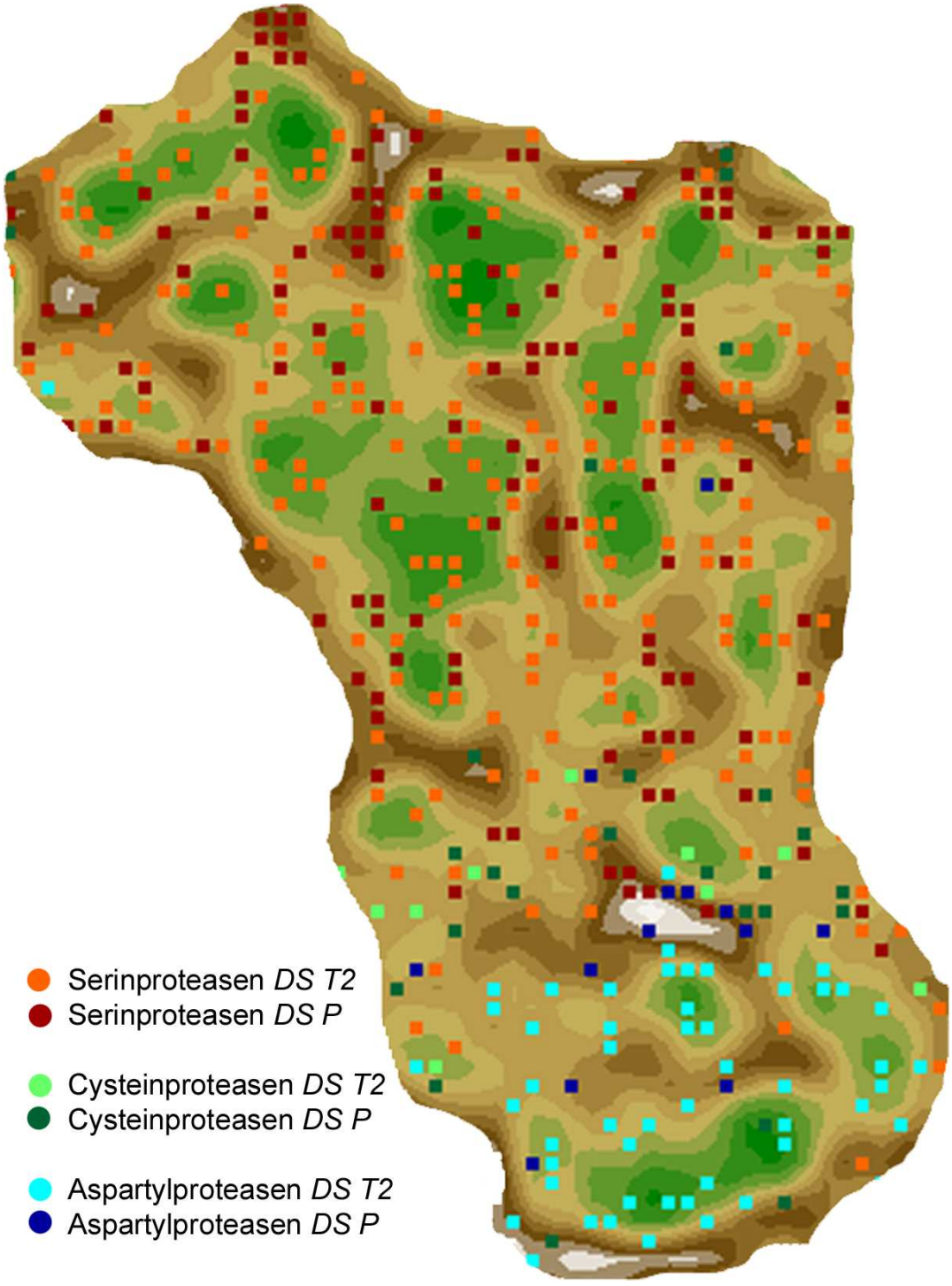


Abbildung 6.29.: Ausschnitt der U-Matrix trainiert mit den Oberflächenpatches der Endopeptidasen des Datensatzes *DS T2* und den darauf projizierten Patches des Datensatzes *DS P*. Die *Bestmatches* sind entsprechend der EC-Nummern und Familien gefärbt. Serinproteasen sind in Rot-, Cysteinproteasen in Grün und Aspartylproteasen in Blau-Tönen gefärbt. Die *Bestmatches* des Datensatzes *DS T2* sind in hellen, die des Datensatzes *DS P* in dunklen Tönen gefärbt.

6.3. Vergleich mit anderen Clusteralgorithmen

Methode zum Vergleich der verschiedenen Clusteralgorithmen

Zum Vergleich der Ergebnisse der ESOM-Clusterung mit denen anderer Clusteralgorithmen ist der Validierungsdatensatz *DS V1* (siehe Kapitel 4.1) herangezogen worden. Dieser Datensatz besteht aus 109 Bindetaschen zu 13 verschiedenen EC-Nummern. Der Datensatz ist so gewählt worden, dass Bindetaschen zu derselben EC-Nummer sehr ähnlich sind, während solche zu unterschiedlichen EC-Nummern keine über den Reaktionsmechanismus oder Ähnlichkeiten der Substrate begründbaren Gemeinsamkeiten aufweisen. Die 109 Bindetaschen sind in 1836 verschiedene Patches der optimalen Größe (siehe Kapitel 5.5) zerlegt worden. Diese Patches sind kanonisch ausgerichtet und für jedes Patch sind die Deskriptionsvektoren beider Verfahren zur Beschreibung der Oberfläche (siehe Kapitel 5.2 und 5.3) berechnet worden.

Da Bindetaschen der Enzyme zu unterschiedlichen EC-Nummern keine trivialen Ähnlichkeiten untereinander aufweisen sollten, haben auch die entsprechenden Patches so gut wie keine Gemeinsamkeiten in ihren Deskriptionsvektoren. Ein direkter Vergleich der vorgegebenen EC-Klassifizierung mit der Clusterung der Patches ist jedoch nicht möglich, da die EC-Nummer eine Vorklassifizierung der gesamten Bindetaschen vornimmt. In dem in dieser Arbeit beschriebenen Ansatz wird jede Bindetasche aber in eine Reihe von Patches zerlegt, so dass für die einzelnen EC-Klassen mehrere Cluster mit den unterschiedlichen Substrukturen der Bindetaschen zu dieser EC-Nummer entstehen.

Eine optimale Clusterung der Patches teilt den Datensatz *DS V1* in Klassen mit Patches zu nur derselben EC-Nummer auf. Ein Maß für die Qualität der Clusterung ist deswegen die Reinheit der Cluster. Unterschieden wird zwischen drei Reinheitsstufen. Reine Cluster enthalten nur Patches zu derselben EC-Nummer, quasi-reine Cluster enthalten mind. 90% zu einer und max. 10% zu einer zweiten EC-Nummer und Mischcluster sind alle übrigen Cluster. Zudem wird noch überprüft, ob zu jeder der EC-Klassen auch mindestens ein reines oder quasi-reines Cluster generiert wird. Die ideale Clusterung würde 100% reine Cluster produzieren, und zu jeder der 13 EC-Klassen würde mindestens ein Cluster mit Patches zu dieser EC-Nummer existieren.

Die Deskriptionsvektoren der Patches sind zum Vergleich der Qualität der ESOM-Clusterung mit den Standard-Cluster-Algorithmen *K-means*, *Single Linkage* und *Ward* (siehe Kapitel 2.1) geclustert worden. Damit die Ergebnisse vergleichbar sind, ist als Anzahl der Cluster die durch die ESOM-Clusterung gegebene Clusteranzahl verwendet

worden. Für die Beschreibung der Patchoberfläche durch Mittelwert und Varianz der Höhenwerte in gleichgroßen Sektoren ergibt die ESOM-Clusterung 101 Cluster, für die Beschreibung der Oberfläche durch Wavelets 52 verschiedene Cluster.

Ergebnisse des Vergleichs mit anderen Clusteralgorithmen

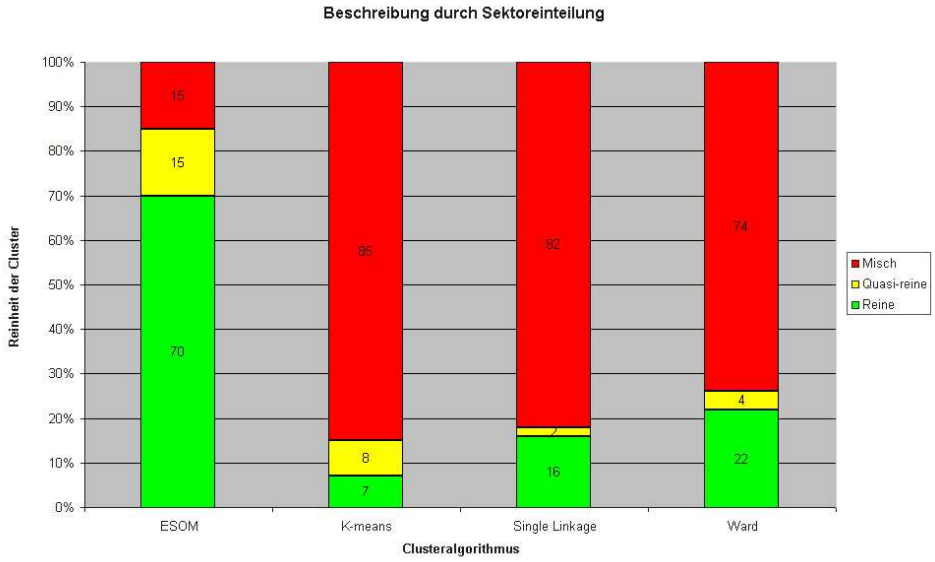
Die Tabelle 6.2 zeigt die Ergebnisse der verschiedenen Verfahren. In der ersten Spalte ist der jeweilige Clusteralgorithmus aufgeführt. In den Spalten zwei bis vier sind die prozentualen Anteile der Daten in reinen, quasi-reinen und gemischten Clustern eingetragen. Die letzte Spalte gibt an, ob für jede der 13 EC-Klassen mindestens ein reiner oder quasi-reiner Cluster existiert.

Tabelle 6.2.: Qualität der ESOM-Clusterung und der Clusteralgorithmen *K-means*, *Single Linkage* und *Ward* für den Datensatz *DS V1* bei optimaler Patchgröße. Angegeben sind die prozentualen Anteile der Daten in reinen, quasi-reinen und gemischten Clustern. Die letzte Spalte gibt an, ob mindestens ein reiner oder quasi-reiner Cluster für jede EC-Nummer existiert.

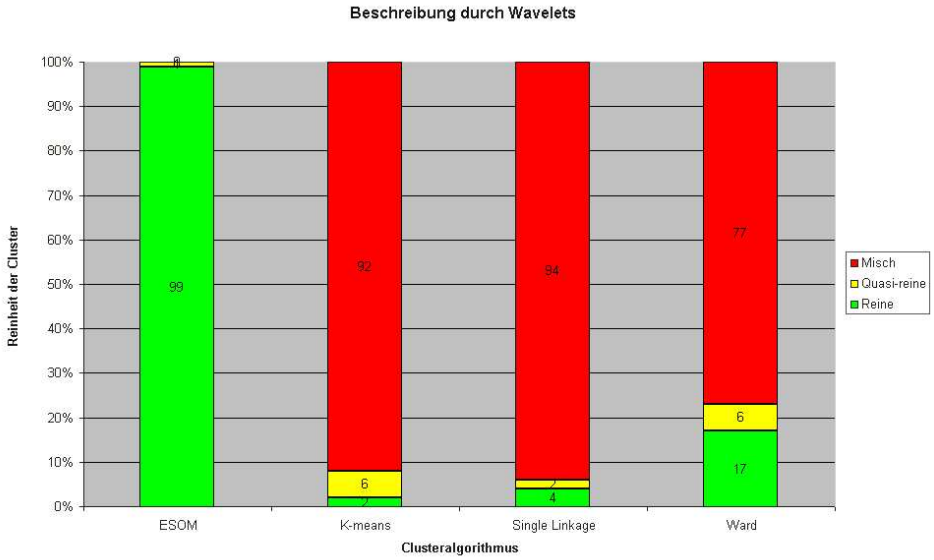
Beschreibung der Patches durch Aufteilung in Sektoren				
Algorithmus	**Reine Cluster**	**Quasi-reine Cluster**	**Misch-Cluster**	**Vollständigkeit**
ESOM-Clusterung	70%	15%	15%	ja
K-means	7%	8%	85%	nein
Single Linkage	16%	2%	82%	nein
Ward	22%	4%	74%	nein
Beschreibung der Patches durch Wavelets				
Algorithmus	**Reine Cluster**	**Quasi-reine Cluster**	**Misch-Cluster**	**Vollständigkeit**
ESOM-Clusterung	99%	1%	0%	ja
K-means	2%	6%	92%	nein
Single Linkage	4%	2%	94%	nein
Ward	17%	6%	77%	nein

Die Abbildung 6.30 zeigt die Werte der Tabelle in einem Diagramm. Der Anteil der reinen Cluster ist grün gefärbt, der der quasi-reinen Cluster gelb und der der Mischcluster ist rot gefärbt.

Für beide Möglichkeiten der Beschreibung der Patchoberfläche liefert die ESOM-



(a) Beschreibung durch Aufteilung in Sektoren



(b) Beschreibung durch Wavelets

Abbildung 6.30.: Ergebnisse der Clusterung des Datensatzes *DS V1* aus 109 Bindetaschen mit verschiedenen Clusteralgorithmen für beide Beschreibungen der geometrischen Gestalt der 2290 Patches.

Clusterung deutlich bessere Ergebnisse als die Standard-Clusteralgorithmen. *K-means* und *Ward* erzeugen zwar viele Cluster ähnlicher Größe, diese enthalten aber meist Patches von Enzymen zu unterschiedlichen EC-Nummern. Die Clusterung durch *Single Linkage* erzeugt einen sehr großen Cluster, welcher fast alle Patches enthält, und 100 bei der Beschreibung durch Sektoren bzw. 51 bei der Beschreibung durch Wavelets sehr kleine Cluster, die zum Teil nur ein Patch enthalten. Bei allen drei Algorithmen wird nicht für jede EC-Nummer mindestens ein reiner Cluster gefunden.

Die ESOM-Clusterung hingegen lässt solche Datenpunkte unklassifiziert, die große Distanzen zu ihren Nachbarn haben und in Bereichen geringer Dichte liegen. Diese Patches haben keine Gemeinsamkeiten zu Patches anderer Bindetaschen. Häufig sind dies Patches vom Rand der Bindetasche oder Bereiche zwischen zwei Subtaschen und sind somit für die Detektion von ähnlichen Substrukturen in den Taschen nicht relevant. Sie müssen für die Clusterung nicht berücksichtigt werden.

Diskussion der Ergebnisse des Vergleichs mit anderen Clusteralgorithmen

Die schlechten Ergebnisse der Standard-Clusterverfahren sind durch den großen Anteil an Ausreißern in dem Datensatz erklärt. Im Gegensatz zur ESOM-Clusterung müssen bei diesen Verfahren alle Datenpunkte einem Cluster zugewiesen werden, es gibt keine Möglichkeit, Ausreißer unklassifiziert zu lassen. Der Target-Datensatz [Ultsch, 2005] ist ein Beispiel für einen zweidimensionalen Datensatz mit Ausreißern. Er enthält zwei große Cluster und vier sehr kleine Ausreißer-Cluster. Die Abbildung 6.31 zeigt den Target-Datensatz. Die verschiedenen Cluster sind unterschiedlich eingefärbt.

Die Ergebnisse für die Algorithmen *K-means*, *Single Linkage* und *Ward* sind in Abbildung 6.32 dargestellt. Sowohl *K-means* als auch *Ward* ordnen die Ausreißer den großen Clustern zu. Nur *Single Linkage* klassifiziert die Ausreißer korrekt in eigenen Clustern.

Mit der ESOM-Clusterung bleiben solche Ausreißer unklassifiziert. Somit werden *Bestmatches*, die zu einer anderen EC-Nummer gehören und auf den Bergen rund um ein Cluster liegen, also einen großen Abstand zu den *Bestmatches* in dem entsprechenden Cluster haben, nicht berücksichtigt. Diese Tatsache verdeutlicht die so schlechten Ergebnisse der anderen Clusteralgorithmen im Vergleich zu den sehr guten Ergebnissen der ESOM-Klassifizierung.

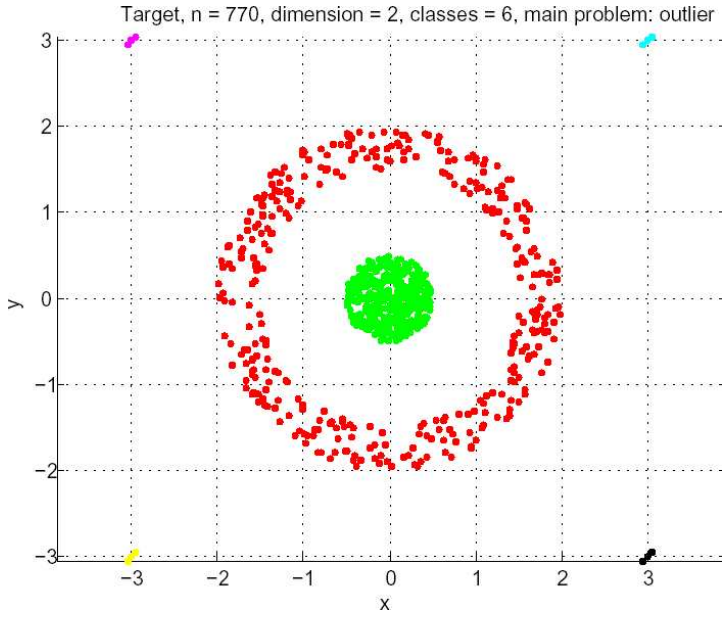


Abbildung 6.31.: Target-Datensatz. Der Datensatz enthält zwei große Cluster und vier kleine Ausreißer-Cluster. Die verschiedenen Cluster sind unterschiedlich eingefärbt.

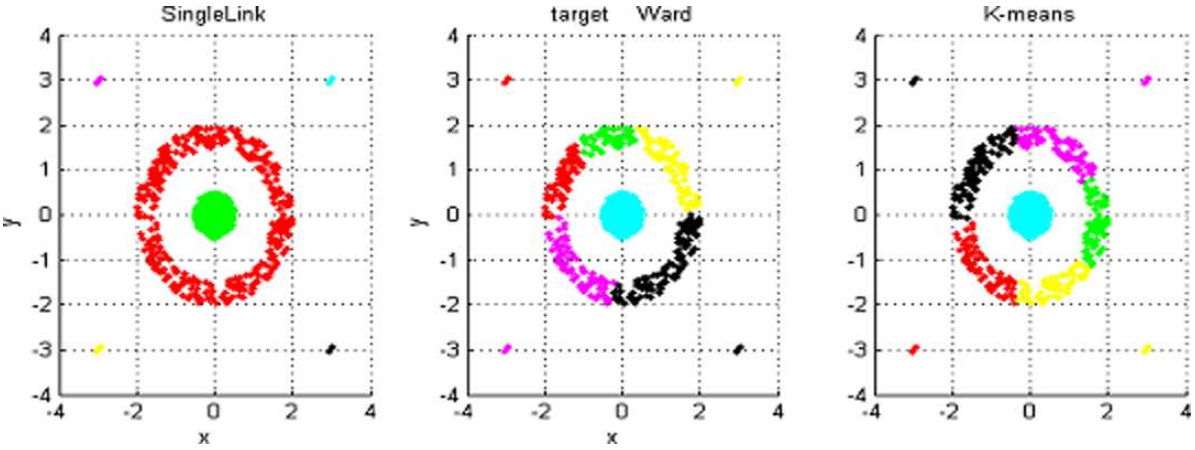


Abbildung 6.32.: Ergebnisse der Clusterung des Target-Datensatzes mit *K-means*, *Single Linkage* und *Ward*.

6.4. Zusammenfassung der Ergebnisse der Klassifizierung von Oberflächenpatches

Es wurden zwei verschiedene Verfahren (*Sector-* und *WaveGeoMap*) zum Vergleich der Oberflächenpatches von Proteinbindetaschen vorgestellt. Die eine Methode zerlegt die Patches in Sektoren und benutzt Mittelwert und Varianz der z-Koordinaten pro Sektor als Deskriptoren. Bei der anderen Methode wird die Oberfläche durch einen Satz von Approximations- und Waveletkoeffizienten beschrieben. Beide Verfahren benutzen die prozentuale Verteilung der physikochemischen Eigenschaften in vier Sektoren der Patches als zusätzliche Deskriptoren.

Die Beschreibung der Patches durch Waveletkoeffizienten liefert wesentlich bessere Clusterergebnisse als die Beschreibung durch einfache Aufteilung in Sektoren. Die verschiedenen EC-Nummern werden bei der Anwendung des Algorithmus auf den Testdatensatz mit 109 Bindetaschen wesentlich besser getrennt.

Die Beschreibung der Patches ist auf verschiedene Datensätze angewendet worden. Bei allen diesen Datensätzen liefert die ESOM-Clusterung sehr gute Ergebnisse. Für den Datensatz *DS T2*, der aus drei Subfamilien von Endopeptidasen besteht, sind beispielhaft die zu den Patches in den Clustern gehörigen Bereiche der Bindetaschen näher untersucht worden. In den Clustern mit *Bestmatches* zu verschiedenen Enzymen einer Subfamilie finden sich die katalytischen Residuen der Bindetaschen, die innerhalb einer Subfamilie weitestgehend konserviert sind. In den reinen Clustern finden sich solche Bereiche wie z.B. der 60er-Loop von Thrombin, die die verschiedenen Enzyme einer Subklasse voneinander diskriminieren.

Ein Vergleich mit den Clusteralgorithmen *K-means*, *Single Linkage* und *Ward* hat gezeigt, dass die ESOM-Clusterung wesentlich bessere Ergebnisse liefert als die drei Standard-Clusterverfahren. Die Tabelle 6.3 stellt die Ergebnisse der verschiedenen Verfahren angewendet auf die unterschiedlichen Datensätze zusammenfassend dar.

Tabelle 6.3.: Tabelle mit den Ergebnissen der Clusterung von Oberflächenpatches für die verschiedenen Datensätze und Beschreibungen der Oberfläche. Als Patchgröße ist der für die jeweilige Methode, die Patches zu beschreiben, optimale Radius verwendet worden.

Datensatz	**Beschreibung der Patches**	**Ergebnisse**
DS V1	nur physikochemische Deskriptoren	einige klar separierte Cluster mit Patches von Enzymen zu nur einer EC-Nummer, große Bereiche niedriger U-Höhen mit Patches zu Enzymen unterschiedlicher EC-Nummern
DS V1	*SectorGeoMap*	klar separierte Cluster mit Patches von Enzymen zu nur einer EC-Nummer, aber auch noch Bereiche mit Patches zu Enzymen unterschiedlicher EC-Nummern mit niedrigen U-Höhen
DS V1	*WaveGeoMap*	klar separierte Cluster mit Patches von Enzymen zu nur einer EC-Nummer, Bereiche mit Patches zu Enzymen unterschiedlicher EC-Nummern haben große U-Höhen und werden nicht berücksichtigt, sehr gute Auftrennung der Patches nach der zugehörigen EC-Nummer
DS V2	*WaveGeoMap*	siehe DS V1
Fortsetzung nächste Seite		

Fortsetzung		
Datensatz	**Beschreibung der Patches**	**Ergebnisse**
DS T1	*WaveGeoMap*	auch hier klar separierte Cluster mit Patches von Enzymen zu nur einer EC-Nummer, obwohl andere EC-Nummern in dem Datensatz als in V1 und V2 enthalten sind, sehr gute Auftrennung der Patches nach der EC-Nummer der zugehörigen Enzyme
DS T2	*WaveGeoMap*	sehr gute Clusterung der Patches nach der Funktion der zugehörigen Enzyme, Patches von ähnlichen Enzymen wie Trypsin und Thrombin landen teilweise im gleichen Cluster, für jede EC-Nummer existieren aber auch reine Cluster, die neben den Gemeinsamkeiten auch die Unterschiede der verschiedenen Enzyme verdeutlichen, Cluster entsprechen funktional bedeutsamen Bereichen der Bindetaschen
DS P1	*WaveGeoMap*	Patches ähnlicher Bindetaschen werden in die entsprechenden Cluster projiziert, solche ohne Ähnlichkeiten landen in Bereichen großer U-Höhen, da keine eigenen Cluster auf der Karte existieren, sehr gute Clusterung nach bekannten Ähnlichkeiten und Unähnlichkeiten

7. Diskussion

Das Ziel dieser Arbeit war es, eine Methode zu entwickeln, mit deren Hilfe Proteinbindetaschen effizient nach ähnlichen Substrukturen durchsucht werden können. Aufgrund der ständig wachsenden Zahl an Röntgenkristallstrukturen in der PDB (Protein Data Bank) ist es notwendig, dass der Algorithmus zum Vergleich der Bindetaschen einen sehr großen Datenbestand an Strukturen miteinander vergleichen kann. Der Vergleich der Strukturen soll eine Funktionszuweisung für neu gelöste Proteinstrukturen ermöglichen, da die Anzahl an Proteinen, für die keine Funktion bekannt ist, immer mehr zunimmt. Zudem sollte die Methode skalierbar sein, um einen effizienten Vergleich bei sehr großen Datenmengen und einen detaillierten Vergleich bei kleineren Datensätzen zu ermöglichen.

In der Literatur finden sich eine ganze Reihe von Verfahren, mit denen man die 3D-Struktur von Proteinen bzw. ihren Bindetaschen miteinander vergleichen kann. Bei vielen dieser Verfahren sind sehr rechenintensive eins-zu-eins-Überlagerungen notwendig. Dabei werden die Bindetaschen oder gesamten Proteinstrukturen häufig stark abstrahiert, z.B. durch ihre C_α-Atome, um auch für große Datensätze noch einen Vergleich möglich zu machen. Viele Verfahren verwenden nur geometrische Eigenschaften und vernachlässigen die für die Bindung des Liganden wichtigen physikochemischen Charakteristika. Für die molekulare Erkennung von Rezeptor und Ligand sind aber sowohl die räumliche Struktur als auch die physikochemischen Eigenschaften der Bindetasche wichtig. Beide müssen für einen Vergleich der Strukturen berücksichtigt werden.

Die in der Datenbank CAVBASE abgelegten Pseudozentren sind eine Approximation der räumlichen Anordnung der physikochemischen Eigenschaften einer Bindetasche. Vorversuche, in denen lokale Regionen bestehend aus vier Pseudozentren für den Vergleich verwendet wurden, haben gezeigt, dass die Pseudozentren eine zu grobe Beschreibung der Taschen darstellen (siehe Anhang C). Es ist nicht möglich, mit diesem Ansatz sicher ähnliche, funktionell bedeutsame Teilbereiche der Bindetaschen zu detektieren. Durch die geringe Komplexität der lokalen Region treten viele zufällige Gemeinsamkeiten zwischen den Bindetaschen auf, die sich auf eine einzelne lokale Region beschränken. Um

diese zufälligen Ähnlichkeiten, die keine Bedeutung hinsichtlich der funktionellen Bereiche der Bindetaschen haben, zu vermeiden, ist die Verwendung einer detaillierteren Beschreibung der Bindetasche notwendig.

In dieser Arbeit werden Substrukturen in Form von Patches der Lösungsmittel-zugänglichen Oberfläche von Proteinbindetaschen verwendet, da die Oberfläche eine sehr genaue Approximation der räumlichen Gestalt der Bindetaschen darstellt. Diese Oberflächenpatches werden sowohl durch geometrische als auch durch physikochemische Deskriptoren beschrieben. Die Deskriptoren müssen für jede neue Bindetasche nur einmal berechnet und gespeichert werden. Eine Überlagerung zum Vergleich der Bindetaschen muss hingegen jedes Mal neu berechnet werden.

Die Zerlegung der Bindetaschen in Oberflächenpatches erfolgt anhand der in der Datenbank abgelegten Pseudozentren. Der Mittelpunkt eines jeden Patches ist der Oberflächenpunkt mit dem geringsten euklidschen Abstand zu diesem Pseudozentrum. Dadurch wird eine systematische Zerlegung in Patches garantiert. Durch die Überlappung der Patches aufgrund ihrer Größe werden Fehler durch Ungenauigkeiten der Röntgenkristallstrukturen und damit verbundene geringe Variabilität bei der räumlichen Lage der Pseudozentren abgefangen. Große Flexibilität in den Taschen und die dadurch bedingten verschiedenen Zustände der räumlichen Lage von Teilbereichen der Bindetasche werden detektiert und führen zu unterschiedlichen Patches und damit auch zu unterschiedlichen Clustern entsprechend der verschiedenen Konformationen.

Die räumliche Gestalt der Oberflächenpatches wird durch einen Satz von Waveletkoeffizienten beschrieben (*WaveGeoMap*). Es gibt eine Reihe von Argumenten gegen die Verwendung von Wavelets. Zum einen ist die Berechnung der Wavelet-Transformation relativ rechenaufwändig. Eine einfache Beschreibung durch Mittelwert und Varianz der Höhen in vier Sektoren der Patches (*SectorGeoMap*) hat jedoch gezeigt, dass damit kleine, aber funktionell bedeutsame Unterschiede der Patchoberfläche nicht genügend diskriminiert werden können. Die aufwändige Berechnung der Koeffizienten muss für jede Bindetasche nur einmal durchgeführt werden, die Koeffizienten können mit den Patches zusammen in einer Datenbank abgelegt werden und dann bei Bedarf abgerufen werden.

Im Gegensatz zur Fourier-Transformation verlangt die Wavelet-Transformation keine Periodizität der Daten. Kugelflächenfunktionen liefern nur globale Koeffizienten, wohingegen die Waveletkoeffizienten lokal bestimmt werden und mit einer Ortsinformation versehen sind. So können bei nur teilweise ähnlichen Koeffizienten die entsprechenden Bereiche der Patches durch diese Ortsinformation bestimmt werden. Die Wavelet-

Transformation erlaubt verschiedene Resolutionsstufen, wodurch die Anzahl der Koeffizienten und damit die Genauigkeit der Beschreibung der Oberfläche und die verbundene Rechenzeit an die Anwendung angepasst werden kann.

Zum anderen ist die Wavelet-Transformation an sich nicht rotations-invariant. Für eine rotations-invariante Beschreibung der Patchoberfläche durch Waveletkoeffizienten muss vor der Berechnung der Deskriptoren eine Transformation in ein kanonisches Koordinatensystem und eine Ausrichtung der Patches erfolgen.

Bei der Koordinatentransformation kann durch die Form der Tasche der Schwerpunkt der Patches so ungünstig liegen, dass bildlich gesprochen die Öffnung der Patchoberfläche nicht parallel zur xy-Ebene liegt und so die Entfaltung dieser Patches nicht fehlerfrei ist. Da aber Patches, deren Randpunkte nach der Entfaltung keinen nahezu vollständigen Kreis beschreiben, nicht betrachtet werden, werden diese falsch rotierten Patches automatisch herausgefiltert.

Unterscheidet sich die Patchoberfläche in den vier Sektoren kaum, so kann es passieren, dass zwei ähnliche Patches durch Ungenauigkeiten der zugrundeliegenden Proteinstrukturen unterschiedlich ausgerichtet werden. Bei den verwendeten Datensätzen *DS V1* und *DS V2* sind es nur ca. 1% der Patches, bei denen der Unterschied zwischen den beiden Sektoren mit den kleinsten Mittelwerten geringer als 10% der Gesamthöhe des Patches ist. Es sind also nur sehr wenige Patches, die falsch ausgerichtet werden könnten. Durch die Redundanz der Patches aufgrund der Überlappungen wird dieser Fehler aber unwahrscheinlich, eines der verschiedenen Patchpaare wird gleich orientiert werden. Die endgültige Verifikation dieser Aussage anhand aller Bindetaschen der Datenbank CAVBASE steht aber noch aus.

Durch mögliche falsch ausgerichtete Patches sowie die Ungenauigkeiten in den verwendeten Röntgenkristallstrukturen der Proteine entstehen viele Ausreißer in den Daten, die keine Ähnlichkeiten zu anderen Patches des Datensatzes haben. Diese Ausreißer entsprechen den vielen Datenpunkten in Bereichen der U-Matrix mit großen U-Höhen. Durch die Möglichkeit des U*C-Clusteralgorithmus, diese Datenpunkte unklassifiziert zu lassen, ist die Ausrichtung der Patches sowie die bestehende Datengrundlage allerdings nicht kritisch für die Ergebnisse der Clusterung.

Als physikochemische Deskriptoren werden die prozentualen Anteile der fünf für die molekulare Erkennung wichtigen physikochemischen Eigenschaften in vier gleichgroßen Sektoren der Patches verwendet. Es sind deshalb vier Sektoren gewählt worden, weil sie einen guten Kompromiss zwischen Anzahl der Deskriptoren und damit Dimension der Deskriptionsvektoren und Genauigkeit der Beschreibung darstellen. Vorversuche haben

gezeigt, dass die Verteilung der Eigenschaften auf dem gesamten Patch nicht ausreichend für eine korrekte Diskriminierung der Patches ist, wohingegen eine größere Anzahl an Sektoren keine wesentliche Verbesserung der Ergebnisse bringt. Die Verwendung von prozentualen Anteilen macht die Beschreibung der physikochemischen Eigenschaften robust gegenüber Ungenauigkeiten der Röntgenkristallstrukturen durch Messfehler und geringen Variabilitäten in den Bindetaschen.

Um die einzelnen Patches effizient auf Ähnlichkeiten zu untersuchen, werden die zugehörigen Deskriptionsvektoren geclustert. Die Clusterung der Oberflächenpatches wurde mit der ESOM-Clusterung durchgeführt. Standard-Clusteralgorithmen wie *K-means*, *Single Linkage* und *Ward* sind nicht in der Lage, die Oberflächenpatches korrekt zu clustern. Dies liegt daran, dass der Datensatz keine klar voneinander abgegrenzten Cluster enthält. Zwischen den einzelnen Clustern von ähnlichen Patches liegen sehr viele hochdimensionale Ausreißer, die zu keiner dieser Cluster gehören. Der Clusteralgorithmus sollte diese Daten unklassifiziert lassen. *K-means*, *Single Linkage* und *Ward* ordnen jedoch jeden Datenpunkt einem Cluster zu. Die ESOM in Kombination mit dem automatischen Cluster-Algorithmus U*C erfüllt diese Forderung und kann Daten unklassifiziert lassen, wenn sie die Kriterien für eine Clusterzugehörigkeit nicht erfüllen.

Der Lernalgorithmus der SOM hat zwar einen Rechenaufwand von $O(n)$, die Rechenzeit ist aber zusätzlich zur Anzahl der Daten n auch von der Anzahl der Neuronen p abhängig. Aber der U*C-Cluster-Algorithmus ist im Gegensatz zu den Standard-Verfahren in der Lage, die Oberflächenpatches korrekt zu clustern. Mit der U-Matrix kann diese Clusterung auch visuell überprüft werden. Zudem ergibt sich die Anzahl der Cluster aus der Dichte- und Distanzstruktur der Daten während des Verfahrens. Es ist nicht notwendig, eine feste Anzahl von Clustern vorzugeben. Das ist besonders dann von Vorteil, wenn man einen neuen Datensatz betrachtet, bei dem man nicht weiß, welche Ähnlichkeiten und Unterschiede die verschiedenen Bindetaschen aufweisen, oder bei dem man die Größe der Taschen nicht kennt und damit nicht weiß, in wieviele Patches die Taschen zerlegt werden.

Durch die EC-Klassifikation der Enzyme existiert für die verwendeten Datensätze eine Vorklassifizierung. Die Güte der Clusterung für die verwendeten Datensätze *DS V1*, *DS V2* und *DS T1* ist anhand der Reinheit der Cluster bemessen worden. Diese Datensätze sind so ausgewählt worden, dass bei einer korrekten Clusterung nur Cluster mit Patches zu derselben EC-Nummer erwartet werden. Standardmaße zur Bewertung der Qualität der Clusterung wie Spezifität und Sensitivität [Bergs, 1981] können nicht verwendet werden, da jede vorgegebene EC-Klasse durch die Zerlegung der Bindetaschen

in Patches auch in eine unterschiedliche Anzahl an Cluster zerfällt, die den Substrukturen in den Bindetaschen entsprechen. Der Anteil der reinen Cluster im Verhältnis zu den Clustern mit Patches unterschiedlicher EC-Nummern ist deshalb ein geeignetes Maß, um die Ergebnisse der Clusterung zu bewerten und die Methode an diesem Maß zu validieren. Die Zerlegung der Bindetaschen ist sinnvoll, da es eine Reihe von Enzymen mit unterschiedlichen EC-Nummern gibt, die aber ähnliche Substrukturen enthalten, z.B. durch die Bindung des gleichen Kofaktors.

Die ESOM selbst ist ein nicht-lineares Projektionsverfahren, welches die Nachbarschaftsbeziehungen der hochdimensionalen Daten weitestgehend erhält. Für andere Verfahren wie Multidimensionale Skalierung (MDS) oder Sammon's Mapping lässt sich zwar ein Fehlermaß für die Projektion berechnen. Diese Verfahren haben einen Aufwand von $O(n^2)$. Mit ihnen ist es jedoch nicht möglich, die Clusterstrukturen des Validierungsdatensatzes *DS V1* zu erkennen, wohingegen durch die Visualisierung der ESOM mit der U-Matrix diese Strukturen sichtbar werden. Die Abbildung 7.1 zeigt das Ergebnis der MDS der Patches des Datensatzes *DS V1* aus 109 Bindetaschen auf zwei Dimensionen. Der Stresswert für diese MDS ist 0,44. Das bedeutet, dass die Abbildung die Nachbarschaftsbeziehungen schlecht erhält.

Im Gegensatz zu einem Vergleich von Bindetaschen durch Überlagerung der Strukturen bietet die ESOM die Möglichkeit, eine Karte für eine gesamte Datenbank von Bindetaschen zu trainieren und neu hinzukommende Datensätze auf die bestehende ESOM zu projizieren, ohne sie neu berechnen zu müssen. Damit muss das sehr zeitaufwändige Training nur einmal durchgeführt werden. Fallen die neuen Patches in ein Cluster, so sind sie ähnlich zu den in dem Cluster enthaltenen Patches.

Auf Grundlage der durchgeführten Literaturrecherche existiert noch kein Verfahren, mit dem Substrukturen von Proteinbindetaschen anhand ihrer physikochemischen und geometrischen Eigenschaften auf Ähnlichkeiten untersucht werden können und welches zusätzlich noch skalierbar ist, d.h. dass die Genauigkeit der Beschreibung an die Größe des Datensatzes anpassbar ist.

Mit der in dieser Arbeit vorgestellten Methode ist es gelungen, Oberflächenpatches von Proteinbindetaschen korrekt nach funktionellen Ähnlichkeiten zu clustern und damit die Bindetaschen effizient miteinander zu vergleichen. Patches von Enzymen gleicher Funktion landen in einem Cluster, während solche von Enzymen ohne über den Reaktionsmechanismus oder Ähnlichkeiten der Substrate begründbaren Gemeinsamkeiten in den Bindetaschen in unterschiedlichen Clustern landen. Über die verwendete Anzahl von Deskriptoren und den damit verbundenen unterschiedlichen Rechenaufwand ist die Me-

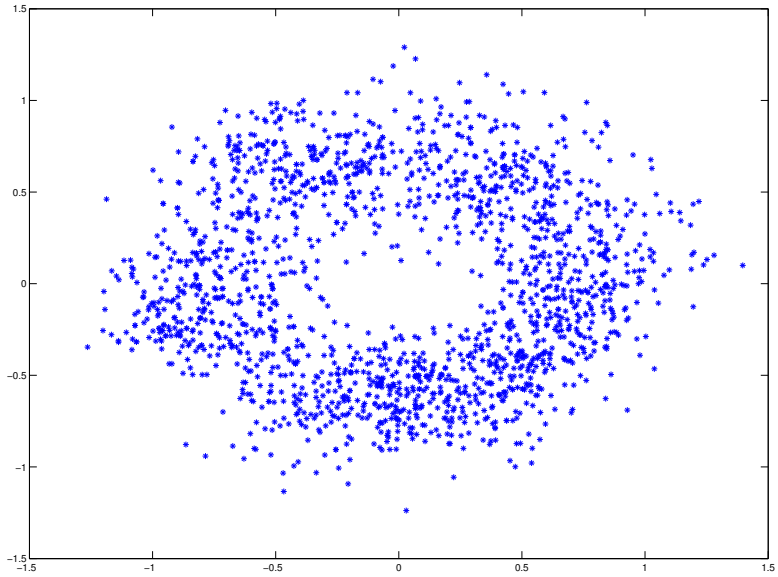


Abbildung 7.1.: Ergebnis der MDS des Datensatzes *DS V1* auf zwei Dimensionen. Die MDS hat einen Stresswert von 0,44, was eine schlechte Abbildung bedeutet.

thode an die gewünschte Anwendung anpassbar und somit wird erwartet, dass sie auch für sehr große Datenmengen wie z.B. den gesamten Inhalt der Datenbank CAVBASE [Schmitt et al., 2002] geeignet ist.

Durch die Clusterung von Oberflächenpatches ergibt sich für alle Bindetaschen eine Signatur gemäß der Clusterzugehörigkeit der verschiedenen Patches der Bindetaschen. Über diese Signatur sollte eine Funktionsvorhersage der Bindetaschen möglich sein. Am Beispiel dreier Unterfamilien der Endopeptidasen ist gezeigt worden, dass die Cluster auch biologisch bedeutsam sind. Sie enthalten Patches zu den für eine Unterfamilie charakteristischen sowie den für die einzelnen Enzyme diskriminierenden Bereichen. Mit der Signatur einer dieser Bindetaschen kann somit zum einen eine eindeutige Zugehörigkeit zu einer der drei Unterfamilien sowie auch eine eindeutige Zuweisung zu einer der verschiedenen EC-Nummern vorgenommen werden. Diese Signatur muss nun im nächsten Schritt für weitere in der CAVBASE abgelegten Strukturen ermittelt und überprüft werden.

Der letztendliche Beweis für die Funktionalität der Methode am gesamten Inhalt der Datenbank CAVBASE steht noch aus, da die Datenbank über 120000 Bindetaschen enthält und somit der Rechenaufwand riesig ist. Das Training einer ESOM für alle dort gespeicherten Bindetaschen würde mehrere Wochen dauern. Die Validität der Methode ist jedoch an fünf verschiedenen Datensätzen mit bekannten Ähnlichkeiten und Unterschieden in den Bindetaschen der einzelnen Strukturen überprüft worden. Die Beschreibung der Oberflächenpatches ist hinreichend diskriminierend für funktionell nicht verwandte Proteine. Die Gemeinsamkeiten von Strukturen einer Enzymfamilie sowie die spezifischen Unterschiede der einzelnen Enzyme in dieser Familie werden korrekt erkannt. Anhand des Datensatzes *DS T2* ist gezeigt worden, dass die gefundenen Cluster auch biologisch bedeutsam sind. Damit ist die Funktionalität der Methode nachgewiesen und kann in Zukunft auf den gesamten Datenbestand der CAVBASE angewendet werden.

8. Zusammenfassung

In dieser Arbeit wird eine neue Methode vorgestellt, um Gemeinsamkeiten in Proteinbindetaschen zu detektieren. Die lösungsmittelzugängliche Oberfläche der Taschen wird dazu systematisch in sogenannte Patches unterteilt. Für diese Patches werden sowohl geometrische als auch physikochemische Deskriptoren berechnet. Die einzelnen Merkmale werden aufbereitet, da sie aufgrund der euklidschen Abstandsberechnung beim ESOM-Training keine groben Ausreißer enthalten und die Verteilungen nicht schief sein dürfen sowie die gleiche Varianz besitzen müssen. Mit den aufbereiteten Deskriptoren wird dann eine ESOM antrainiert, womit die hochdimensionalen Daten topologiegetreu auf ein zweidimensionales Gitter von Neuronen projiziert werden. Um Strukturen in den Daten sehen zu können, wird das Ergebnis des ESOM-Trainings mit Hilfe der U-Matrix visualisiert. Die U-Matrix spiegelt die Distanzstrukturen im Datenraum wieder. Für eine automatische Clusterung der Daten wird der U*C-Clusteralgorithmus verwendet, der lokale Distanz- und Dichteinformationen für die Clusterung auf der ESOM verwendet. Ein Vergleich mit den Standard-Clusteralgorithmen *K-means*, *Single Linkage* und *Ward* hat bewiesen, dass die ESOM-Clusterung wesentlich bessere Ergebnisse liefert als diese gängigen Verfahren, da die Daten sehr viele Ausreißer enthalten.

Die Datengrundlage für Tests und Validierung ist die Datenbank CAVBASE, die über 120000 Proteinbindetaschen enthält. Für jede Bindetasche ist eine Menge von Pseudozentren, die eine räumliche Approximation der Bindungseigenschaften der die Tasche begrenzenden Aminosäuren darstellt, und eine Menge von Oberflächenpunkten, denen gemäß der zugehörigen Pseudozentren physikochemische Eigenschaften zugewiesen sind, in der Datenbank abgelegt. Ein kleiner Datensatz bestehend aus 109 Bindetaschen zu 13 verschiedenen Enzymfamilien ist für die Validierung der Methode ausgewählt worden. Dieser Datensatz ist erweitert worden auf 708 Bindetaschen, um die Ergebnisse mit dem kleinen Validierungsdatensatz zu überprüfen. Ein Testdatensatz, der die vorgeschlagene Methode bestätigt, besteht aus 1378 Bindetaschen der 13 Enzymfamilien, für die die meisten Taschen in der Datenbank CAVBASE abgelegt sind. Diese Enzymfamilien unterscheiden sich zum größten Teil von denen des Validierungsdatensatzes, so dass

sichergestellt ist, dass die Methode allgemeingültig und nicht nur auf den Validierungsdatensatz zugeschnitten ist. Ein weiterer Testdatensatz bestehend aus Bindetaschen aus drei Unterfamilien von Endopeptidasen ist verwendet worden, um zu zeigen, dass die Methode auch Gemeinsamkeiten zwischen ähnlichen Bindetaschen richtig detektiert.

Zwei verschiedene Ansätze zur Beschreibung der geometrischen Gestalt der Patches werden vorgeschlagen. Für den einen Ansatz werden die Patches in vier gleichgroße Sektoren unterteilt und die geometrische Gestalt durch Mittelwert und Varianz der Höhenwerte der Patchoberfläche in diesen Sektoren beschrieben (*SectorGeoMap*). Der zweite Ansatz verwendet die Koeffizienten von Oberflächenwavelets zur Beschreibung der Patchoberfläche (*WaveGeoMap*). Beide Ansätze zeigen für den Validierungsdatensatz vielversprechende Ergebnisse. Der Ansatz der Verwendung von Wavelets ist jedoch flexibler und liefert auch wesentlich bessere Clusterergebnisse. Die Anwendung der Beschreibung der Patchoberfläche mit Wavelets auf beide Testdatensätze beweist, dass die Methode in der Lage ist, unterschiedliche Enzymfamilien anhand der Substrukturen korrekt zu trennen und biologisch relevante Gemeinsamkeiten in den Bindetaschen von Enzymen derselben Familie zu finden. Daher ist die Methode geeignet, für unbekannte Proteine Gemeinsamkeiten zu bekannten Proteinen korrekt zu bestimmen und so die Funktion und/oder potentielle Liganden vorherzusagen.

A. Mathematische Symbole

Neuronale Netze

$bm(x)$	Bestmatch zum Eingabevektor x
d	Distanzfunktion
f_a	Aktivierungsfunktion
f_o	Ausgabefunktion
$i = (w_i, p_i)$	Neuron
r_p	Paretoradius
w	Gewichtsvektor
x	Eingabevektor

D	Eingaberaum
$I(S)$	Information einer Menge S
N	Gitter von Neuronen
$NN^M(i)$	Menge der äquidistanten Nachbarneuronen zu Neuron i
$PH(i)$	P-Höhe des Neurons i
$UH(i)$	U-Höhe des Neurons i
$UH^*(i)$	U*-Höhe des Neurons i
W	Menge der Gewichtsvektoren

Clusterung

C_k	Klasse k
$sig(i,j,k)$	Signifikanz eines Merkmals
a_i	i-tes Merkmal eines Datenpunktes

Wavelet-Transformation

$\phi_{j,v}(x)$	Skalierungsfunktion
$\psi_{u,s}(t)$	Wavelet
m	approximierter Punkt zwischen zwei Gitterpunkten
v_i	i-ter Gitterpunkt
s	Skalierungsfaktor der Wavelet-Koeffizienten
M	Menge von Zwischenpunkten m
$NN_j(*)$	Menge der direkten Nachbarn des Levels j
S	Menge von Oberflächenpunkten
V_j	Menge von Gitterpunkten des Levels j
$Wf(u,s)$	Wavelet-Transformation

Oberflächenpatches

$d_E(x, M)$	euklidsche Distanz zwischen Oberflächenpunkt x und Mittelpunkt
$d_F(x, M)$	Floyd-Distanz zwischen Oberflächenpunkt x und Mittelpunkt
dR	Distanz zwischen zwei Ringen auf dem Patch
dS	Distanz zwischen zwei Sektoren auf dem Patch
$d_{WM}(x, y)$	gewichtete Distanz zwischen zwei Vektoren x und y
r	Radius des Patches

KR	Menge der Randpunkte des Patches
M	Mittelpunkt des Patches
P	Menge der Oberflächenpunkte eines Patches
WM	Gewichtsmatrix bei der gewichteten Distanzberechnung

B. Algorithmen zur Klassifikation der Oberflächenpatches

Generierung der Patches und deren Ausrichtung

Algorithmus 2 Patchgenerierung und Transformation in kanonisches Koordinatensystem

Gegeben: Datensatz von Bindetaschen Cavs, Radius r, Pseudozentren PZ und Oberflächenpunkte SP für jede Bindetasche, nächste Nachbarn M_i der Pseudozentren als Mittelpunkte der Patches, Distanzen d_{F_i} von den Mittelpunkten M_i zu allen Oberflächenpunkten der Bindetasche

1: **for all** Cavs **do**
2: **for** $i = 1$ to $|PZ|$ **do**
3: $\text{Patch}_i = \{\text{SP} \mid d_{F_i}(\text{M}_i, \text{SP}) \leq r\}$
4: $\text{TransPatch}_i = \text{Patch}_i - \text{M}_i$
5: $\text{Schwerpunkt}_i = \text{mean}(\text{TransPatch}_i)$
6: $\text{normSchwerpunkt}_i = \frac{\text{Schwerpunkt}_i}{|\text{Schwerpunkt}_i|}$
7: $\text{RotAchse} = \text{normSchwerpunkt} \times [0, 0, 1]$
8: $\text{alpha} = \angle(\text{normSchwerpunkt}, [0, 0, 1])$
9: $\text{RotPatch}_i = \text{rotateAroundAxis}(\text{TransPatch}_i, \text{alpha}, \text{RotAchse})$
10: **end for**
11: **end for**

Algorithmus 3 Entfaltung der Patches

Gegeben: Koordinaten der Punkte PP = (P_x, P_y, P_z) eines ausgerichteten Patches Rot-Patch, Distanzen d_F(PP,M) der Patchpunkte vom Mittelpunkt

1: **for all** PP **do**
2: **if** PP $\neq 0$ **then**
3: $\mathrm{L} = \sqrt{{P_x}^2 + {P_y}^2}$
4: (newP$_x$,newP$_y$) = (P$_x$,P$_y$)/L*d_F(PP,M)
5: newP$_z$ = P$_z$
6: **else**
7: newPP = PP
8: **end if**
9: **end for**

Algorithmus 4 Bestimmung V_0

Gegeben: Punkte PP eines entfalteten Patches, Patchradius r, Anzahl der Punkte NrPoints in V_0

1: M = [0,0,0]
2: vektor = [r,0]
3: V0$_1$ = NearestNeighbor(vektor,[PP$_x$,PP$_y$])
4: **for** $i = 2$ to NrPoints **do**
5: vektor$_i$ = rotateAroundZAxis(vektor$_{i-1}$,$\frac{2\pi}{\mathrm{NrPoints}}$)
6: V0$_i$ = NearestNeighbor(vektor$_i$,[PP$_x$,PP$_y$])
7: **end for**

Algorithmus 5 Löschen von Patches mit unvollständigem Rand

Gegeben: V_0 eines Patches, Anzahl der Punkte NrPoints in V_0

1: **for** $i = 1$ to NrPoints **do**
2: diff$_i$ = $|[V0_{i,x}, V0_{i,y}] - [\mathrm{vektor}_{i,x}, \mathrm{vektor}_{i,y}]|$
3: **end for**
4: inddiff = find(diff< 1)
5: **if** length(inddiff)< 8 **then**
6: delete(Patch)
7: **end if**

Algorithmus 6 Bestimmen der optimalen Sektoreinteilung und Ausrichtung des Patches

Gegeben: Punkte PP eines entfalteten Patches, Distanzen d_F(PP,M) der Patchpunkte vom Mittelpunkt, Anzahl der Sektoren AnzSectors = [AnzSlices,0]

MaxAngle = $\frac{360°}{\text{AnzSectors}}$
AnzDrehungen = $\frac{\text{MaxAngle}}{10}$
for $j = 1$ to AnzDrehungen **do**
    Slices = segmentPatch2Slices(PP,AnzSlices)
    for $i = 1$ to AnzSlices **do**
        Mittelwert = mean($\text{PP}_z \in \text{Slice}_i$)
        $\text{Abstand}_k = d_E$(Mittelwert,$\text{PP}_z \in \text{Slice}_i$)
        $\text{Fehler}_i = \sum_k \text{Abstand}_k{}^2$
        $\text{Absolutfehler}_j = \sum_i \text{Fehler}_i$
    end for
    PP = rotateAroundZAchsis(PP,10°)
end for
Minind = find(j=min(Absolutfehler))
OptimalAngle = 10*Minind
PP = rotateAroundZAchsis(PP,OptimalAngle)
Slices = segmentPatch2Slices(PP,AnzSlices)
for $i = 1$ to AnzSlices **do**
    Mittelwert_i = mean($\text{PP}_z \in \text{Slice}_i$)
end for
ind = find(Mittelwert=min(Mittelwert))
CanonicalRotationAngle = ((ind-1)*MaxAngle)
PP = rotateAroundZAchsis(PP,CanonicalRotationAngle)

Bestimmen der physikochemischen Eigenschaften

Algorithmus 7 Berechnen der physikochemischen Deskriptoren

Gegeben: Punkte PP eines entfalteten Patches, physikochemische Eigenschaften PCP $= \{DO, AK, DA, AR, AL\}$ zu den Oberflächenpunkten eines Patches, Anzahl der Sektoren AnzSectors = [AnzSlices,0]

```
Slices = segmentPatch2Slices(PP,AnzSlices)
for i = 1 to AnzSlices do
    SliceChemie = PCP(Slice_i)
    Donor = find(SliceChemie=DO)
    Akzeptor = find(SliceChemie=AK)
    DonorAkzeptor = find(SliceChemie=DA)
    Aromatic = find(SliceChemie=AR)
    Aliphatic = find(SliceChemie=AL)
    DonorProzent_i = 1/length(SliceChemie)*length(Donor)
    AkzeptorProzent_i = 1/length(SliceChemie)*length(Akzeptor)
    DonorAkzeptorProzent_i = 1/length(SliceChemie)*length(DonorAkzeptor)
    AromaticProzent_i = 1/length(SliceChemie)*length(Aromatic)
    AliphaticProzent_i = 1/length(SliceChemie)*length(Aliphatic)
end for
```

Shape Histograms

Algorithmus 8 Bestimmung von Mittelwert und Varianz der Sektoren

Gegeben: Punkte PP eines entfalteten Patches, Anzahl der Sektoren AnzSectors = [AnzSlices,AnzRings]

```
1: s = 1
2: Slices = segmentPatch2Slices(PP,AnzSlices)
3: for i = 1 to AnzSlices do
4:     Rings = segmentPatch2Rings(Slice_i)
5:     for j = 1 to AnzRings do
6:         Mittelwert_s = mean(Ring_{i,z})
7:         Varianz_s = var(Ring_{i,z})
8:         s = s+1
9:     end for
10: end for
```

Wavelet-Transformation der Patches

Algorithmus 9 Berechnung der Zwischenpunkte des Gitters V_k

Gegeben: Punkte PP eines entfalteten Patches, Gitter V_k von Oberflächenpunkten

1: triV = DelaunayTriangulation(V_x,V_y)
2: r = ♯ Zeilen von triV
3: d = ♯ Spalten von triV
4: s = 1
5: **for** $i = 1$ to r **do**
6: **for** $j = 1$ to $d - 1$ **do**
7: PredictedV$_s$ = $\frac{1}{2}$*V(triV$_{i,j}$)+V(triV$_{i,j+1}$)
8: L$_s$ = $\sqrt{\text{dist(V(triV}_{i,j}\text{),V(triV}_{i,j+1}\text{)}}$
9: Lvektor = V(triV$_{i,j}$)-V(triV$_{i,j+1}$)
10: **if** $j = 1$ **then**
11: l$_s$ = $\sqrt{\text{dist(PredictedV(s),V(triV}_{i,3}\text{)}}$
12: lvektor = PredictedV$_s$-V(triV$_{i,3}$)
13: **else if** $j = 2$ **then**
14: l$_s$ = $\sqrt{\text{dist(PredictedV}_s\text{,V(triV}_{i,1}}$
15: lvektor = PredictedV$_s$-V(triV$_{i,1}$)
16: **end if**
17: alpha$_s$ = $\frac{pi}{2}$*$\measuredangle$(Lvektor,lvektor)
18: s = s+1
19: **end for**
20: PredictedV$_s$ = 1/2*(V(triV$_{i,1}$)+V(triV$_{i,d}$))
21: L$_s$ = $\sqrt{\text{dist(V(triV}_{i,1}\text{),V}_{i,d}\text{))}}$
22: Lvektor = V(triV$_{i,1}$)-V(triV$_{i,d}$)
23: l$_s$ = $\sqrt{\text{dist(PredictedV(s),V(triV}_{i,2}\text{))}}$
24: lvektor = PredictedV$_s$-V(triV$_{i,2}$)
25: alpha$_s$ = $\frac{pi}{2}$*$\measuredangle$(Lvektor,lvektor)
26: s = s+1
27: [PredictedV,m,n] = unique(PredictedV) {[b,m,n]=unique(A) mit b=A(m) and A=b(n)}
28: **end for**

Algorithmus 10 Berechnung der Skalierungsfaktoren

Gegeben: L, l, alpha, m, n (siehe 9)

1: **for** $i = 1$ to length(L) **do**
2: $\text{scale}_i = \text{L}_i \left(\frac{3}{4} \frac{L_i^2}{l_i^2} (\cos(alpha_i) + \sin(alpha_i))^2 + 4 \right)^{\frac{-1}{2}}$
3: **end for**
4: **for** $i = 1$ to length(L) do **do**
5: findindex = find(n_i=n)
6: **if** length(findindex)=2 **then**
7: Durchschnitt = $\frac{\text{scale}(\text{findindex}_1)+\text{scale}(\text{findindex}_2)}{2}$
8: **else**
9: Durchschnitt = scale(findindex_1)
10: **end if**
11: S(findindex) = Durchschnitt
12: **end for**
13: Skalierungsfaktor = S_m

Algorithmus 11 Berechnung der skalierten Waveletkoeffizienten

Gegeben: Punkte PP eines entfalteten Patches, dazugehörige Zwischenpunkte PredictedV und Skalierungsfaktoren

1: **for** $i = 1$ to length(PredictedV) **do**
2: $\text{CorrespondingRealVertex}_i$ = NearestNeighbor(PredictedV_i,PP)
3: WavKoeff_i = $\text{CorrespondingRealVertex}_{i,z}$-$\text{PredictedV}_{i,z}$
4: **end for**
5: **for** $i = 1$ to length(Skalierungsfaktor) **do**
6: ScaledWavKoeff_i = $\text{Skalierungsfaktor}_i$*$\text{WavKoeff}_i$
7: **end for**

Algorithmus 12 Berechnung des neuen Gitters V_{k+1}

Gegeben: Gitter V_k von Oberflächenpunkten, berechnete Zwischenpunkte PredictedV, dazugehörige reale Oberflächenpunkte CorrespondingRealVertex und Waveletkoeffizienten WavKoeff zu diesem Gitter

1: **for** $i = 1$ tolength V_k **do**
2: NNWavK = direkt benachbarte Wavkoeffs von $\text{V}_{k,i}$
3: $\text{newV}_{k,i} = \text{V}_{k,i} + \frac{3}{4} \sum \text{NNWavK}$
4: **end for**
5: $\text{V}_{k+1} = \text{newV}_k \cup \text{CorrespondingRealVertex}$

C. Klassifizierung minimaler Substrukturen in Protein-Bindetaschen

Algorithmus zur Klassifizierung minimaler Substrukturen

Um Teilbereiche einer Bindetasche vergleichen zu können, ist es notwendig, diese in Substrukturen zu zerlegen. Diese Substrukturen sollen sowohl über geometrische als auch physikochemische Charakteristika verfügen. Im ersten Schritt wird der Ansatz einer „minimalen“ Substruktur verfolgt. Sie zeichnet sich durch einfache Generierung, geringe Größe und einfache Beschreibbarkeit aus. Grundlage für solch „minimale“ Substrukturen stellt die Menge der Pseudozentren einer Bindetasche dar, die eine einfache räumliche Darstellung der physikochemischen Eigenschaften einer Bindetasche sind. Der Nachteil einer solchen „Minimalstrategie“ ist die starke Vereinfachung der Eigenschaften dieses Teilbereichs einer Bindetasche und die damit verbundenen ungenaueren Ergebnisse bei der Klassifizierung der Bindetaschen.

Die lokale Struktur aus vier Pseudozentren

Als eine „lokale Struktur“ definieren wir zunächst vier Pseudozentren einer Bindetasche. Sie wird aus einem Hauptzentrum und seinen drei nächsten Nachbarn als Nebenzentren gebildet [Siemon, 2001]. Die lokale Struktur ist deswegen so gewählt worden, weil sie die kleinstmögliche räumliche Einheit darstellt, die aus der Menge von Pseudozentren gebildet werden kann. Betrachtet man die Verbindungsstrecken zwischen den Pseudozentren, haben die lokalen Strukturen die Form einer Dreieckspyramide. Sie sind zum einen durch ihre geometrischen Eigenschaften, wie z.B. Höhe und Schiefe der Pyramide, Abstände zwischen den Pseudozentren, Grundfläche, Umfang etc., als auch durch die Komposition der physikochemischen Eigenschaften der vier Pseudozentren charak-

terisiert. Die Bindetaschen werden systematisch in lokale Strukturen zerlegt. Dazu wird jedes Pseudozentrum einer Bindetasche als Hauptzentrum gewählt und bildet mit seinen drei nächsten Nachbarn eine lokale Region. Jede Bindetasche wird also in n lokale Regionen mit $n = Zahl\ der\ Pseudozentren$ zerlegt.

Aufbereitung der Daten

Zur Validierung und für das Testen der Methode ist der in Kapitel 4.1 beschriebene Datensatz von 109 Enzymbindetaschen verwendet worden, da eine erste Studie dieses Ansatzes mit einem Datensatz bestehend aus acht Bindetaschen erfolgreich war [Siemon, 2001]. Die Bindetaschen sind in 8462 lokale Regionen zerlegt worden. Es sind sehr viele geometrische Deskriptoren für die lokalen Regionen berechnet worden. Diese waren zum großen Teil stark korreliert, deswegen sind die drei Eigenschaften

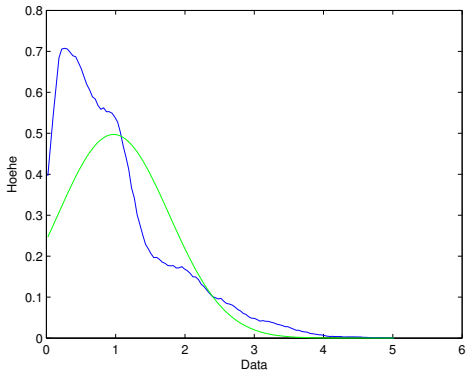
- Höhe:= Abstand des Hauptzentrums zu der durch die drei Nebenzentren definierten Grundfläche,
- Schiefe:= Abstand zwischen dem Schwerpunkt der Grundfläche und dem Fußpunkt der Höhe und
- Fläche:= Flächenmaßzahl der Grundfläche

ausgewählt worden. Sie beschreiben die geometrische Gestalt der Pyramide hinreichend. Die deskriptive Statistik des Datensatzes ist in Tabelle C.1 dargestellt.

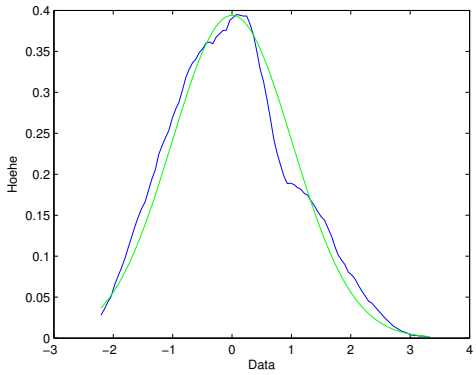
Tabelle C.1.: Deskriptive Statistik des Datensatzes von 8462 lokalen Regionen, generiert aus 72 Bindetaschen unterschiedlicher EC-Nummern

Merkmal	Minimum	Mittelwert	Median	Maximum	Std.-Abweichung
Höhe	0	0.9701	0.78018	5.0371	0.80237
Schiefe	0.000926	1.3858	1.318	5.7867	0.74331
Fläche	0.001044	2.9048	2.1138	22.294	2.6506

Die Verteilungen aller drei Merkmale sind nicht symmetrisch, sondern links-schief. Die am besten geeigneten Transformationen zur Normalisierung der Verteilung sind die Quadratwurzel für die Höhe und die Schiefe und die dritte Wurzel für die Fläche. Die Abbildungen C.1, C.2 und C.3 zeigen die *PDE-Plots* [Ultsch, 2003a] der Original-Merkmale (links) und die der transformierten Werte (rechts).

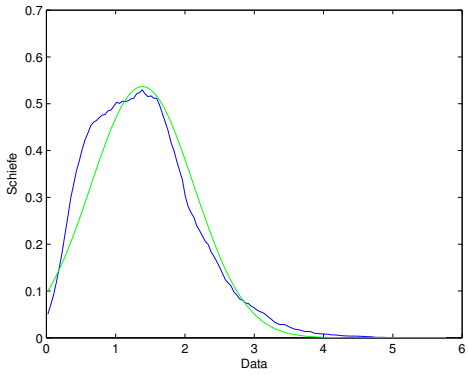


(a) Originalwerte für die Höhe der lokalen Regionen

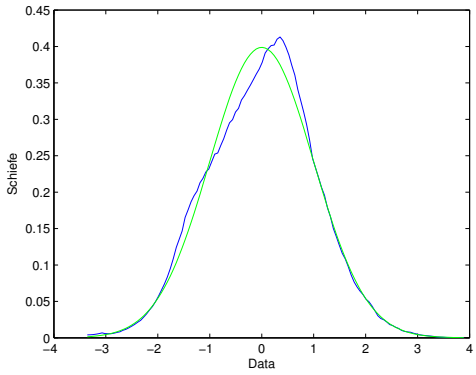


(b) Transformierte Werte für die Höhe der lokalen Regionen

Abbildung C.1.: *PDE-Plot* der Original- und transformierten Werte für die Höhe der lokalen Regionen

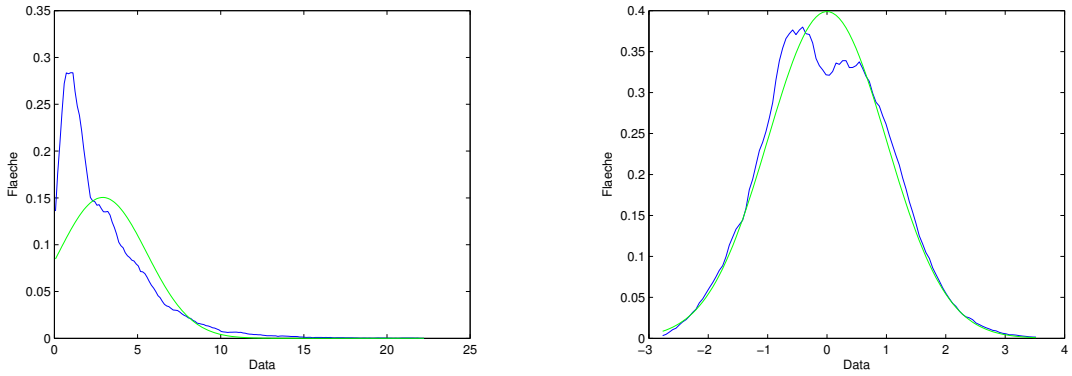


(a) Originalwerte für die Schiefe der lokalen Regionen



(b) Transformierte Werte für die Schiefe der lokalen Regionen

Abbildung C.2.: *PDE-Plot* der Original- und transformierten Werte für die Schiefe der lokalen Regionen



(a) Originalwerte für die Fläche der lokalen Regionen

(b) Transformierte Werte für die Fläche der lokalen Regionen

Abbildung C.3.: *PDE-Plot* der Original- und transformierten Werte für die Fläche der lokalen Regionen

Die Abbildung C.4 zeigt die drei Merkmale als dreidimensionale Abbildung. Die Abbildung zeigt, dass alleine auf Basis der Distanzen zwischen den Datenpunkten keine Gruppen von lokalen Regionen ausgemacht werden können. Es gibt aber Unterschiede in der Dichte zwischen den einzelnen Punkten.

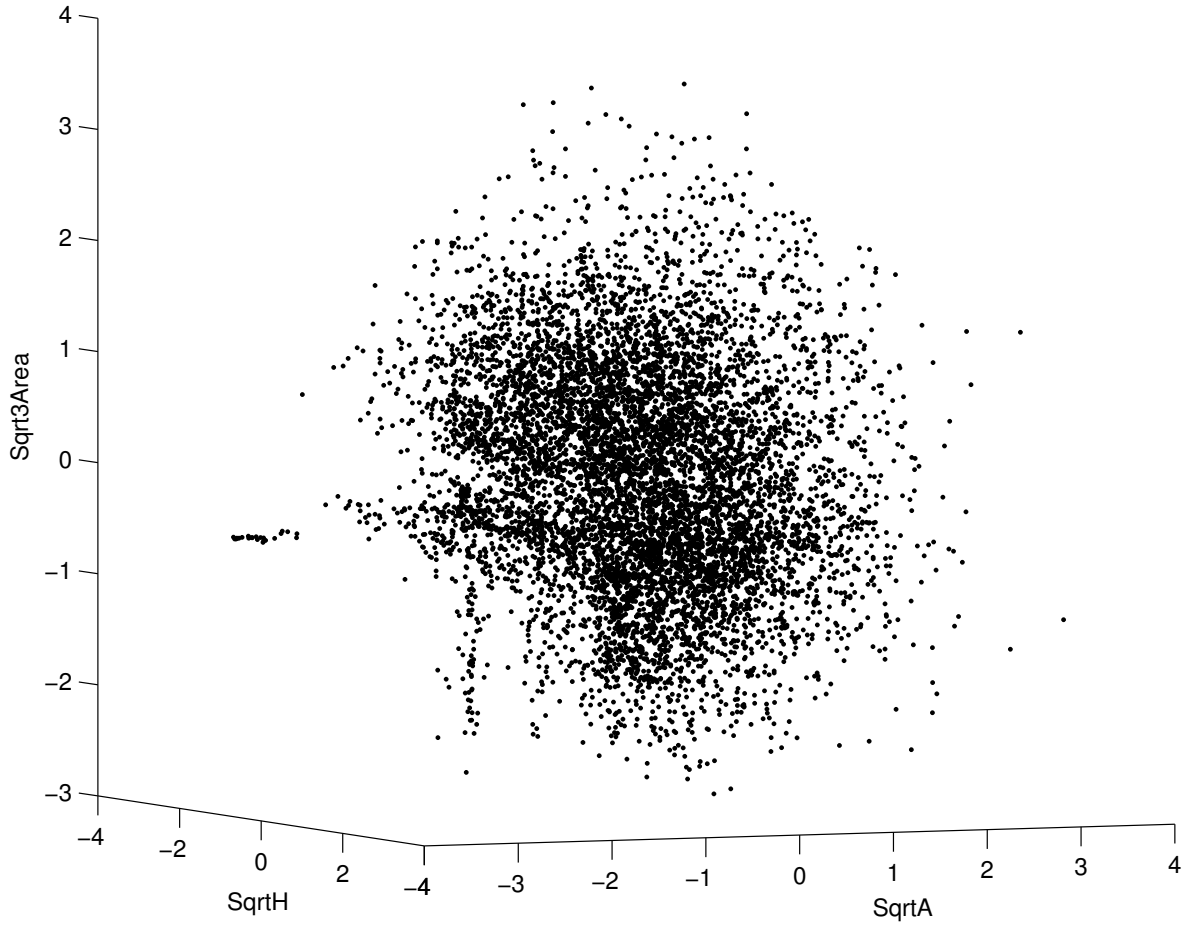


Abbildung C.4.: Der Plot zeigt die drei geometrischen Eigenschaften der lokalen Regionen nach Transformation. SqrtH ist die Wurzel der Höhe, SqrtA ist die Wurzel der Schiefe and Sqrt3Area ist die dritte Wurzel der Fläche.

ESOM-Clusterung der lokalen Regionen

Die Werte für die geometrischen Eigenschaften der lokalen Struktur sind aus $\mathbb{R}^+$, wohingegen die Werte für die physikochemischen Eigenschaften aus $\mathbb{Z} \cap [1, 5]$ sind. Deshalb wird auf ein gleichzeitiges ESOM-Training mit geometrischen und physikochemischen Eigenschaften der lokalen Regionen verzichtet. Die Clusterung der lokalen Regionen vollzieht sich somit in zwei Schritten. Für das ESOM-Training werden nur die geometrischen Deskriptoren verwendet. Die ermittelten Cluster werden im nächsten Schritt in Subcluster gemäß der physikochemischen Eigenschaften der lokalen Regionen unterteilt.

Es wurde eine toroide ESOM mit 100×164 Neuronen in 50 Lernzyklen mit den oben beschriebenen geometrischen Deskriptoren (siehe Kapitel C) antrainiert [Kupas and

Ultsch, 2005]. Die Clusterung der lokalen Regionen wird mit U- und P-Matrix vollzogen [Kupas et al., 2004a,b]. Alle diejenigen lokalen Regionen, die in zusammenhängenden Regionen mit großer Dichte auf der P-Matrix und mit geringen Distanzen zwischen den Datenpunkten auf der U-Matrix liegen, werden einem Cluster zugeordnet. Die Clustergrenzen stellen weniger dichte Bereiche der P-Matrix dar. Regionen mit sehr geringer Dichte und großen Distanzen zwischen den Datenpunkten werden keinem Cluster zugeordnet.

Die so generierten Cluster enthalten lokale Regionen mit ähnlichen geometrischen, aber allen möglichen, unterschiedlichen Kombinationen von physikochemischen Eigenschaften der vier Pseudozentren. Deswegen werden die Cluster weiter in Subcluster unterteilt. Lokale Regionen innerhalb eines Clusters mit gleichen physikochemischen Eigenschaften werden einem Subcluster zugeordnet. Solche mit unterschiedlichen physikochemischen Eigenschaften werden auch unterschiedlichen Subcluster zugeordnet. In den Subclustern sind die lokalen Regionen dann sowohl gemäß ihrer räumlichen Struktur als auch gemäß ihrer physikochemischen Eigenschaften geclustert.

Die ESOM-Clusterung hat verschiedene Vorteile:

- Es hat sich gezeigt, dass ESOM in der Lage ist, auch ineinander verwobene Clusterstrukturen korrekt zu entflechten [Ultsch, 2003c].
- Die Anzahl der Cluster ist nicht fest vorgegeben, sondern entwickelt sich automatisch während des ESOM-Trainings. Enthält ein Datensatz keine Cluster, so findet man auch keine Clusterstrukturen auf der Karte.
- Es müssen nicht alle Datenpunkte einem Cluster zugewiesen werden. Ausreißer in den Daten lassen sich leicht identifizieren und können von der Clusterung ausgeschlossen werden.

Validität der Clusterung

Die oben beschriebene Methode liefert eine Clusterung der Substrukturen der Proteinbindetaschen. Eine Clusterung heißt dann valide, wenn die lokalen Strukturen innerhalb desselben Clusters eine große Ähnlichkeit aufweisen, die aus verschiedenen Clustern sich jedoch in mindestens einem Merkmal deutlich unterscheiden. Um die Validität der Clusterung zu überprüfen, wird der sig*-Algorithmus verwendet [Ultsch, 1992]. Sig* generiert für jedes Cluster Regeln, die angeben, in welchen Intervallen die Werte der Deskriptoren für die Substrukturen innerhalb dieses Clusters liegen. Für die Überprüfung der

Validität wird die Schwelle s, die angibt, zu welchem Prozentsatz die Cluster durch die Regeln charakterisiert werden, auf 100% gesetzt, damit die Regeln für alle Deskriptoren berücksichtigt werden.

Seien $C_1, ..., C_k$ die zu betrachtenden Klassen, $v_1, ..., v_n$ die Merkmale der d Eingabedaten $x_1, ..., x_d \in D$, $W(v_1), ..., W(v_n)$ der Wertebereich dieser Merkmale. Seien $S_{1i}, ..., S_{ni}$ die Intervalle für die Werte der Merkmale $v_1, ..., v_n$ der Klasse C_i. Die Clusterung ist genau dann valide, wenn gilt:

1. $max(S_{ti}) - min(S_{ti}) \leq \frac{2}{k} * W(v_t) \; \forall \; t \in [1, n] \subset \mathbb{Z}, \forall \; i \in [1, k] \subset \mathbb{Z}$
2. $i, j \in [1, k] \subset \mathbb{Z}, i \neq j : \exists \; t \in [1, n] \subset \mathbb{Z} : \; S_{ti} \cap S_{tj} \leq \frac{1}{2k} * W(v_t)$

D.h. sind die Substrukturen innerhalb desselben Clusters sehr ähnlich, so sind die Intervalle für die Werte der Deskriptoren klein. Die Intervalle für mindestens ein Merkmal der Substrukturen zweier unterschiedlicher Cluster dürfen nur eine kleine Schnittmenge haben, damit sich die Substrukturen hinreichend unterscheiden.

Bewertung der gefundenen Übereinstimmungen

Die Ähnlichkeit zweier Substrukturen von zwei Bindetaschen muss nun noch bewertet werden, um Rückschlüsse auf die Ähnlichkeit der beiden Taschen zu ziehen. Dabei werden nur solche Substrukturen betrachtet, die innerhalb desselben Clusters liegen, da solche Substrukturen aus unterschiedlichen Clustern kaum Ähnlichkeiten aufweisen.

Soll die Ähnlichkeit der gesamten Bindetaschen anhand der gefundenen ähnlichen Substrukturen bewertet werden, so werden die beiden Substrukturen überlagert. Aus dieser Überlagerung ergibt sich eine Koordinatentransformation, die auf die beiden zugehörigen Bindetaschen angewandt wird. Dann werden alle Paare von Pseudozentren bzw. Oberflächenpunkten gezählt, die die gleiche physikochemische Eigenschaft besitzen und einen Abstand kleiner als 1 Å haben. Die Anzahl der Paare ist der *Score* für diese beiden Substrukturen. Die Überlagerung wird für alle ähnlichen Substrukturen dieser beiden Taschen ausgeführt. Der grösste Wert für den *Score* ist dann der absolute *Score* für diese beiden Taschen. Dieser absolute *Score* wird dann noch in Relation zum maximal möglichen *Score*, das ist die Menge aller Pseudozentren bzw. Oberflächenpunkte der kleineren der beiden Taschen, gesetzt. Dies wird durchgeführt, um die *Scores* der verschiedenen Paare von Bindetaschen miteinander vergleichen zu können. Die so erhaltene Prozentzahl macht dann den *Score* für dieses Paar von Bindetaschen aus. Die Erfahrung hat gezeigt, dass Gemeinsamkeiten in Bindetaschen, die kleiner als neun Pseudozentren

sind, in der Regel bedeutungslos sind. Aus diesem Grund werden alle die Überlagerungen mit einem *Score* kleiner als neun verworfen, es werden nur die Paare von Bindetaschen weiter betrachtet, die mindestens neun gemeinsame Pseudozentren aufweisen.

Ergebnisse der Klassifizierung minimaler Substrukturen

Clustering der lokalen Regionen

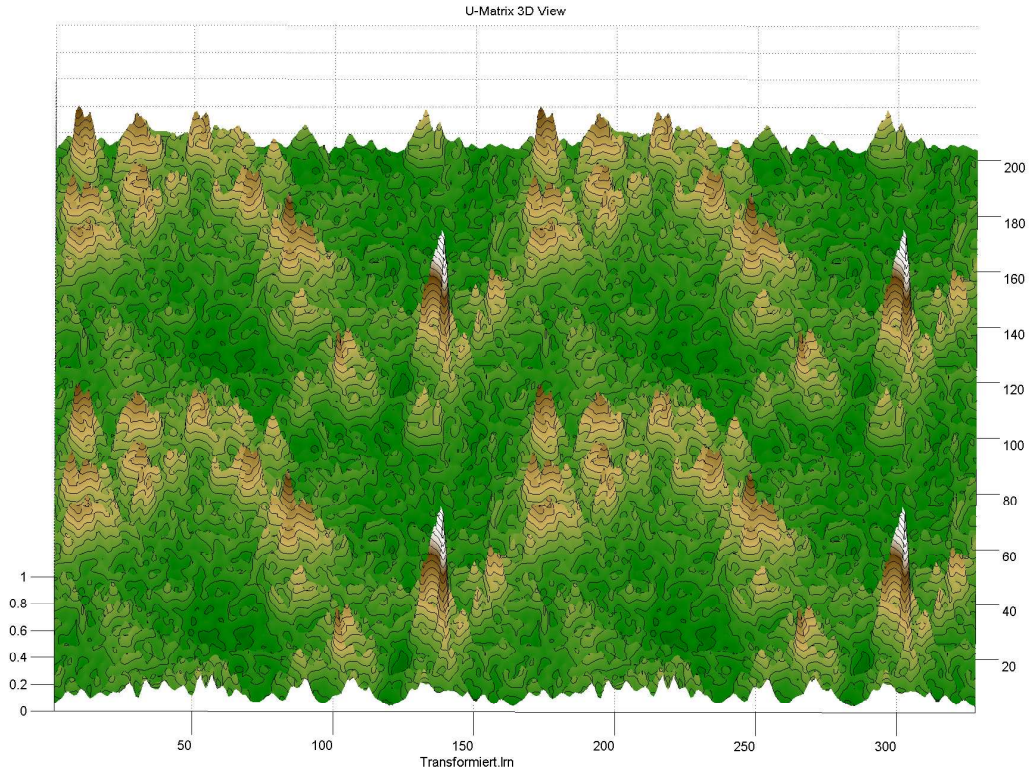


Abbildung C.5.: Die Abbildung zeigt die U-Matrix zu der mit den geometrischen Eigenschaften der lokalen Regionen antrainierten ESOM. Die Bestmatches der lokalen Regionen sind als weiße Punkte eingezeichnet.

Die Abbildung C.5 zeigt die zu dem Datensatz aus 109 Bindetaschen gehörige U-Matrix in geteilter Darstellung. Wie in Abbildung C.4 sind auch auf der U-Matrix keine Clustergrenzen sichtbar. Auf der zugehörigen P-Matrix sind jedoch zusammenhängende Bereiche mit hoher Dichte sichtbar, die durch weniger dichte Zonen voneinander abge-

grenzt sind. Die Abbildung C.6 zeigt diese P-Matrix, die Abbildung C.7 die zugehörige U*-Matrix.

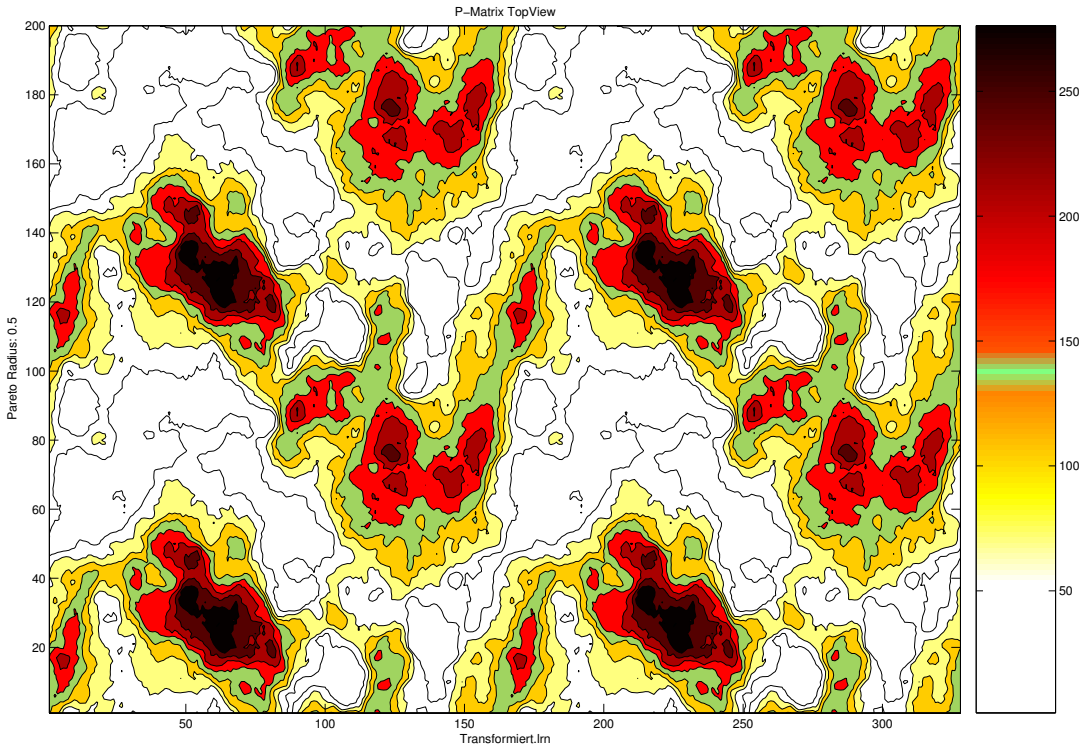


Abbildung C.6.: Die Abbildung zeigt die P-Matrix zu der mit den geometrischen Eigenschaften der lokalen Regionen antrainierten ESOM.

Anhand der P-Matrix sind die lokalen Regionen geclustert worden [Kupas et al., 2004b]. Alle Datenpunkte innerhalb eines zusammenhängenden Bereiches mit großer Datendichte wurden einem Cluster zugewiesen. Weniger dichte Regionen bilden die Clustergrenzen. Datenpunkte in Regionen mit geringer Dichte wurden keinem Cluster zugewiesen. Die lokalen Regionen sind mit dem P-Matrix-Clusterverfahren in sechzehn Cluster eingeteilt worden. Die Abbildung C.8 zeigt die U*-Matrix mit den sechzehn Klassen. Verschiedene Farben der Datenpunkte bedeuten dabei unterschiedliche Klassenzugehörigkeit. Alle Datenpunkte, die mit der P-Matrix einer Klasse zugeordnet wurden, liegen in den Bereichen auf der Karte mit geringen Distanzen zwischen den Datenpunkten. Diejenigen in Bereichen mit großen Distanzen sind keiner Klasse zugewiesen worden.

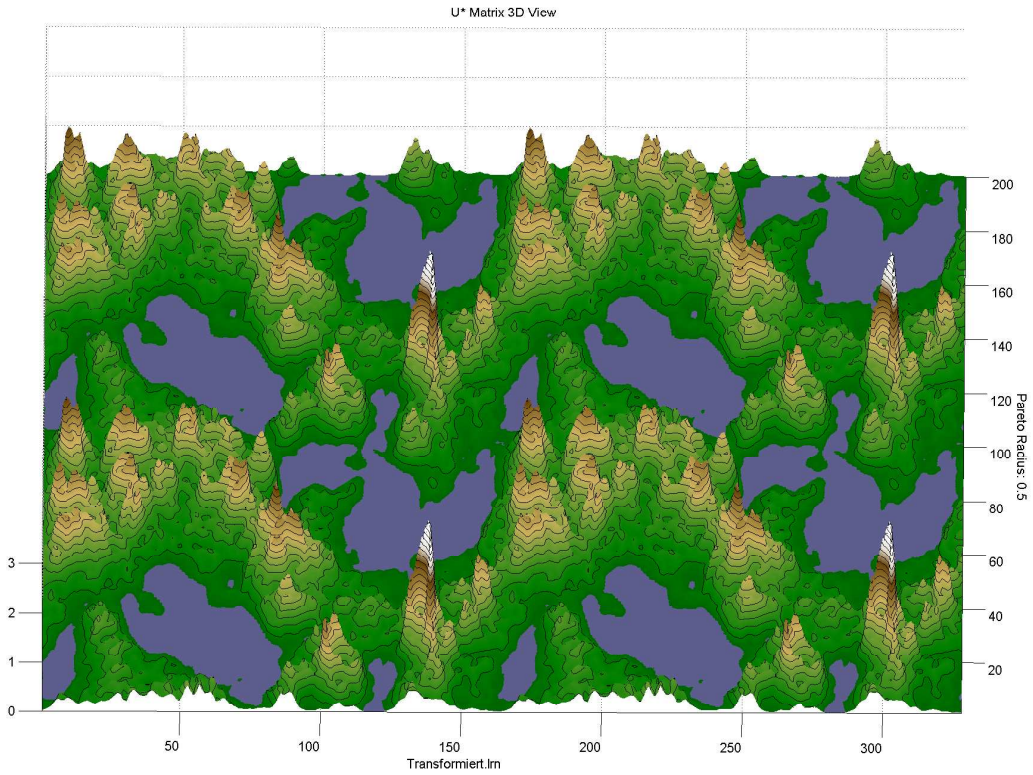


Abbildung C.7.: Die Abbildung zeigt die U*-Matrix zu der mit den geometrischen Eigenschaften der lokalen Regionen antrainierten ESOM.

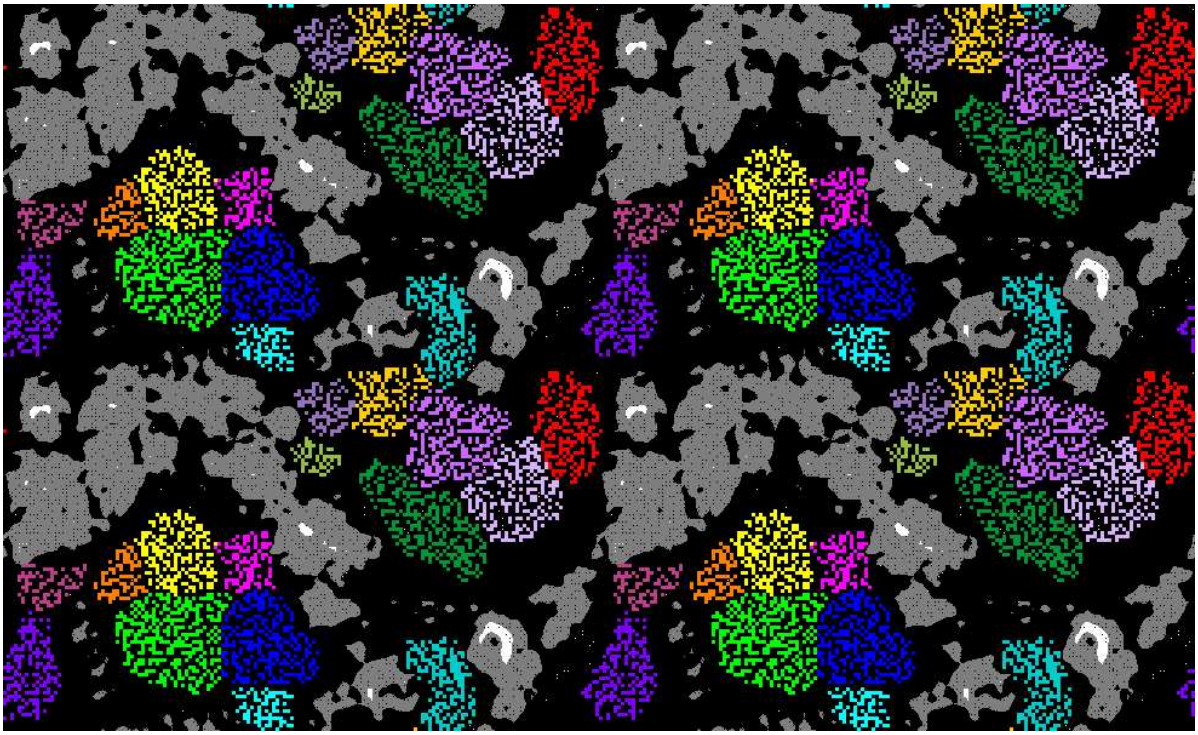


Abbildung C.8.: U*-Matrix mit den 16 P-Matrix-Klassen. Unterschiedliche Farben bedeuten unterschiedliche Klassenzugehörigkeit, gleiche Farbe bedeutet eine Klasse.

Die Validität der Clusterung nach den geometrischen Eigenschaften ist mit dem sig*-Algorithmus überprüft worden (siehe Kapitel C). Diese sechzehn P-Matrix-Cluster sind dann weiter in Subcluster gemäß der physikochemischen Eigenschaften der lokalen Regionen unterteilt worden.

Vergleich der ESOM-Clusterung mit der vorgegebenen EC-Klassifizierung

Viele der oben beschriebenen Subcluster enthalten hauptsächlich lokale Regionen von Enzymen, die derselben vierstelligen EC-Nummer zugewiesen sind. Ein Subcluster wird deswegen als charakteristisch definiert, wenn lokale Regionen von 80% der in dem Datensatz enthaltenen Enzyme derselben EC-Nummer in dem Subcluster vertreten sind. Die Tabelle C.2 enthält für jede der EC-Klassen in dem Datensatz die Nummern der ESOM-Cluster, die charakteristische Subcluster enthalten. Bei den meisten EC-Klassen zeigt sich, dass die zugehörigen charakteristischen Subcluster nur in wenigen ESOM-Clustern enthalten sind.

Tabelle C.2.: Nummern der ESOM-Cluster, die charakteristische Subcluster für die jeweilige EC-KLasse enthalten

Name	**Cluster-Nummer**
Aldehyde reductase	—
Isocitrate dehydrogenase (NADP(+))	1, 7, 13
4-hydroxybenzoate 3-monooxygenase	7, 9, 10, 12, 13
Protein kinase	—
dTMP kinase	—
Subtilisin	4
Penicillopepsin	4, 12
Carbonate dehydratase	4, 12
Methionine gamma-lyase	3, 4, 5, 11, 12, 13
Phosphoglycerate mutase	1, 2, 4, 6, 7, 10, 11, 13, 16
Glutathione synthase	1, 2, 4, 12
UDP-N-acetylmuramoylalanine–D-glutamate ligase	2, 7, 9, 12

Für die Enzyme der Aldose-Reduktasen, Proteinkinasen und Thymidylat-Kinasen sind keine charakteristischen Subcluster gefunden worden. Die Enzyme zu diesen EC-Nummern zeigen eine große konformative Variabilität in ihren Bindetaschen. Da die lokalen Regionen über nächste Nachbarn generiert werden, kann es sein, dass sich die Nachbarschaftsstrukturen der Pseudozentren bei Konformationsänderungen auch verändern und so andere Pseudozentren zu lokalen Regionen zusammengefügt werden.

Die charakteristischen Subcluster werden als signifikant definiert, wenn die entsprechende EC-Klasse in dem zugehörigen ESOM-Cluster anteilmäßig nicht überrepräsentiert ist. Zur Identifizierung der signifikanten Subcluster ist ein Binomialmodell verwendet worden, welches die signifikanten Unterschiede der erwartetetn gegenüber den tatsächlichen lokalen Regionen dieser EC-Nummer basierend auf der Gesamtzahl lokaler Regionen von Enzymen dieser EC-Nummer und der Klassengröße berechnet. Der Fehlerwert α wurde als 0.05 gewählt.

Die Tabelle C.3 zeigt für jede Enzymfamilie die Nummern der ESOM-Cluster, die signifikante Unterschiede in der Anzahl der lokalen Regionen aufweisen. Dabei bedeutet die Spalte „under“, dass weniger lokale Regionen, die Spalte „over“, dass mehr lokale Regionen als erwartet in den aufgeführten ESOM-Clustern auftreten.

Tabelle C.3.: Nummer der ESOM-Cluster, die signifikante Anzahlen lokaler Regionen enthalten

Name	**under**	**over**
Aldehyde reductase	2, 9, 16	—
Isocitrate dehydrogenase (NADP(+))	—	—
4-hydroxybenzoate 3-monooxygenase	—	11
Protein kinase	16	—
dTMP kinase	9, 16	15
Subtilisin	—	16
Penicillopepsin	14, 16	—
Carbonate dehydratase	—	4, 12
Methionine gamma-lyase	—	13
Phosphoglycerate mutase	—	12
Glutathione synthase	—	—
UDP-N-acetylmuramoylalanine–D-glutamate ligase	—	9

Signifikante Subcluster können nun mit Hilfe der Tabellen C.2 und C.3 leicht identifiziert werden. Die zugehörigen ESOM-Cluster der signifikanten Subcluster sind in Tabelle C.2 aufgeführt, stehen aber nicht in der „over“-Spalte der Tabelle C.3. So sind zum Beispiel alle charakteristischen Subcluster für die 4-Hydroxybenzoat-3-Monooxygenasen auch signifikant, da sie nicht in Cluster Nr. 11 vorkommen, in dem diese Enzymfamilie anteilsmäßig überrepräsentiert ist und somit in der „over“-Spalte steht. Im Gegensatz dazu sind alle für die Carbonat-Dehydratasen charakteristischen Subcluster nicht signifikant, da alle ESOM-Cluster, die diese charakteristischen Subcluster enthalten, in der „over“-Spalte der Tabelle C.3 aufgeführt sind.

Scoring der Übereinstimmungen zwischen den Enzymen mit lokalen Regionen in einem charakteristischen Subcluster

Alle Bindetaschen von Enzymen, die lokale Regionen in demselben charakteristischen Subcluster haben, sind anhand dieser dort geclusterten lokalen Regionen überlagert worden und es ist der *Score* für diese Überlagerung berechnet worden. Wie erwartet, haben alle die Paare von Bindetaschen zu der gleichen EC-Nummer einen hohen *Score* erhalten. Die Abbildung C.9 zeigt eine solche Überlagerung für zwei Bindetaschen zur gleichen EC-Nummer. In den charakteristischen Subclustern sind aber auch Enzyme zu unterschiedlichen EC-Nummern enthalten. Die Überlagerungen solcher Paare von Bindetaschen mit verschiedenen EC-Nummern haben einen sehr niedrigen *Score*. Meist hatten diese Bindetaschen außer dieser einen lokalen Region keine Gemeinsamkeiten. Die Abbildung C.10 zeigt eine solche Überlagerung zweier Taschen mit unterschiedlicher EC-Nummer, bei der als einzige Gemeinsamkeit die lokale Region auftritt.

Durch den Filterschritt, der alle Überlagerungen mit einem Score $S \leq 8$ herausfiltert, wird jede solche nur in einer lokalen Region übereinstimmende Überlagerung eliminiert. Alle Überlagerungen, bei der die zugehörigen Enzyme zur gleichen EC-Nummer gehören, haben einen hohen *Score*. Alle die Überlagerungen, bei denen die zugehörigen Enzyme unterschiedliche EC-Nummern hatten, wurden herausgefiltert. Somit ist über die charakteristischen Subcluster und das anschließende Scoring eine Klassifikation der Enzyme gemäß ihrer EC-Nummern entstanden.

Problematik der Methode

Eine lokale Region, bestehend aus vier Pseudozentren, ist zum einen zu klein, um sicher Ähnlichkeiten in Proteinbindetaschen zu detektieren. Über die charakteristischen

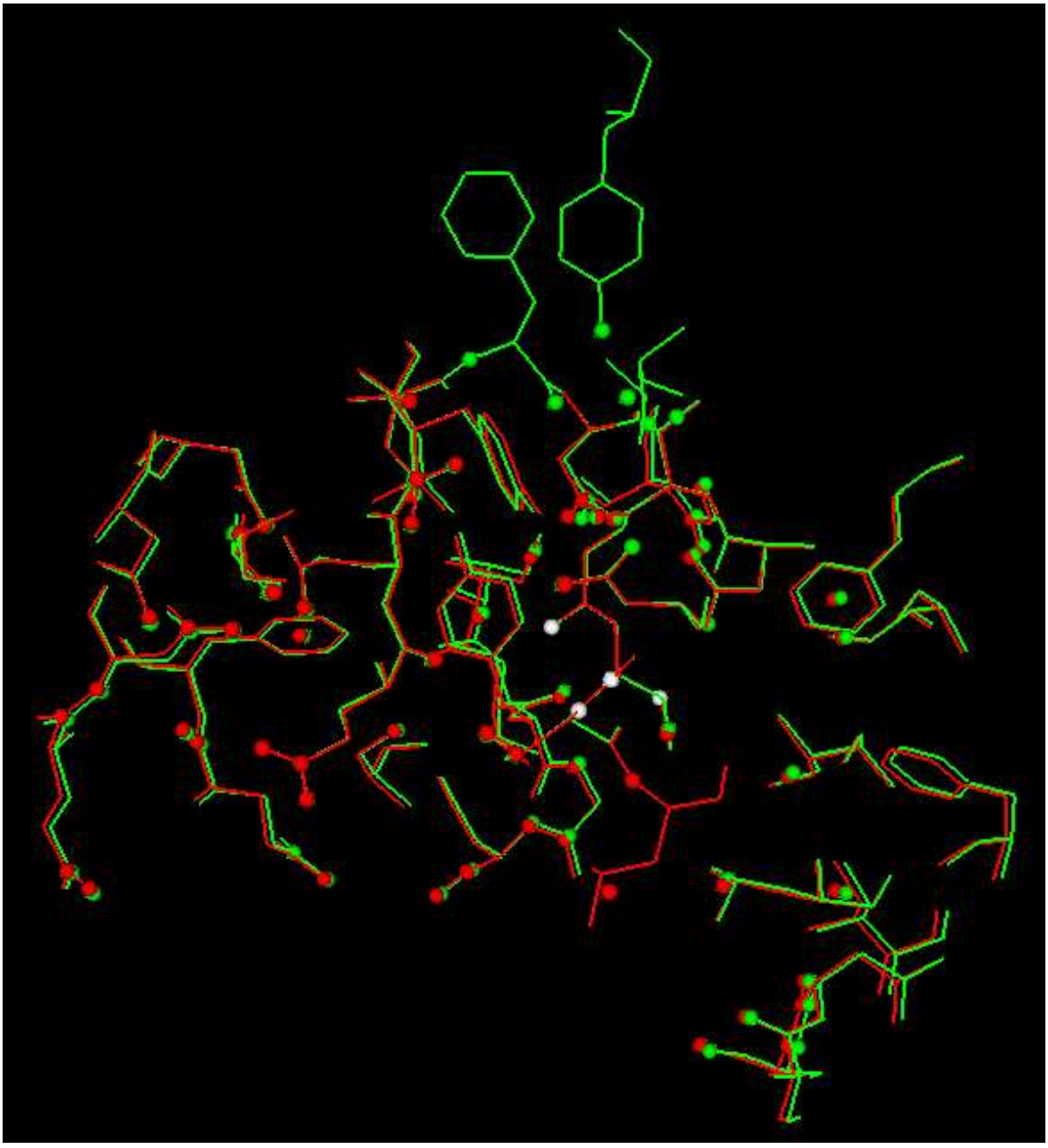


Abbildung C.9.: Überlagerung der Bindetaschen zweier verschiedener Strukturen des Penicillopepsin. Beide Enzyme haben die gleiche EC-Nummer. Die Pseudocenter der lokalen Region, die die Basis für die Überlagerung darstellen, sind weiß dargestellt. Die Aminosäuren und die Pseudozentren der Struktur mit dem PDB-Code 1apv sind rot, die der Struktur mit dem PDB-Code 1apw sind grün gefärbt. Diese Bindetaschen haben 60 Pseudozentren gemeinsam und damit ist der *Score* hoch genug, um den Filterschritt zu passieren.

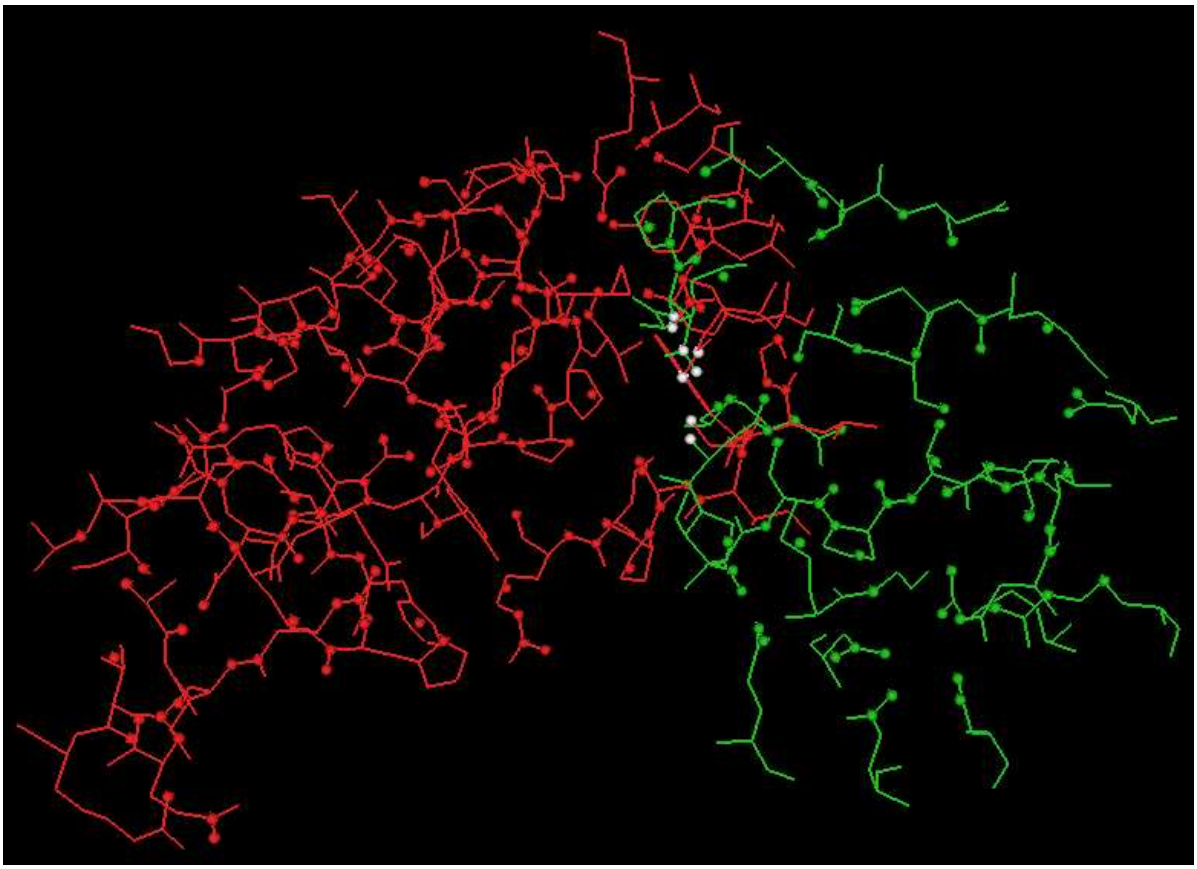


Abbildung C.10.: Überlagerung der beiden Bindetaschen der Enzyme Hydroxybenzoate Hydroxylase und Glutathion Synthase mit unterschiedlichen EC-Nummern. Die Pseudocenter der lokalen Region, die die Basis für die Überlagerung darstellen, sind weiß dargestellt. Die Aminosäuren und die Pseudozentren der Hydroxybenzoate Hydroxylase sind rot, die der Glutathion Synthase sind grün gefärbt. Die beiden Bindetaschen haben ausser der lokalen Region keine Gemeinsamkeiten. Der *Score* für diese Überlagerung ist also vier, und somit wird dieses Paar von Bindetaschen im Filterschritt eliminiert, da die Gemeinsamkeit zufällig und damit bedeutungslos ist.

Subcluster und den anschließenden Scoring-Schritt sind die Enzyme zwar gemäß ihrer EC-Nummer klassifiziert worden, es sind aber sehr viele Subcluster aufgetreten, die lokale Regionen von Enzymen aus anderen EC-Klassen im Datensatz zusammenfassen. Zudem sind auf Basis der so definierten lokalen Regionen nicht für jede EC-Klasse charakteristische Subcluster gefunden worden.

Zum anderen werden die lokalen Regionen über die nächsten Nachbarn generiert. Bei dicht liegenden Pseudozentren ändern sich im Falle der konformativen Adaption von Proteinen auf gebundene Liganden auch die Nachbarschaftsbeziehungen zwischen den Pseudozentren. Somit entstehen dann unter Umständen sehr unterschiedliche lokale Regionen, je nachdem, in welcher Konformation eine Tasche vorliegt.

Eine Methode, die größere Substrukturen verwendet, die nicht so stark abhängig von einem starren Gefüge der zugeordneten Pseudozentren ausgeht, sollte daher zu deutlich besser diskriminierenden und klassifizierenden Ergebnissen führen.

Abbildungsverzeichnis

2.1. Quadgrid vs. Hexgrid 10
2.2. Farbcodierung der U-Matrix 12
2.3. Unterteilung eines Dreiecksnetzes 19
2.4. Skalierungs- und Waveletfunktion 20
2.5. Berechnung der Waveletkoeffizienten 21
2.6. Berechnung des Skalierungsfaktors 22

5.1. Entfaltung von Oberflächenpatches 52
5.2. Schema der Nummerierung der Sektoren der Oberflächenpatches 55
5.3. Parameterstudien zur Methode SectorGeoMap 59
5.4. Parameterstudien zur Methode WaveGeoMap 59

6.1. U-Matrix der SectorGeomap für drei Bindetaschen 66
6.2. Patches der Taschen 1cil und 1cin aus derselben U-Matrix-Klasse 67
6.3. U-Matrix zum Validierungdatensatz *DS V1* 70
6.4. Politische Karte zum Validierungdatensatz *DS V1* 71
6.5. Überlagerung der Patches des Cluster Nr. 32 des Validierungdatensatzes *DS V1*. 73
6.6. U-Matrix zum erweiterten Validierungdatensatz *DS V2* 75
6.7. Politische Karte zum Validierungdatensatz *DS V2* 76
6.8. Patches des Cluster Nr. 206 des erweiterten Validierungdatensatzes *DS V2* und ihre Überlagerung. 79
6.9. U-Matrix zum Testdatensatz *DS T1* 81
6.10. Ausschnitt 1 der U-Matrix zu den Oberflächenpatches charakterisiert durch Wavelets 82
6.11. Ausschnitt 2 der U-Matrix zu den Oberflächenpatches charakterisiert durch Wavelets 83
6.12. Ausschnitt 3 der U-Matrix zu den Oberflächenpatches charakterisiert durch Wavelets 84

6.13. Ausschnitt 4 der U-Matrix zu den Oberflächenpatches charakterisiert durch Wavelets 85
6.14. Ausschnitt 5 der U-Matrix zu den Oberflächenpatches charakterisiert durch Wavelets 86
6.15. Ausschnitt 6 der U-Matrix zu den Oberflächenpatches charakterisiert durch Wavelets 87
6.16. Politische Karte zum Testdatensatz *DS T1* 88
6.17. Patches des Cluster Nr. 119 des Testdatensatzes *DS T1.* 89
6.18. Überlagerte Patches des Cluster Nr. 119 des Testdatensatzes *DS T1.* . . 90
6.19. U-Matrix zu Trypsin und Thrombin 91
6.20. U-Matrix zum Testdatensatz *DS T2* 93
6.21. Politische Karte zum Testdatensatz *DS T2* 94
6.22. Überlagerung der Patches zweier Thrombin-Strukturen 95
6.23. U-Matrix zum Datensatz *DS T2* mit markierten Clustern 97
6.24. Überlagerung der Aminosäuren von Serinproteasen zu den Patches eines Clusters 98
6.25. Überlagerung der Aminosäuren von Serinproteasen zu den Patches eines Clusters 99
6.26. Überlagerung der Aminosäuren von Aspartylproteasen zu den Patches eines Clusters 100
6.27. Überlagerung der Aminosäuren von Aspartylproteasen zu den Patches eines Clusters 101
6.28. Projektion des Datensatzes *DS P* auf die U-Matrix zum Datensatz *DS T2* 103
6.29. Ausschnitt der Projektion des Datensatzes *DS P* auf die U-Matrix zum Datensatz *DS T2* 104
6.30. Vergleich verschiedener Clusteralgorithmen 107
6.31. Target-Datensatz 109
6.32. Ergebnisse der Clusterung des Target-Datensatzes mit *K-means*, *Single Linkage* und *Ward.* 109

7.1. Ergebnis der MDS des Datensatzes *DS V1* auf zwei Dimensionen 118

C.1. *PDE-Plot* der Original- und transformierten Werte für die Höhe der lokalen Regionen 134
C.2. *PDE-Plot* der Original- und transformierten Werte für die Schiefe der lokalen Regionen 134

C.3. *PDE-Plot* der Original- und transformierten Werte für die Fläche der lokalen Regionen 135
C.4. 3D-Plot der geometrischen Merkmale der lokalen Regionen 136
C.5. U-Matrix mit den Bestmatches der lokalen Regionen 139
C.6. P-Matrix der lokalen Regionen 140
C.7. U*-Matrix der lokalen Regionen 141
C.8. U*-Matrix mit den 16 P-Matrix-Klassen 142
C.9. Überlagerung der Bindetaschen zweier verschiedener Strukturen des Penicillopepsin 146
C.10. Überlagerung der beiden Bindetaschen der Enzyme Hydroxybenzoate Hydroxylase und Glutathion Synthase mit unterschiedlichen EC-Nummern . 147

Tabellenverzeichnis

3.1. Zusammenfassung der Eigenschaften der Ansätze zur Klassifizierung von Proteinen 39

4.1. Tabelle mit den EC-Nummern, den Namen und der Anzahl der Strukturen des Datensatzes *DS V1* (109 Einträge) 43
4.2. Tabelle mit den EC-Nummern und den Namen der Strukturen des Datensatzes *DS V2* (708 Einträge) 44
4.3. Tabelle mit den EC-Nummern und den Namen der Strukturen des Datensatzes *DS T1* (1494 Einträge) 45
4.4. Tabelle mit den EC-Nummern und den Namen der Strukturen des Datensatzes *DS T2* (283 Einträge) aus Endopeptidasen verschiedener Familien 46
4.5. Tabelle mit den EC-Nummern und den Namen der Strukturen des Datensatzes *DS P* (118 Einträge) aus Endopeptidasen verschiedener Familien 47

5.1. Parameterstudien zur Patchgröße 61

6.1. Clusternummern, PDB-Codes, EC-Nummern und Namen der Enzyme für die quasi-reinen Cluster des erweiterten Validierungdatensatzes *DS V2* . 77
6.2. Vergleich mit anderen Clusteralgorithmen 106
6.3. Zusammenfassung der Ergebnisse der Clusterung von Oberflächenpatches 111

C.1. Deskriptive Statistik des Datensatzes von 8462 lokalen Regionen, generiert aus 72 Bindetaschen unterschiedlicher EC-Nummern 133
C.2. Nummern der ESOM-Cluster, die charakteristische Subcluster für die jeweilige EC-KLasse enthalten 143
C.3. Nummer der ESOM-Cluster, die signifikante Anzahlen lokaler Regionen enthalten 144

Literaturverzeichnis

V. Anantharaman, L. Aravind, and E. V. Koonin. Emergence of diverse biochemical activities in evolutionarily conserved structural scaffolds of proteins. *Curr Opin Chem Biol*, 7:12–20, 2003.

M. Ankerst, G. Kastenmüller, H.-P. Kriegel, and T. Seidl. 3D Shape Histograms for Similarity Search and Classification in Spatial Databases. In *Proc. 6th Int. Symposium on Spatial Databases (SSD' 99), Hong Kong, China*, volume 1651 of *Lecture Notes in Computer Science*, pages 207–226. Springer, Juli 1999a.

M. Ankerst, G. Kastenmüller, H.-P. Kriegel, and T. Seidl. Nearest Neighbor Classification in 3D Protein Databases. In *Proc. 7th Conference on Intelligent Systems for Molecular Biology (ISMB' 99), Heidelberg*, pages 34–43. AAAI Press, 1999b.

P. J. Artymiuk, A. R. Poirette, H. M. Grindley, D. W. Rice, and P. Willet. A Graph-theoretic Approach to the Identification of Three-dimensional Patterns of Amino Acid Side-chains in Protein Structures. *J Mol Biol*, 243:327–344, 1994.

A. Bairoch. The ENZYME database in 2000. *Nucleic Acids Res*, 28(1):304–305, 2000.

S. Bergs. *Optimalität bei Clusteranalysen*. PhD thesis, Universität Münster, 1981.

B. Boeckmann, A. Bairoch, R. Apweiler, M. Blatter, A. Estreicher, E. Gasteiger, M. Martin, K. Michoud, C. O'Donovan, I. Phan, S. Pilbout, and M. Schneider. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res*, 31:365 – 70, 2003.

P. Bucher and A. Bairoch. A generalized profile syntax for biomolecular sequence motifs and its function in automatic sequence interpretation. . In *Proc Int Conf Intell Syst Mol Biol*, pages 53–61, 1994.

T. Cormen, C. Leiserson, and R. Rivest. *Introduction to Algorithms*. MIT Press / McGraw-Hill, 1990.

M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution Analysis of Arbitrary Meshes. Technical report, University of Washington, Seattle, WA, Microsoft Research, Redmond, WA, Alias Research, Toronto, Ontario, Canada, January, 2nd 1995.

D. Fischer, C. Tsai, R. Nussinov, and H. Wolfson. A 3D sequence-independent representation of the protein data bank. *Protein Eng*, 8(10):981–997, 1995. URL `http://protein.oupjournals.org/cgi/content/abstract/8/10/981`.

H. M. Grindley, P. J. Artymiuk, D. W. Rice, and P. Willet. Identification of tertiary structure resemblance in proteins using a maximal common subgraph isomorphism Algorithm. *J Mol Biol*, 229:707–721, 1993.

D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining*. MIT Press, 2001.

M. Hendlich, F. Rippmann, and G. Barnickel. LIGSITE: Automatic and efficient detection of potential small molecule-binding sites in proteins. *J Mol Graph*, 15:359–363, 1997.

M. Hendlich, A. Bergner, J. Günther, and G. Klebe. Relibase: Design and Development of a Database for Comprehensive Analysis of Protein-Ligand Interactions. *J Mol Biol*, 326:607–620, 2003.

L. Holm and C. Sander. Mapping the protein universe. *Science*, 273:595–602, 1996a.

L. Holm and C. Sander. The FSSP database: fold classification based on structure-structure alignment of proteins. *Nucleic Acids Res*, 24:206–209, 1996b.

L. Holm and C. Sander. Searching protein structure databases has come of age. *Proteins*, 19:165–73, 1994.

S. Holm. Touring protein fold space with DALI/FSSP. *Nucleic Acids Research*, 26(1): 316–319, 1998.

IHG. Finishing the euchromatic sequence of the human genome. *Nature*, 431:915–916, 2004.

A. Jalobeanu. Fractal 3D Modeling of Asteroids using Wavelets on Arbitrary Meshes. In *IAFA 2003, Bucharest (Romania)*, 2003.

M. Jambon, A. Imberty, G. Deléage, and C. Geourjon. A New Bioinformatic Approach to Detect Common 3D Sites in Protein Structures. *PROTEINS: Structure, Function, and Genetics*, 52:137–145, 2003.

S. Kaski. *Data Exploration Using Self-Organizing Maps*. PhD thesis, Helsinki University of Technology, 1997.

G. Kastenmüller, H.-P. Kriegel, and T. Seidl. Similarity Search in 3D Protein Databases. In *Proc. German Conference on Bioinformatics (GCB' 98), Köln*, 1998.

K. Kinoshita and H. Nakamura. Identification of protein functions from a molecular surface database: eF-site. *Journal of Structural and Functional Genomics*, 2:9–22, 2001.

K. Kinoshita and H. Nakamura. Identification of protein biochemical functions by similarity search using the molecular surface database eF-site. *Protein Science*, 12: 1589–1595, 2003.

T. Kohonen. Self-Organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59–69, 1982.

T. Kohonen. *Self-Organizing Maps*. Springer, Berlin, 1997.

K. Kupas and A. Ultsch. Data Mining in Protein Binding Cavities. In C. Weihs and W. Gaul, editors, *Classification - the Ubiquitous Challenge, Proceedings 28th Annual Conference of the German Classification Society (GfKl 2004), Dortmund, Germany*, pages 354–361. Springer, Heidelberg, 2005.

K. Kupas, A. Ultsch, and G. Klebe. An algorithm for finding similarities in protein active sites. In M. He, G. Narasimhan, and S. Petoukhov, editors, *Advances in Bioinformatics and its Applications, Proceedings of the International Conference, Nova Southeastern University, Fort Lauderdale, Florida, USA*, volume 8 of *Series in Mathematical Biology and Medicine*, pages 373–380. World Scientific, December 16-19 2004a.

K. Kupas, A. Ultsch, and G. Klebe. Comparison of substructural epitopes in enzyme active sites using self-organizing maps. *J Comput Aided Mol Des*, 18:697–708, 2004b. doi: 10.1007/s10822-004-6553-x.

E. Lander. Initial sequencing and analysis of the human genome. *Nature*, 409(6822): 860–921, 2001.

R. Laskowski. SURFNET: a program for visualizing molecular surfaces, cavities, and intermolecular interactions. *J Mol Graph*, 13(5):323–30, 1995.

J. V. Lehtonen, K. Denessiouk, A. C. May, and M. S. Johnson. Finding local structural similarities among families of unrelated protein structures: a generic nonlinear alignment algorithm. *PROTEINS: Structure, Function, and Genetics*, 34:341–355, 1999.

N. Leibowitz, Z. Y. Fligelman, R. Nussinov, and H. J. Wolfson. Multiple Structural Alignment and Core Detection by Geometric Hashing. In T. Lengauer, R. Schneider, P. Bork, D. L. Brutlag, J. I. Glasgow, H.-W. Mewes, and R. Zimmer, editors, *Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology, August 6-10, 1999, Heidelberg, Germany*, pages 169–177. AAAI Press, 1999.

N. Leibowitz, R. Nussinov, and H. Wolfson. MUSTA - A General, Efficient, Automated Method for Multiple Structure Alignment and Detection of Common Motifs: Application to Proteins. *Journal of Computational Biology*, 8(2):93–121, 2001.

U. Lessel and D. Schomburg. Similarities between protein 3-D structures. *Protein Eng*, 7(10):1175–87, October 1994.

L. Lo Conte, S. E. Brenner, T. J. Hubbard, C. Chothia, and A. G. Murzin. SCOP database in 2002: refinements accommodate structural genomics. *Nucleic Acids Res*, 30:264–267, 2002.

M. Lounsbery, T. DeRose, and J. Warren. Multiresolution Analysis for Surfaces of Arbitrary Topological Type. *ACM Transactions on Graphics*, 16(1):34–73, January 1997.

V. Luc and P. Soille. Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 13(6):583–598, 1991.

A. C. Martin, C. A. Orengo, E. G. Hutchinson, S. Jones, M. Karmirantzou, R. A. Laskowski, J. B. Mitchell, C. Taroni, and J. M. Thornton. Protein folds and functions. *Structure*, 6:875–884, 1998.

K. Mason, N. Patel, A. Ledel, C. Moallemi, and E. Wintner. Mapping protein pockets through their potential small-molecule binding volumes: QSCD applied to biological protein structures. *Journal of Computer-Aided Molecular Design*, 18:55–70, 2004.

F. Mörchen. Time series feature extraction for data mining using dwt and dft. Reihe Informatik 33, Fachbereich Mathematik und Informatik, Philipps-Universität Marburg, October 2003.

R. Morris, R. Najmanovich, A. Kahraman, and J. Thornton. Real spherical harmonic expansion coefficients as 3D shape descriptors for protein binding pocket and ligand comparisons. *Bioinformatics*, 21(10):2347–2355, February 2005. doi: 10.1093/bioinformatics/bti337.

A. G. Murzin, S. E. Brenner, T. Hubbard, and C. Chothia. SCOP: A Structural Classification of Proteins Database for the Investigation of Sequences and Structures. *J Mol Biol*, 247:536–540, 1995.

N. Nagano, C. A. Orengo, and J. M. Thornton. One fold with many functions: the evolutionary relationships between TIM barrel families based on their sequences, structures and functions. *J Mol Biol*, 321:741–765, 2002.

S. B. Needleman and W. C. D. A general method applicable to the search for similarities in the amino acids of two proteins. *J Mol Biol*, 48:443–453, 1970.

R. Nussinov and H. J. Wolfson. Efficient detection of three-dimensional structural motifs in biological macromolecules by computer vision techniques. *Proc. Natl. Acad. Sci. USA*, 88:10495–10499, December 1991.

C. A. Orengo, A. D. Michie, S. Jones, D. T. Jones, M. B. Swindells, and J. M. Thornton. CATH–a hierarchic classification of protein domain structures. *Structure*, 5:1093–1108, 1997.

C. A. Orengo, I. Sillitoe, G. Reeves, and F. M. Pearl. Review: what can structural classifications reveal about protein evolution? *J Struct Biol*, 134:145–165, 2001.

C. Oshiro and I. Kuntz. Characterization of receptors with a new negative image: use in molecular docking and lead optimization. *Proteins*, 30(3):321–36, Feb. 1998.

W. R. Pearson. Rapid and sensitive sequence comparison with FASTP and FASTA. *Methods Enzymol*, 183:63–98, 1990.

W. R. Pearson and D. J. Lipman. Improved tools for biological sequence comparison. *Proc. Natl. Acad. Sci. USA*, 85:2444–2448, 1988.

S. Pickering, A. Bulpitt, N. Efford, N. Gold, and D. Westhead. AI-based algorithms for protein surface comparisons. 26(1):79–84, December 2001.

A. R. Poirrette, P. J. Artymiuk, D. W. Rice, and P. Willett. Comparison of protein surfaces using a genetic algorithm. *J Comput Aided Mol Des*, 11:557–569, 1997.

N. Rawlings, F. Morton, and A. Barrett. MEROPS: the peptidase database. *Nucleic Acids Res*, 34:270–272, 2006.

M. Rosen, S. L. Lin, H. Wolfson, and R. Nussinov. Molecular shape comparisons in searches for active sites and functional similarity. *Protein Eng*, 11:263–277, 1998.

R. B. Russell. Detection of protein three-dimensional side-chain patterns: new examples of convergent evolution. *J Mol Biol*, 279:1211–1227, 1998.

S. Schmitt, D. Kuhn, and G. Klebe. A New Method to Detect Related Function Among Proteins Independent of Sequence and Fold Homology. *J Mol Biol*, 323(2):387–406, 2002.

I. Schomburg, A. Chang, and D. Schomburg. BRENDA, enzyme data and metabolic information. *Nucleic Acids Res*, 30:47–49, 2002.

P. Schröder and W. Sweldens. Spherical wavelets: Efficiently representing functions on the sphere. In *Proceedings of SIGGRAPH 95*, pages 161–172, Los Angeles, California, August 1995.

A. Shulman-Peleg, R. Nussinov, and H. Wolfson. Recognition of functional sites in protein structures. *J Mol Biol*, 339(3):607–33, June 2004.

R. Siemon. Einige Werkzeuge zum Einsatz von selbstorganisierenden Neuronalen Netzen zur Strukturanalyse von Wirkstoff-Rezeptoren. Master's thesis, AG Datenbionik, Fachbereich Mathematik & Informatik, Philipps-Universität Marburg, 2001.

R. V. Spriggs, P. J. Artymiuk, and P. Willett. Searching for patterns of amino acids in 3D protein structures. *J Chem Inf Comput Sci*, 43:412–421, 2003.

M. Stahl, C. Taroni, and G. Schneider. Mapping of protein surface cavities and prediction of enzyme class by a self-organizing neural network. *Protein Engineering*, 13(2):83–88, 2000.

W. R. Taylor and C. A. Orengo. Protein Structure Alignment. *J Mol Biol*, 208:1–22, 1989.

A. E. Todd, C. A. Orengo, and J. M. Thornton. Evolution of function in protein superfamilies, from a structural perspective. *J Mol Biol*, 307:1113–1143, 2001.

A. Ultsch. Pareto Density Estimation. In *Proc. GfKl 2003, Cottbus, Germany*, 2003a.

A. Ultsch. U*-Matrix: a Tool to visualize Clusters in high dimensional Data. In *Technical Report No. 36, Departement of Mathematics and Computer Science, Philipps-University Marburg, Germany*, 2003b.

A. Ultsch. Maps for the Visualization of high dimensional Data Spaces. In *Proc. WSOM, Japan*, 2003c.

A. Ultsch. Density Estimation and Visualization for Data containing Clusters of unknown Structure. In *Proc. GfKl 2004, Dortmund, Germany*, 2004.

A. Ultsch. Clustering with SOM: U*C. In *Proceedings Workshop on Self-Organizing Maps (WSOM 2005), Paris, France*, pages 75–82, 2005.

A. Ultsch. Self-Organizing Neural Networks for Visualization and Classification. In *Proc. Conf. Soc. for Information and Classification*, April 1992.

A. Ultsch. Data mining and knowledge discovery with emergent self-organizing feature maps for multivariate time series, 1999.

A. Ultsch and L. Herrmann. The architecture of emergent self-organizing maps to reduce projection errors. In V. M., editor, *Proceedings of the European Symposium on Artificial Neural Networks (ESANN 2005)*, pages 1–6, 2005.

A. Ultsch and H. Siemon. Kohonen's Self Organizing Feature Maps for Exploratory Data Analysis. In *Proc. Intern. Neural Networks, Kluwer Academic Press, Paris*, pages 305–308, 1990.

J. e. a. Venter. The sequence of the human genome. *Science*, 291(5507):1304–1351, 2001.

A. C. Wallace, N. Borkakoti, and J. M. Thornton. TESS: a geometric hashing algorithm for deriving 3D coordinate templates for searching structural databases. Application to enzyme active sites. *Protein Science*, 6:2308–2323, 1997.

P. P. Wangikar, A. V. Tendulkar, S. Ramya, D. N. Mali, and S. Sarawagi. Functional sites in protein families uncovered via an objective and automated graph theoretic approach. *J Mol Biol*, 326:955–978, 2003.

M. Waterman. General methods for sequence comparison. *Bull. Math. Biol.*, 46:473–500, 1984.

N. Weskamp, D. Kuhn, E. Hüllermeier, and G. Klebe. Efficient similarity search in protein structure databases by k -clique hashing. *Bioinformatics*, 20(10):1522–1526, February 2004.

Lebenslauf

Name	Katrin Kupas
Geburtstag	24. Mai 1979
Geburtsort	Duisburg
1986 - 1989	Grundschule
1989 - 1998	BMV-Schule Essen, Gymnasium
Juni 1998	Abitur
Okt 1998 - Aug 2000	Grundstudium des Diplomstudienganges Reine Mathematik an der BUGH Wuppertal mit Nebenfach Informatik
Aug 2000	Diplomvorprüfung in Reiner Mathematik
Okt 2000	Quereinstieg in den Diplomstudiengang Biomathematik an der EMAU Greifswald
Sep 2001	Diplomvorprüfung in Biomathematik
Juli 2003	Diplom in Biomathematik
Okt 2003	Beginn der Arbeiten zur vorliegenden Dissertation unter Betreuung von Prof. Ultsch am Fachbereich Mathematik & Informatik und Prof. Klebe am Institut für Pharmazeutische Chemie der Philipps-Universität Marburg
Okt 2003 - Sep 2006	wissenschaftliche Mitarbeiterin am Fachbereich Mathematik & Informatik der Philipps-Universität Marburg