



UNIVERSITÄT
KOBLENZ · LANDAU

Fachbereich 4: Informatik



Skelettierung von 3D-Objekten für die Ähnlichkeitsbestimmung

Bachelorarbeit
zur Erlangung des Grades
BACHELOR OF SCIENCE
im Studiengang Informationsmanagement

vorgelegt von

David Jonathan Barnowsky

Betreuer: Dr.-Ing. Marcin Grzegorzek, Institut für Computervisualistik,
Fachbereich Informatik, Universität Koblenz-Landau

Erstgutachter: Dr.-Ing. Marcin Grzegorzek, Institut für
Computervisualistik, Fachbereich Informatik, Universität Koblenz-Landau

Zweitgutachter: Prof. Dr.-Ing. Dietrich Paulus, Institut für
Computervisualistik, Fachbereich Informatik, Universität Koblenz-Landau

Koblenz, im August 2010

Kurzfassung

In der vorliegenden Arbeit wird dem Leser aufgezeigt, welche Methoden zur Kurvenskelettierung von 3D-Modellen existieren und welche Ansätze bei bisherigen Forschungsergebnissen von Skelettierungsmethoden in der Bildverarbeitung verfolgt werden.

Der Autor geht im weiteren Verlauf auf einen aktuellen Kurvenskelettierungsansatz ein. Die Implementierung dieses Ansatzes wird detailliert analysiert und ein algorithmisches Verfahren entwickelt, um die Kurvenskelette der analysierten Methode hin zu 3D-Skeletten einer eigenen Kurvenskelettdefinition zu modifizieren, mit der es möglich ist, Merkmalsvektoren für Ähnlichkeitsvergleiche zwischen 3D-Objekten zu berechnen.

Es wird eine Ground Truth durch menschliche Ähnlichkeitsbestimmung gebildet. Des Weiteren wird auf Grundlage der modifizierten Skelette eine Ähnlichkeitsberechnung durchgeführt. Danach wird die menschliche Ähnlichkeitsbestimmung der algorithmisch berechneten Ähnlichkeitsberechnung gegenüber gestellt und nach bekannten Verfahren aus dem Information Retrieval ausgewertet.

Abstract

The present work starts with an introduction of methods for three-dimensional curve skeletonization. Different kinds of historic and recent skeletonization approaches are analysed in detail.

Later on, a state-of-the-art skeletonization algorithm is introduced. This algorithm deals as a basis for the own approach presented subsequently. After the description and definition of a new method improving the state-of-the-art algorithm, experiments are conducted to get appraisable results.

Next, a ground truth is described which has been set up manually by humans. The human similarity evaluations are compared with the results of the automatic computer-based similarity measures provided by the own approach. For this comparison, standard evaluation criteria from the field of information retrieval have been used.

Vorgehensweise

Zuerst erfolgt die grundlegende Analyse des Forschungsgebietes und der Analyse aktueller Forschungsergebnisse zur Skelettierung von 3D-Objekten. Danach wird eine Ground Truth zur späteren Evaluierung des angewandten Ähnlichkeitsmaßes erstellt. Ein aktuelles kombiniertes Skelettierungsverfahren bildet die Grundlage zur weiteren Modifikation der daraus erhaltenen Kurvenskelette.

Die Methode der Modifikation bildet einen grundlegenden Teil der vorliegenden Arbeit. Die Skelettmodifikationen werden in Java durchgeführt. Dabei wird das auf C++ basierende Visual Toolkit (VTK) als Framework eingesetzt. Zudem wird auf Grundlage des VTK eine Voxelengine programmiert, mit der visuelle Tests durchgeführt werden können. Standbilder der Voxelengine werden in dieser Arbeit für die Darstellung von algorithmischen Problemen genutzt. Danach erfolgt die Ähnlichkeitsberechnung der modifizierten Kurvenskelette anhand bekannter Ähnlichkeitsmaße auf Basis von topologisch basierten Merkmalsvektoren.

Im letzten Teil werden Ähnlichkeiten zwischen der Ground Truth und dem algorithmisch berechneten Ähnlichkeitsmaß durch Methoden aus dem Information Retrieval durchgeführt. Diese Berechnung des Information Retrieval werden automatisiert in einem weiteren Java Programm durchgeführt.

Erklärung

Ich versichere, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Die Vereinbarung der Arbeitsgruppe für Studien- und Abschlussarbeiten habe ich gelesen und anerkannt, insbesondere die Regelung des Nutzungsrechts.

Mit der Einstellung dieser Arbeit in die Bibliothek bin ich einverstanden. ja nein

Der Veröffentlichung dieser Arbeit im Internet stimme ich zu. ja nein

Koblenz, den 31. August 2010

Inhaltsverzeichnis

1	Einführung	13
1.1	Problemstellung	13
1.2	Motivation	13
1.3	Eigener Beitrag	14
1.4	Überblick	14
2	Skelettierung von 3D-Objekten	17
2.1	Begriffserklärungen	17
2.1.1	Der dreidimensionale Raum	17
2.1.2	Euklidische Distanz	18
2.1.3	Geodätische Distanz	18
2.1.4	Voxelrepräsentation	18
2.1.5	Skelett	20
2.1.6	Skelettierung	21
2.2	Entstehung der 3D-Skelettierung	22
2.3	Skelettierungsansätze	23
2.3.1	Thinning	24
2.3.2	Voronoi-Diagramme	25
2.3.3	Distanzfeld	26
3	Eigener Lösungsansatz	29
3.1	Einführung	29
3.2	Skelettierung nach Reniers	30
3.3	Modifikationen der Skelette nach DF-Methode	34
3.3.1	Topologisch orientiertes Ähnlichkeitsmaß	34
3.3.2	Vorarbeit	35
3.3.3	Dijkstra-Skeleton Algorithmus	38
3.3.4	Beispiel einer Kurvenskelettumwandlung mit dem DSA	40
3.3.5	Programmstruktur	41
3.4	Ähnlichkeitsbestimmung	42
3.4.1	Algorithmische Ähnlichkeitsberechnung	42

3.4.2	Ähnlichkeitssuche in Objektdatenbanken	42
3.4.3	Anfrage-Definition	43
4	Experimente und Ergebnisse	47
4.1	Einführung	47
4.2	Experimentenmethodik	49
4.3	Merkmalsvektor- und Ähnlichkeitsberechnungen	51
4.4	Information Retrieval mit Recall- und Precisionergebnissen	52
4.5	Diskussion	52
5	Zusammenfassung und Ausblick	57
5.1	Zusammenfassung	57
5.2	Ausblick	58

Tabellenverzeichnis

3.1	Berechnung von Recall und Precision	44
3.2	Beispielberechnung Treffermenge H_G von Anfrageobjekt O_h	44
4.1	Objektbeschreibungen und Zuordnung	49
4.2	Ähnlichkeitswerte aus der Ground Truth	50
4.3	Berechnete Merkmalsvektoren	51
4.4	Ähnlichkeiten auf Grundlage der Merkmalsvektoren nach dem Kosinusmaß	55
4.5	Anfrage- sowie Recall und Precision Ergebnisse	56

Abbildungsverzeichnis

1.1	Überblick und Kapitelübersicht	15
2.1	Adjazenzmodell eines betrachteten Voxels	19
2.2	Grafische Beschreibung der Medialen Achse	23
2.3	Grafische Beschreibung der Berechnung von Skeletten im Zweidimensionalen Raum	23
2.4	Thinning-Algorithmus nach Kálmán und Kuba	25
2.5	Voronoi-Diagramm und Bestimmung des Kurvenskeletts	26
2.6	EFT bei der Bildung von Geodäten	26
3.1	Toleranzbasierte Feature Transformation	31
3.2	Berechnung, Ausweitung und Randextraktion von Jordankurven	33
3.3	Beispielskelett nach Reniers Kuvenskelettdefinition	36
3.4	Beispielskelett in der eigenen VTK-Voxelengine	36
3.5	Ausschnitt eines dilatatierten Kuvenskeletts	37
3.6	Anwendung morphologischer Operatoren	38
3.7	Konzeptionelle Darstellung eines Skelettgraphen	41
3.8	Klassendiagramm 'Main Package' zur Berechnung von Merkmalsvektoren	45
3.9	Klassendiagramm 'Util Package' zur Berechnung der Recall / Precision	46
4.1	Objekte aus der Ground Truth	48
4.2	Recall / Precision im RP-Diagramm	53
4.3	Topologisch dargestellte Kurvenskelette	53
5.1	Algorithmus von Bai und Latecki für den Ähnlichkeitsvergleich zwischen 2D-Objekten	59

Kapitel 1

Einführung

Das Ziel dieser Arbeit besteht zum einen in der Darstellung und Analyse vorhandener Skelettierungsverfahren von 3D-Objekten. Es wird daraufhin ein aktuelles Skelettierungsverfahren analysiert und erweitert. Ein weiteres Ziel besteht darin, ein einfaches topologisch orientiertes Ähnlichkeitsmaß zu definieren und zu implementieren. Dieses Ähnlichkeitsmaß wird daraufhin mit einer eigens für diese Arbeit erhobenen Ground Truth auf Analogien der Bewertung von Ähnlichkeiten hin verglichen.

1.1 Problemstellung

Der erste Teil der Problemstellung bildet die Einarbeitung in das Forschungsgebiet der Skelettierung. Dazu gehört die Analyse des historischen Verlaufs der Skelettierung bei 2D-Objekten und der schrittweisen Übertragung und Verbesserung von 3D-Verfahren zur Berechnung von Kurvenskeletten.

Im darauf folgenden Schritt sollen Kurvenskelette berechnet und für den Ähnlichkeitsvergleich vorbereitet werden. Dies geschieht durch die Definition einer neuen Methode und deren algorithmische Umsetzung.

Ein auf eigenen Methoden basierender Ähnlichkeitsvergleich bildet den letzten Teil der Problemstellung. Dieser Ähnlichkeitsvergleich wird zuerst definiert, beschrieben und danach umgesetzt, sodass Experimente durchgeführt und Ergebnisse ausgewertet werden können.

1.2 Motivation

Sollen aus 3D-Objekten Kurvenskelette erzeugt werden, so gibt es bislang wenige Verfahren, die dieses Problem lösen können. Werden auf Grundlage der Kurven-

skelette Ähnlichkeitsvergleiche durchgeführt, so existieren bisher ebenso wenige konkrete Ansätze, die Kurvenskelette bestmöglich für den Ähnlichkeitsvergleich einsetzen. In der 2D-Skelettierung gibt es sehr innovative Verfahren zur Ähnlichkeitsberechnung auf Grundlage von Kurvenskeletten, die die Vorteile der Kurvenskelette möglichst optimal ausnutzen. Ein innovatives Verfahren dieser Art von Bai und Latecki [BL08] wird in Kapitel 5 dargestellt. Es wird daher ein Einstieg in die Thematik vermittelt, dem ein anschließender Ähnlichkeitsvergleich auf Grundlage berechneter 3D-Kurvenskelette folgt.

1.3 Eigener Beitrag

Der eigene Beitrag besteht zum einen aus der Definition der notwendigen Modifikationen der Skelette eines bisherigen Skelettierungsverfahrens, welches aus 3D-Objekten voxelbasierte Kurvenskelette berechnet und zum anderen aus der Umsetzung dieser Definition in einen lauffähigen Algorithmus.

Daraufhin wird ein eigenes, topologisch basiertes Ähnlichkeitsmaß für die modifizierte Kurvenskelette definiert und umgesetzt. Alle Experimente werden auf der Grundlage dieser eigenen Methoden durchgeführt und deren Ergebnisse ausgewertet.

1.4 Überblick

Die Inhalte der vorliegenden Arbeit werden in logischer Abfolge vorgestellt. Nach der Beschreibung des Forschungsgebietes der 3D-Skelettierung werden neue Methoden zur Skelettierung und Ähnlichkeitsberechnung definiert und deren algorithmische Umsetzung beschrieben. Auf Grundlage dieser Ansätze werden Experimente entwickelt und durchgeführt und die daraus resultierenden Ergebnisse präsentiert und diskutiert. Daraufhin werden einige Denkanstöße zur Weiterentwicklung und Verbesserung der entwickelten Methoden gegeben.

Im **Kapitel 2** der vorliegenden Arbeit wird das Forschungsgebiet der Skelettierung vorgestellt und Begriffsdefinitionen aufgestellt und erläutert. Es werden verschiedene grundlegende Methoden der Skelettierung vorgestellt.

Daraufhin wird im **Kapitel 3** eine aktuelle Skelettierungsmethode nach Reniers [Ren08] im Detail analysiert und die Implementierung dieser Methode um einige wichtige Erweiterungen modifiziert. Die modifizierten Skelette bilden die Grundlage für die Definition eines Ähnlichkeitsmaßes und die Methode des Vergleiches

zwischen einer Ground Truth und dem algorithmisch berechneten Ähnlichkeitsmaß.

Im **Kapitel 4** werden Experimente zu den Objekten der Ground Truth und der algorithmischen Ähnlichkeitsbestimmung durchgeführt und die beiden Ähnlichkeitsmaße miteinander verglichen. Daraufhin wird eine Diskussion über die Ergebnisse angeführt.

Im **Kapitel 5** wird auf weitere Verfahren der Ähnlichkeitsbestimmung von 3D-Objekten auf Grundlage der Kurvenskelette eingegangen und es werden Anstöße und Vorschläge zur Weiterentwicklung und Verbesserung der vorgestellten Methoden gegeben.

In Abbildung 1.1 sind die Inhalte und die Aufeinanderfolge der Kapitel zum Verständnis grafisch aufbereitet dargestellt.

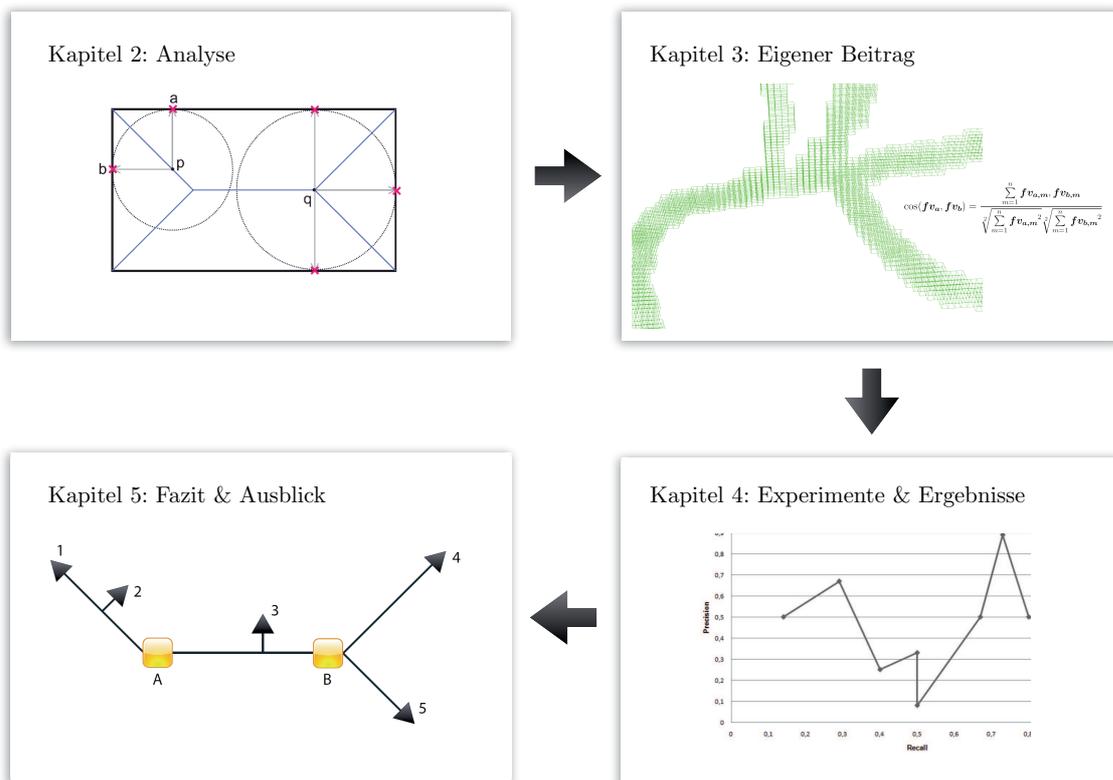


Abbildung 1.1: Überblick und Kapitelübersicht

Kapitel 2

Skelettierung von 3D-Objekten

2.1 Begriffserklärungen

In diesem Kapitel werden die grundlegenden Begriffe und Methoden der vorliegenden Arbeit analysiert und erklärt. Die in diesem Kapitel angegebenen Definitionen bilden die Grundlage für alle nachfolgenden Kapitel.

2.1.1 Der dreidimensionale Raum

Der visuelle dreidimensionale Raum wird auch als dreiminensionaler Euklidischer Raum \mathbb{R}^3 bezeichnet und bildet einen reellen Raum. In diesem reellen Raum kann nach Scheck [Sch07] mit der Euklidischen Geometrie gearbeitet werden. Wie in Scriba [SS09] nachzulesen, ist die wohlbekannte Euklidische Geometrie durch Punkte als atomare Teile des Raums, Linien als breitenlose Längen und Geraden als spezielle Linien, die bezüglich der auf ihr existierenden Punkte immer gleich liegen, beschrieben.

Zudem postuliert Euklid folgende Forderungen:

- Von jedem beliebigen zu jedem anderen beliebigen Punkt kann eine Linie gezogen werden.
- Jede begrenzte gerade Linie kann zusammenhängend verlängert werden.
- Mit jedem angegebenen Mittelpunkt und einem Abstand kann ein Kreis gezeichnet werden.
- Alle rechten Winkel sind immer gleich.
- Zu einer geraden Linie existiert zu einem gegebenen Punkt außerhalb der Linie höchstens eine parallele gerade Linie.

Die im Folgenden angeführten Definitionen dieses Kapitels beruhen auf der Annahme des reellen, dreidimensionalen Euklidischen Raumes \mathbb{R}^3 .

2.1.2 Euklidische Distanz

Die Euklidische Distanz, oder auch der Euklidische Abstand d zweier Punkte \mathbf{x}, \mathbf{y} , ist nach Euklid [Goh07] durch

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2.1)$$

definiert. Im dreidimensionalen Raum sind die Punkte \mathbf{x}, \mathbf{y} durch ihre drei Koordinaten

$$\mathbf{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \quad \text{und} \quad \mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} \quad (2.2)$$

bestimmt.

Ein Spezialfall bildet der Abstand zweier Punkte mit $n = 2$. Dieser Spezialfall ist als Satz des Pythagoras bekannt.

2.1.3 Geodätische Distanz

Die geodätische Distanz zwischen zwei Punkten, die auf der gleichen Oberfläche liegen, wird oft als geodätischer Pfad bezeichnet. Nach Patrikalakis und Meakawa [PM09] ist die geodätische Distanz zwischen zwei Punkten, die auf derselben Oberfläche liegen, der kürzeste Weg dieser Punkte zueinander.

Der kürzeste Pfad zwischen diesen zwei Punkten wird mit Algorithmen der kürzesten Pfade, wie zum Beispiel dem Dijkstra Algorithmus berechnet. Die Definition der geodätischen Distanz wird im Zusammenhang mit der vorliegenden Arbeit für die kürzesten Pfade zwischen zwei Punkten auf einer Objektoberfläche eines 3D-Objektes verwendet.

2.1.4 Voxelrepräsentation

In der Voxelrepräsentation wird der Raum in ein regelmäßiges Raster unterteilt. Jeder Teilbereich im Raumgitter wird als Voxel bezeichnet. Das Raumgitter besteht im Normalfall aus Würfeln und bietet so eine einfache algorithmische Behandlung in der Bildverarbeitung. Dabei existiert eine proportionale Abhängigkeit zwischen

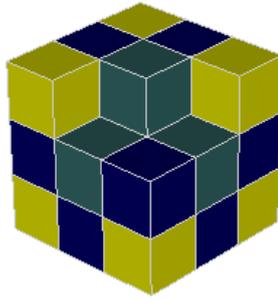


Abbildung 2.1: Adjazenzmodell eines betrachteten Voxels

dem Detailgrad, also der Exaktheit der Abbildung zur Auflösung des Rasters. Je höher die Auflösung des Rasters, desto exakter können geometrische Objekte abgebildet werden. Da der Raum in gleich große Teilbereiche (hier Würfel) unterteilt wird, spricht man deshalb vom diskreten Raum \mathbb{Z}^3 .

Das Adjazenzmodell im diskreten Raum ist die Kategorisierung von Nachbarvoxeln. Jeder Voxel $\mathbf{v} \in \mathbb{Z}^3$ hat 26 angrenzende Nachbarvoxel. In Abbildung 2.1 sind die Nachbar- oder Adjazenzvoxel in den drei Kategorien farblich dargestellt. Die sechs direktadjazenten Voxel (Kategorie 1) sind die Voxel, die durch eine Ebene mit dem betrachteten Voxel verbunden sind (grün dargestellt). Die 18 adjazenten Voxel, die durch eine Kante mit dem betrachteten Voxel zusammenhängen (Kategorie 2), sind blau dargestellt. Die letzte Kategorie 3 bilden die 8 Adjazenzvoxel, die nur durch eine Ecke mit dem betrachteten Voxel verbunden sind (gelb dargestellt).

So befinden sich zu jedem Voxel $\mathbf{v} \in \mathbb{Z}^3$, welcher in keiner Dimension x, y, z ein Randvoxel des diskreten Raumes \mathbb{Z}^3 bildet, $0 < v_x < x$, $0 < v_y < y$ und $0 < v_z < z$, 26 adjazente Voxel. Das Adjazenzmodell findet in der Berechnung von Distanzen zwischen Voxeln im diskreten Raum Anwendung. So sind die Euklidischen Distanzen zwischen den Kategorien von Adjazenzvoxeln unterschiedlich groß. Die Euklidische Distanz wird zwischen Kategorie 1 adjazenten Voxeln oft mit der Länge $l = 1$ definiert. Die Kategorie 2 adjazenten Voxel haben nach der Berechnung auf Grundlage der Definition von Kategorie 1 adjazenten Voxeln dann eine Euklidische Distanz mit der Länge von $l = \sqrt{2}$. Und die Kategorie 3 adjazenten Voxel werden durch eine Euklidische Distanz mit der Länge $l = \sqrt{3}$ berechnet.

Ein Beispiel zur Berechnung einer Distanz soll diesen Sachverhalt verdeutlichen. Der Mittelpunkt des betrachteten Voxels \mathbf{x}_v hat die Koordinaten

$$\mathbf{x}_v = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}. \quad (2.3)$$

Nach Definition ist der Abstand zwischen Kategorie 1 Adjazenzvoxeln genau 1. Ein Adjazenzvoxel \mathbf{y}_v in der Kategorie 2 und ein anderer Adjazenzvoxel \mathbf{z}_v in der Kategorie 3 haben hier die Koordinaten

$$\mathbf{y}_v = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \text{ und } \mathbf{z}_v = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}. \quad (2.4)$$

Der Euklidische Abstand beträgt somit

$$d(\mathbf{x}_v, \mathbf{y}_v) = \sqrt{\sum_{i=1}^3 (x_i - y_i)^2} = \sqrt{(0-1)^2 + (0-1)^2} = \sqrt{2} \quad (2.5)$$

$$d(\mathbf{x}_v, \mathbf{z}_v) = \sqrt{\sum_{i=1}^3 (x_i - y_i)^2} = \sqrt{(0-1)^2 + (0-1)^2 + (0-1)^2} = \sqrt{3} \quad (2.6)$$

Eine andere Möglichkeit, die Distanzen zwischen den Adjazenzvoxeln zu bestimmen, beschreiben Kiryati et al. [KS93]. Sie definieren die Längen durch ihre Experimentenergebnisse mit

$$\begin{aligned} \text{Kategorie 1: } & 0.9016, \\ \text{Kategorie 2: } & 1.289, \\ \text{Kategorie 3: } & 1.615, \end{aligned} \quad (2.7)$$

da diese Längen eine bessere Annäherung an Kurvenlängen bei digitalisierten Oberflächen bieten.

2.1.5 Skelett

Das Skelett ist eine dimensionale Reduktion eines Objektes. Im 3D-Raum ist ein Skelett $S^{3,2}$, welches um eine Dimension reduziert wurde, ein Flächenskelett. Ein um zwei Dimensionen reduziertes Skelett $S^{3,1}$ wird als Kurvenskelett bezeichnet. Die in diesem Abschnitt 2.1.5 beschriebenen Definitionen gelten für Kurvenskelette im dreidimensionalen diskreten Raum.

Endpunkte Alle Endpunkte sind dadurch definiert, dass genau ein Kategorie 1 adjazenter Skelettvoxel an einem Endpunkt angrenzt. Anders ausgedrückt befindet sich in dem $3 \times 3 \times 3$ Raster um einen betrachteten Skelettvoxel herum nur genau ein Kategorie 1 Adjazenzvoxel.

Knotenpunkte Knotenpunkte sind dadurch definiert, dass vom betrachteten Skelettvoxel aus mehr als zwei Pfade zu verschiedenen Endpunkten ausgehen. Knotenpunkte können durch die Definition von Endpunkten maximal vier ausgehende Skelettpfade haben, mindestens aber genau drei.

Zentriertheit Ein Kurvenskelett befindet sich im Mittelpunkt des Objektes. Diese Linie, auf der das Kurvenskelett liegt, wird oft als Centerline [HF05] oder mediale Achse [TS06] bezeichnet.

Dicke Das Kurvenskelett hat den Durchmesser von einem Voxel. Nach dieser Definition besteht das Kurvenskelett aus einer Aneinanderreihung von adjazenten Voxeln. Dabei existiert zwischen zwei beliebigen Punkten an jeder Stelle des Kurvenskelettpfades niemals die Möglichkeit, auf einem Pfad mehrere unterschiedliche Wege zu gehen.

Zusammenhängend Das Kurvenskelett ist immer zusammenhängend. Auf allen Pfaden eines Kurvenskeletts dürfen keine Abstände, also keine Skelettvoxel ohne Adjazenzvoxel im $3 \times 3 \times 3$ Raster existieren. Anders ausgedrückt hat jeder Skelettpunkt, der nicht End- oder Knotenpunkt ist, genau zwei adjazente Voxel im $3 \times 3 \times 3$ Nachbarschaftsraster.

2.1.6 Skelettierung

Skelettierung im Sinne der Bildverarbeitung ist die Reduktion eines Objektes in einem k -dimensionalen Raum um eine $k - f$ dimensionale Objektrepräsentation. Das bedeutet, dass in einem zweidimensionalen Raum ein Skelett auf eine eindimensionale Struktur transformiert wird. In diesem Fall werden Flächen des zweidimensionalen Raumes zu Kurven des eindimensionalen Raumes transformiert.

Ein dreidimensionales Objekt im Euklidischen Raum kann somit in ein 2D-Flächenobjekt reduziert werden. Bei der Skelettierung spricht man dann von einem Flächenskelett. Eine erneute Reduktion der Dimensionen bei der Skelettierung im 3D-Raum erzeugt dann ein sogenanntes 1D-Kurvenskelett. Dieses Kurvenskelett befindet sich nach wie vor im 3D-Raum, beschreibt das ursprüngliche Objekt jedoch ausschließlich durch Linien beziehungsweise Kurven.

Das 1D-Kurvenskelett ist dem in der Medizin definierten Skelett ähnlich, ist aber nicht identisch mit der medizinischen Definition, da sich das in der Medizin

und Zoologie definierte Skelett nicht zwingend in der Mitte des Körpers befinden muss.

2.2 Entstehung der 3D-Skelettierung

Die Ursprünge der Skelettierung in der Bildverarbeitung liegen in den späten 50er Jahren, als Harry Blum [Blu67] in seinen wissenschaftlichen Untersuchungen feststellte, dass es in der zwei Millenien alten Geschichte der Geometrie bisher keine konkreten mathematischen Ansätze für die Skelettierung zur Beschreibung geometrischer Formen gibt.

'A visual stimulus from which the contours or edges have been extracted, impinges on such a plane at some fixed time and excites the plane at those points.'

H. Blum beschreibt hier einen visuellen Stimulus, der sich an den Kanten und Ecken eines 2D-Objektes wellenartig in alle Richtungen fortbewegt; die Impuls- wellen sich jedoch gegenseitig nicht beeinträchtigen, beziehungsweise überlagern. Diese Ausbreitung des Stimulus ist in Abbildung 2.2 dargestellt, die Blum in seiner Arbeit nutzt, um den Stimulus und den daraus entstehenden Deskriptor zu beschreiben.

Er schlussfolgert bei konvexen Objekten, dass Wellen, die sich in das Innere des Objektes fortbewegen, immer an einem Mittelpunkt eines sogenannten maximal einfassenden Kreises aufeinander treffen. Dieser einfassende Kreis ist dadurch definiert, dass die Kontur des Kreises immer genau zwei Konturpunkte des Objektes berühren muss. Im Normalfall ist der durch Blum erstmals definierte Formdeskriptor ein Liniengebilde, welches in einem späteren Teil seiner Arbeit als mediale Achse und das Verfahren als die mediale Achsenfunktion (im Englischen: 'medial axis function', MAF) bezeichnet wird. Ein Spezialfall bildet der Kreis, dessen durch Blum definierter Formdeskriptor ein einziger Punkt ist. H. Blums Skelett S als Formdeskriptor eines k -Dimensionalen stetigen Objekts Ω hat immer $k - 1$ Dimensionen. Der Objektrand $\delta\Omega$ ist durch mindestens zwei Randpunkte \mathbf{a}, \mathbf{b} (Feature Punkte) des Objekts Ω definiert. Der Kreismittelpunkt \mathbf{p} ist durch den minimalen Abstand d_{min} der beiden Randpunkte \mathbf{a}, \mathbf{b} definiert.

Das Bloomsche Skelett ist mathematisch definiert durch:

$$S = S^{k,k-1}(\Omega) \{ \mathbf{p} \in \delta\Omega, \mathbf{a} \neq \mathbf{b}, \| \mathbf{p} - \mathbf{a} \| = \| \mathbf{p} - \mathbf{b} \| = d_p(\mathbf{p}) \}, \quad (2.8)$$

wobei $d_p(\mathbf{p})$ die Distanzfunktion ist, die zu jedem Skelettpunkt die minimale Distanz zum Objektrand $\delta\Omega$ bildet.

$$d_p(\mathbf{p}) = \min_{\mathbf{k} \in \delta\Omega} \| \mathbf{p} - \mathbf{k} \| \quad (2.9)$$

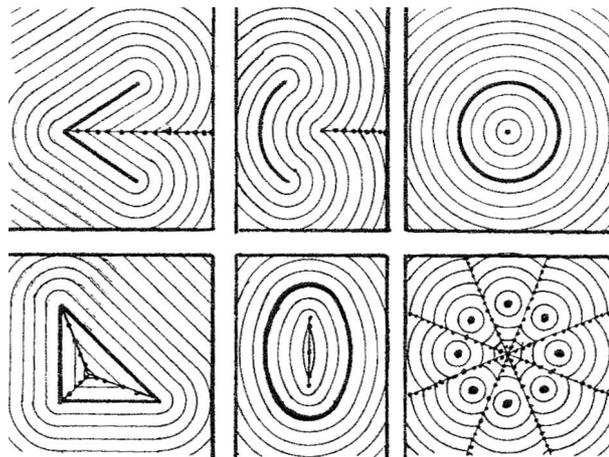


Abbildung 2.2: Grafische Beschreibung der Medialen Achse

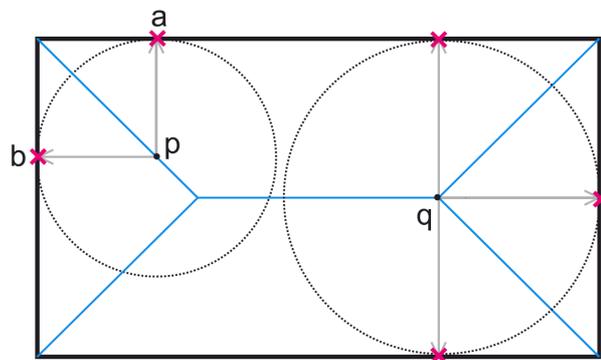


Abbildung 2.3: Grafische Beschreibung der Berechnung von Skeletten im Zweidimensionalen Raum

2.3 Skelettierungsansätze

Im Forschungsgebiet der Skelettierung im 2D- und 3D-Raum gibt es verschiedene Ansätze zur Berechnung von Kurvenskeletten. Dazu gehören im Wesentlichen die im Folgenden dargestellten Ansätze.

Zudem gibt es noch einige weitere Methoden wie die sogenannte Generalfeldmethode oder die "Methode der deformierbaren, sich kontinuierlich ausbreitenden Fronten" nach Sharf et al. [SLSK07].

2.3.1 Thinning

Thinning Algorithmen zur Berechnung von Kurvenskeletten bilden die am häufigsten vertretene Klasse von Methoden bei der Skelettierung von 3D-Objekten. Die ersten Thinning-Methoden entstanden zur Skelettierung von 2D-Skeletten und wurden schon in den späten 80ern auf 3D-Modelle übertragen. Eine gute Einführung in einfache 2D-Thinning-Methoden geben Lam et al. [LLS92] in der Veröffentlichung "Thinning Methodologies - A Comprehensive Survey".

Die zentrale Idee der Thinning Algorithmen ist ein iterationsweises Abtragen der Oberfläche, bis nur noch eine Linie des Objektes im Objektmittelpunkt bestehen bleibt. Moderne Thinning Algorithmen wie der 6-fache Subiterationen Algorithmus von Kálmán et al. [PK98] basieren dabei auf mehrfach iterativen Parallelitätsalgorithmen, die sicherstellen, dass die Objektoberfläche gleichmäßig abgetragen wird, sodass die Topologie des ursprünglichen Objektes im Kurvenskelett erhalten bleibt. Es werden bei Thinning Algorithmen immer Bedingungen an die Adjazenzvoxel (meist in Form der $3 \times 3 \times 3$ Nachbarschaft) eines betrachteten Voxels gestellt, sodass der betrachtete Voxel entfernt werden kann, falls alle aufgestellten Bedingungen erfüllt sind.

Die Thinning Algorithmen teilen sich wiederum in drei Untergruppen, die jeweils unterschiedliche Vorgehensweisen haben. Dazu gehört:

- Der "Border Sequential Algorithm", welcher für jeden Voxel die $3 \times 3 \times 3$ Nachbarschaft untersucht. Wenn die definierten Bedingungen für die Adjazenzvoxel im $3 \times 3 \times 3$ Raster erfüllt sind, wird der betrachtete Objektvoxel entfernt. Der Algorithmus untersucht in jeder Iteration jeden Voxel, bis kein Objektvoxel mehr nach der Definition abgetragen, bzw. entfernt werden muss. Ein Beispiel hierfür ist der Algorithmus von Lee et al. [LKC94], der unter anderem die Grundlage für ein im Framework ITK (Insight Toolkit [ITK10]) vorgeschlagenes, aber nie zertifiziertes Plugin von Homann [Hom07] bildet.
- Eine Algorithmengruppe, die durch eine größere Nachbarschaftsbetrachtung weniger oder keine Iterationen benötigt. Dazu gehört der Algorithmus von Ma et al. [MS96].
- Die Algorithmengruppe der "Subfield Sequential Method", bei der das Objekt in mehrere Teilbereiche unterteilt wird, welche sukzessiv aktiviert und bearbeitet werden. Ein Beispielalgorithmus wurde durch Bertrand und Aktouf [Ber94] beschrieben.

Die Abbildung 2.4 zeigt die Skelettierung eines 3D-Objektes der Thinning-Methode von Kálmán und Kuba [PK98].

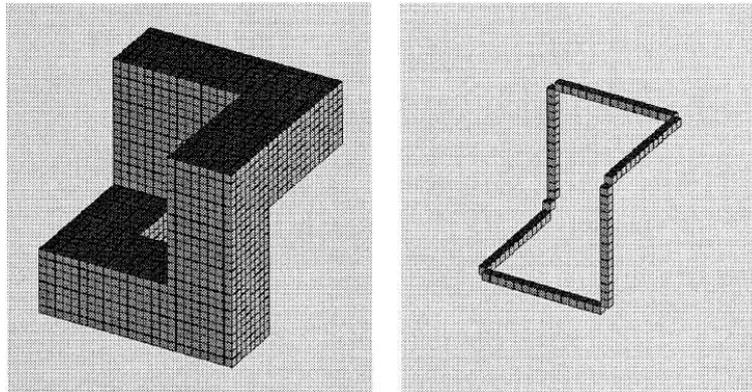


Abbildung 2.4: Thinning-Algorithmus nach Kálmán und Kuba

2.3.2 Voronoi-Diagramme

Voronoi-Diagramme bestehen zum einen aus Voronoi-Zentren. Diese Zentren sind bei der Skelettierung einzelne Samplingpunkte auf dem Objektrand. Des Weiteren haben Voronoi-Diagramme sogenannte Voronoi-Kanten, die mit den minimalen Euklidischen Distanzen zwischen den Voronoi-Zentren verlaufen.

Die Skelette sind demnach durch eine Teilmenge der Voronoi-Kanten definiert. Voronoi-Kanten, die durch den Objektrand hindurch außerhalb des Objektes verlaufen, werden verworfen. Die restlichen Kanten bilden das Kurvenskelett. Diese Vorgänge des Voronoi-Zentren Samplings und der Definition des Kurvenskelettes sind in Abbildung 2.5 dargestellt.

Das Problem bei der Berechnung von Skeletten auf Grundlage von Voronoi-Diagrammen ist, dass bei hohen Samplingraten auch sehr viele Voronoi-Kanten entfernt werden müssen. Zudem müssen zum Sampling und zur Auswahl der Voronoi-Zentren aufwendige Methoden definiert werden, die die richtige Auswahl von Samplingpunkten berechnen, um die topologische Struktur trotzdem korrekt auf das Kurvenskelett übertragen zu können.

Eine wichtige Grundlage für die Übertragung des Verfahrens der Berechnung von Kurvenskeletten im 2D- auf den 3D-Raum bieten Ogniewicz und Kübler [OK95] in ihrer Veröffentlichung "Hierarchic Voronoi skeletons". Bisher gibt es wenige Verfahren, die direkt aus Voronoi Diagrammen 3D-Kurvenskelette bilden können. Ein Ansatz, der skalierungsinvariant ist, stellen Cornea et. al [CSYB05] in der Veröffentlichung über die "Repulsive Force Methode" vor.

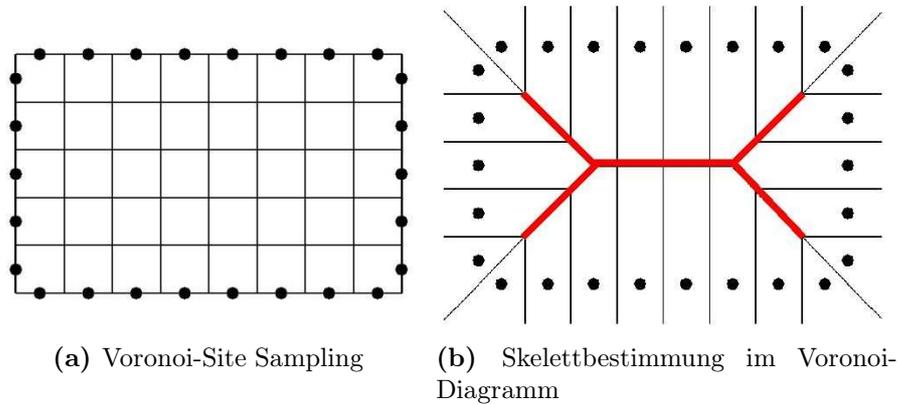


Abbildung 2.5: Voronoi-Diagramm und Bestimmung des Kurvenskeletts

2.3.3 Distanzfeld

Das Skelett ist in der Distanzfeldmethode durch die Maxima eines Vektor-Randfeldes vom Objektrand definiert. Das Randfeld wird durch einen Algorithmus der kürzesten Pfade, der von jedem Randpunkt des Objektes aus gestartet wird, berechnet. Die Singularitäten, die nicht differenzierbar sind und auf einer Höhenkarte die Gipfel bilden, sind Teile des Kurvenskeletts.

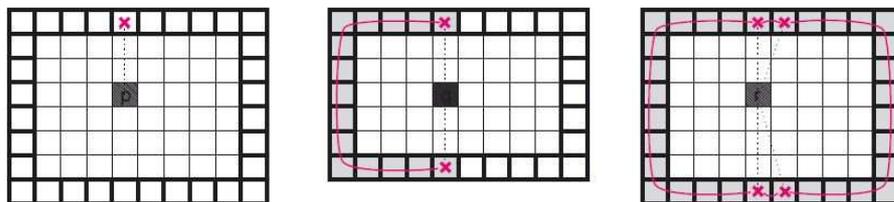


Abbildung 2.6: EFT bei der Bildung von Geodäten

Ein wichtiger Unterschied besteht dabei bei der Berechnung von kontinuierlichen und diskreten Objekten. Ge und Fitzpatrick [GF96] beschreiben die Probleme, die bei der Berechnung der Distanzfelder auftreten. Im diskreten Raum kann es passieren, dass auf Grund einer geraden Anzahl von Voxeln in Pfaden keine Mittelpunkte gefunden werden können.

Zu diesem Problem beschreibt Reniers [Ren08] eine Möglichkeit der erweiterten Feature Transformation (EFT), indem er für die Skelettierung sogenannte Jordan-Kurven mit Hilfe der berechneten EFT bildet. Eine detaillierte Betrachtung der

Vorgehensweise von Reniers ist in Kapitel 3 beschrieben. Ein Beispiel der erweiterten EFT ist in Abbildung 2.6 dargestellt. Das Hauptproblem der Skelettierung mit der Distanzfeld-Methode besteht darin, im diskreten Raum mit der gleichen Skelettierungsmethode stetig und zuverlässig sowohl zusammenhängende als auch dünne Kurvenskelette zu erhalten.

Kapitel 3

Eigener Lösungsansatz

3.1 Einführung

Reniers wendet in seiner Doktorarbeit [Ren08] eine Distanzfeld-Methode zur Skelettierung von 3D-Objekten an, die er in den darauf folgenden Kapiteln seiner Arbeit für die Segmentierung von 3D-Objekten nutzt. Diese Distanzfeld-Methode wird im folgenden Unterkapitel 3.2 im Detail analysiert und beschrieben.

In der Skeleton-Sandbox, einer eigens entwickelten C++ Software von Reniers, werden geladene 3D-Objekte von einer VTK (Visualization Toolkit) Polygon- in eine Voxelrepräsentation umgewandelt. Nach dieser Umwandlung werden im nächsten Schritt die Skelette der Voxelobjekte extrahiert. Die extrahierten Skelette dienen dann als Grundlage zur weiteren Verarbeitung in dem vorgestellten Ansatz in diesem Kapitel und den Experimenten in Kapitel 4.

Die Notwendigkeit der Weiterverarbeitung der Skelette von Reniers begründet sich darauf, dass Reniers in seinem Ansatz keine ein-Voxel-breiten Skelette extrahiert. Somit kann keine Klassifizierung einzelner wichtiger Skelettmerkmale wie nach der exakten Definition aus Kapitel 2.1.5 von Endpunkten, Knotenpunkten und kürzesten Distanzen erfolgen. Reniers erzeugt vielmehr teilweise mehrere Voxel-breite Skelette, aus denen keine direkte Definition für die markanten Skelettpunkte wie End- und Knotenpunkte gebildet werden kann.

In ein-Voxel-breiten Skeletten finden sich einfache und exakte Definitionen für markante Skelettpunkte, wie in Kapitel 2.1.5 beschrieben. Ein Skelett-Endpunkt ist dabei dadurch gekennzeichnet, dass zu ihm nur ein Kategorie 1 angrenzender adjazenter Voxel existiert. Einfache Skelettpunkte haben somit genau zwei adjazente Voxel und die Definition für Knotenpunkte ist, dass diese Voxel mindestens drei oder maximal vier adjazente Voxel in der direkten Nachbarschaftsumgebung besitzen.

Der im Folgenden vorgestellte Ansatz hat zum Ziel aus den vorliegenden mehreren Voxel breiten Skeletten die wichtigen Merkmale eines Skelettes zu extrahieren und vorhandene Merkmale aus dem Ansatz von Reniers zu nutzen, um ein Ähnlichkeitsmaß entwickeln zu können. Dazu gehört die Bestimmung von Endpunkten, Knotengebieten und den kürzesten Distanzen zwischen End- und Knotenpunkten innerhalb eines dreidimensionalen Skelettes.

3.2 Skelettierung nach Reniers

Die Skelettierungsmethode von Reniers beruht im Wesentlichen auf dem Distanzfeld-Ansatz, welcher in 2.3.3 beschrieben wird. Das Ziel seiner Arbeit ist aber nicht die Skelettierung, sondern die Segmentierung von 3D-Objekten auf Grundlage der Skelette. Skelette sind bereits eine Objektbeschreibung auf höchster Ebene und bieten eine geeignete Grundlage, um darauf aufsetzend die Segmentierung der Objekte durchzuführen.

Ein Problem der Kurvenskelett-Extraktion von Reniers ist, dass er in seiner Kurvenskelettdefinition nicht exakt ein-Voxel-breite Linien als Kurvenskelette festlegt, sondern für eine exakte Rekonstruktion und somit aufgrund der Topologieerhaltung des Objektes eine mehrere Voxel-breite Skelettlinie berechnet. Diese Diskrepanz zwischen den nach Reniers berechneten mehreren Voxel-breiten Kurvenskeletten und der Kurvenskelettdefinition mit ein-Voxel-breiten Kurvenskeletten wird mit den in dieser Arbeit vorgestellten Ansätzen entgegen gewirkt.

Reniers Algorithmus basiert auf der Distanzfeldmethode zur Skelettierung. Er besteht dabei aus drei Hauptschritten:

1. Berechnung der (erweiterten) Feature-Transformation
2. Berechnung der kürzesten Geodäten
3. Berechnung des Kurvenskeletts

Die Berechnung der toleranzbasierten Feature Transformation (TFT) ist ein nicht triviales Problem im diskreten Raum und wird u.a. in einer Veröffentlichung von Reniers und Telea [RT06] genau analysiert. Bei der in dem vorgestellten Skelettierungsalgorithmus angewendeten TFT werden zu jedem Punkt \mathbf{p} innerhalb des 3D-Objekts Ω die Featurepunkte auf dem Außenrand $\delta\Omega$ berechnet.

Die Featurepunkte werden mit einer beim Skelettierungsprozess einstellbaren Toleranzbasis berechnet. In Abbildung 3.1 wird verdeutlicht, wie sich der Toleranzparameter auf die Featureberechnung auswirkt. Im diskreten Raum \mathbb{Z}^3 wird die

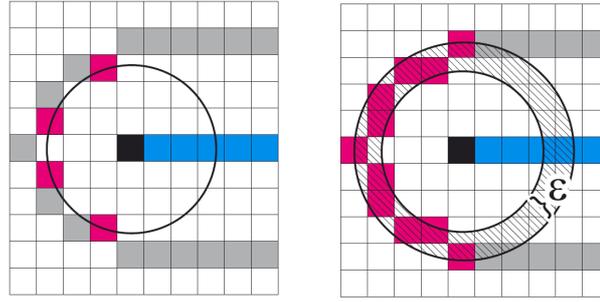


Abbildung 3.1: Toleranzbasierte Feature Transformation

toleranzbasierte Featureberechnung deshalb benötigt, weil beispielsweise bei gerader Voxelanzahl in einer beliebigen Dimension an einem betrachteten Skelettpunkt keine zwei Featurepunkte berechnet werden können, da keine zwei gleich langen Euklidischen Distanzen vom Skelettpunkt zu den potentiellen Featurepunkten existieren. Die Berechnung der kürzesten Geodäten zwischen den Featurepunkten wird mit dem Dijkstra Algorithmus realisiert und ist der zeitintensivste Teil des Algorithmus von Reniers. Der Dijkstra Algorithmus ist ein Algorithmus aus der Klasse der Greedy-Algorithmen und berechnet in diesem Anwendungsfall die kürzesten Abstände zu jedem Objekttrandvoxel, bis ein weiterer Featurepunkt gefunden wurde.

Existieren mehr als zwei Geodäten zwischen den berechneten Featurepunkten, so bilden diese Geodäten eine sogenannte Jordankurve. Eine Jordankurve nach Lawson [Law06] ist eine einfache, zusammenhängende Kurve, die ein Objekt in zwei voneinander getrennte Segmente, beispielsweise Regionen, unterteilt.

Die formale Definition, dass der untersuchte Punkt $\mathbf{p} \in \Omega$ ein Kurvenskelettpunkt des Skeletts S ist, lässt sich durch:

$$\mathbf{p} \in S \Leftrightarrow |\Gamma| \geq 2 \quad (3.1)$$

ausdrücken, wobei Γ die Menge aller gefundenen Geodäten zwischen den Featurepunkten $\mathbf{f}_{1p}, \mathbf{f}_{2p}, \dots, \mathbf{f}_{n_p}$ von \mathbf{p} bildet. Der Skelettierungsalgorithmus von Reniers ist im Folgenden als Pseudocode dargestellt:

- 1: compute feature transform F on Ω
- 2: **for** each object voxel $\mathbf{p} \in \Omega$ **do**
- 3: $\bar{F} \leftarrow \bigcup_{x,y,z \in \{0,1\}} F(p_x + x, p_y + y, p_z + z)$
- 4: $\Gamma \leftarrow \bigcup_{\mathbf{a} \neq \mathbf{b} \in \bar{F}} \text{shortestpath}(\mathbf{a}, \mathbf{b})$
- 5: **if** Γ contains a Jordan curve **then**
- 6: $S \leftarrow S \cup \{\mathbf{p}\}$

```

7:   else
8:     surface skeleton or non-skeleton voxel
9:   end if
10: end for

```

In **Zeile 1 und 3** wird die Feature-Transformation einfach und toleranzbasiert erweitert und für jeden Punkt \mathbf{p} innerhalb des Objekts Ω berechnet. In **Zeile 3** wird die Menge \overline{F} aller Features, die durch die TFT mit den Toleranzwerten $x, y, z \in \{0, 1\}$ berechnet wurden, gespeichert.

In **Zeile 4** wird die Menge Γ von Geodäten, die zwischen der Menge \overline{F} der errechneten Featurepunkte liegen, berechnet.

In **Zeile 5-9** wird aus der Menge Γ der gefundenen Geodäten untersucht, ob die Geodäten eine Jordankurve bilden. Falls dies der Fall ist, ist der aktuell untersuchte Punkt \mathbf{p} ein Kurvenskelettpunkt und wird dem Kurvenskelett

$$C \leftarrow C \cup \mathbf{p} \quad (3.2)$$

zugeordnet.

Konkret stellt sich im Algorithmus die Berechnung der Jordankurve aufwendiger dar. So werden zuerst die Geodäten, also die potentiellen Jordankurven, gebildet. Daraufhin werden diese Kurven mit einem Toleranzwert, der sogenannten dilation distance D_d , auf dem Objektrand $\delta\Omega$ ausgeweitet. Als einen in seinen Untersuchungen und Experimenten sehr oft ausreichend hohen D_d -Wert gibt Reniers den Wert von 5 (Voxeln) an.

Nachdem die Jordankurven zu Jordanbändern ausgeweitet wurden, wird eine Randextraktion der Jordanbänder vorgenommen. Die Randextraktion bewirkt eine Segmentierung der Jordanbänder. Auf Grund der Segmentierung kann algorithmisch bestimmt werden, ob der untersuchte Punkt ein Kurvenskelettpunkt ist. Bedingung ist, dass mindestens zwei oder mehr Segmente, das heißt voneinander getrennte extrahierte Jordanband-Bereiche, existieren müssen (farblich hervorgehobene Kurven im rechten Teil der Abbildung 3.2). Existieren drei oder mehr Segmente, so lässt sich dabei auf einen Knotenpunkt schließen. Hier ist deutlich die Analogie der Blumschen Skelettdefinition, die in Kapitel 2.2 aufgeführt wurde, zu erkennen. Im dreidimensionalen Raum lässt sich dabei die Blumsche Definition statt mit maximalen Kreisen mit maximal einfassenden Kugeln erweitern.

Die von Reniers stammende Abbildung 3.2 veranschaulicht den Prozess des Bildens, Ausweitens und der Randextraktion der Jordankurven. Reniers errechnet mit

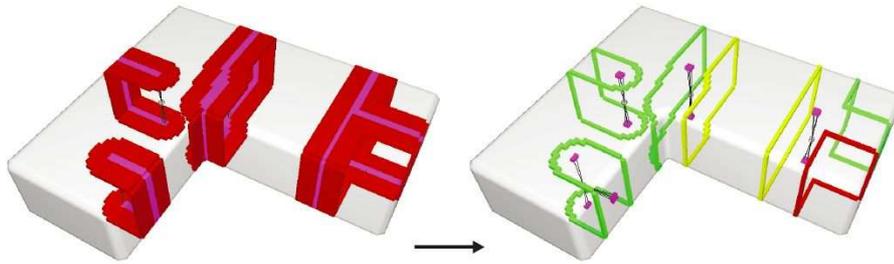


Abbildung 3.2: Berechnung, Ausweitung und Randextraktion von Jordankurven

seiner aus mehreren Verfahren gebündelten Distanzfeldmethode zusammenhängende Kurvenskelette, die sich jedoch zur Kurvenskelettdefinition aus Kapitel 2.1.5 in einigen grundlegenden Dingen unterscheidet. Folgende Feststellungen gelten nach der Berechnung der Kurvenskelette nach Reniers DF-Methode:

1. Die Skelettlinie ist nicht dünn, also nicht genau ein Voxel breit.
2. Die Skelettlinie ist nicht zusammenhängend, da auch indirekt adjazente Voxel bei Reniers als zusammenhängend definiert sind.
3. Knotenpunkte sind auf Grund der TFT nur ungenau definiert und dadurch unpräzise auf dem Skelett markiert.
4. Endpunkte entsprechen auf Grund der TFT nicht mehr der Definition aus Kapitel 2.1.5.
5. Die Distanzen zwischen Knoten- und Endpunkten sind nicht eindeutig definiert (aufgrund von Feststellung 1).

Diese starken Abweichungen von der Skelettdefinition aus Kapitel 2.1.5 verhindern eine direkte Berechnung der Merkmalsvektoren für Skelette, die wiederum die Basis für Vergleiche zwischen Ground Truth mit menschlicher Ähnlichkeitsbestimmung und einer skelettbasierten Ähnlichkeitsbestimmung sind.

Nach der Skelettdefinition aus Kapitel 2.1.5 lassen sich die wichtigen topologischen Maße zur Merkmalsvektorberechnung wie Endpunkte, Knotenpunkte, Distanzen zwischen End- und Knotenpunkten und die Anzahl von ausgehenden Pfaden an Knotenpunkten einfach berechnen.

Im Folgenden wird deshalb eine Methode vorgestellt wie Skelette, die durch die Distanzfeld-Methode berechnet werden, weiter verändert werden müssen, um die

für einen einfachen topologisch orientierten Ähnlichkeitsvergleich und der Berechnung von Merkmalsvektoren erforderlichen Voraussetzungen zu schaffen. Es wird dabei ausschließlich auf die Methode von Reniers eingegangen und es verbleibt die Notwendigkeit, die neu entwickelte Methode auf weitere zukünftig entwickelte Skelettierungsverfahren, die ebenfalls mit der Distanzfeldmethode arbeiten, zu testen und gegebenenfalls anzupassen.

3.3 Modifikationen der Skelette nach DF-Methode

3.3.1 Topologisch orientiertes Ähnlichkeitsmaß

Den Überlegungen und der Umsetzung zur Veränderung des berechneten Skeletts durch Reniers geht die Definition eines geeigneten Merkmalvektors für Ähnlichkeitsvergleiche von Objekten auf Basis der Kurvenskelette voraus. Der Merkmalvektor ist somit eine wichtige Grundlage zur Veränderung der Skelettstrukturen. Auch wenn andere Ähnlichkeitsmaße definiert werden, kann der nachfolgende Algorithmus zur Strukturveränderung des Skeletts eingesetzt werden, jedoch können bei der angewandten Methode schon während der Veränderung der Skelettstruktur wichtige Bestandteile des Merkmalvektors performant parallel dazu berechnet werden.

Der vorgestellte Merkmalvektor hat das Ziel einer soliden topologischen und kompakten Repräsentation des Objekts durch fünf Merkmale, die anhand des Skeletts nach der Skelettdefinition aus Kapitel 2.1.5 berechnet werden können.

$$\mathbf{f}_v = (m_1, m_2, m_3, m_4, m_5)^T \quad (3.3)$$

Merkmal 1: Anzahl m_1 aller Endpunkte $\mathbf{e}_0, \mathbf{e}_1, \dots, \mathbf{e}_n$. Diese Summe der Endpunkte spiegelt die Komplexität des Objektes wieder.

Merkmal 2: Anzahl m_2 aller Knotenpunkte $\mathbf{n}_0, \mathbf{n}_1, \dots, \mathbf{n}_n$. Diese Summe ergänzt den Komplexitätswert aus Merkmal 1 und lässt ebenso Rückschlüsse auf die Komplexität des Objektes schließen.

Merkmal 3: Mittlere Anzahl der ausgehenden Pfade von Knotenpunkten. Dieser Durchschnittswert zeigt die Verteilung von Endpunkten auf die Knotenpunkte. Je höher dieser Wert, desto mehr Endpunkte knüpfen im Durchschnitt an den selben Knotenpunkt an.

$$m_3 = \frac{\sum_{i=0}^{m_2} \sum_{j=0}^{m_1} c(\mathbf{n}_i, \mathbf{e}_j)}{m_2} \quad (3.4)$$

Merkmal 4: Mittlere Distanz aller kürzesten Pfade N_p von End- zu Knotenpunkten. Dieser Mittelwert komprimiert die Längen aller Objektsegmente in einen Wert.

$$m_4 = \frac{\sum_{i=0}^e \sum_{j=0}^{\lceil e/2 \rceil} d(\mathbf{e}_i, \mathbf{e}_j)}{N_p} \quad (3.5)$$

Merkmal 5: Varianz der mittleren Distanz aller kürzesten Pfade $P_{d_0}, P_{d_1}, \dots, P_{d_n}$ von End- zu Knotenpunkten. Dieses Merkmal spiegelt die Gleichmäßigkeit der Länge von verschiedenen Objektsegmenten wieder.

$$m_5 = \frac{\sum_{i=0}^{N_p} (P_{d_i} - m_4)^2}{N_p - 1} \quad (3.6)$$

3.3.2 Vorarbeit

Das nach der Distanzfeld-Methode berechnete Skelett entspricht nicht der Kurvenskelettdefinition aus Kapitel 2.1.5. Um den in Kapitel definierten Merkmalsvektor berechnen zu können, müssen jedoch als minimale Anforderung End- und Knotenpunkte auf dem Skelett definiert sein.

Das Ziel einen eigenen Algorithmus zur Neuberechnung von Skeletten zu entwerfen entsteht erst aus der Notwendigkeit heraus, die spezifischen Skelettpunkte wie End- und Knotenpunkte berechnen zu müssen. Bisher existieren nur wenige zuverlässige und implementierte Skelettierungsverfahren für 3D-Objekte. Umso wichtiger ist es, den bisherigen Ansatz und die Umsetzung von Reniers weiterzuentwickeln.

Reniers setzt in seiner Methode die Knotenpunkte an die korrekten Positionen innerhalb der Skelettlinien. Es existieren für den neuen und erweiterten Skelettierungsalgorithmus als Eingabeobjekte Skelette, die durch Voxel mit jeweils einem Skalarwert beschrieben sind. Das Eingabeobjekt ist als dreidimensionaler Würfel zu verstehen, in dem die Skelettvoxel mit dem Skalarwert 2 gespeichert sind. Die Knotenpunkte der Skelette sind gewöhnlich mit dem Skalarwert 3 oder 4 gespeichert. Alle anderen Voxel, die nicht zum Skelett gehören, haben den Skalarwert 0.

In Abbildung 3.3 wird ein selbst erzeugtes Beispielskelett nach Reniers Algorithmus dargestellt. Die Skalarwerte sind von Reniers so gewählt, dass die Skalare eine direkte Aussagekraft über die Adjazenz- bzw. Nachbarvoxel haben. Der Skalar 2 bedeutet, dass der Skelettvoxel i.d.R. zwei Adjazenzvoxel hat. Eine Ausnahme

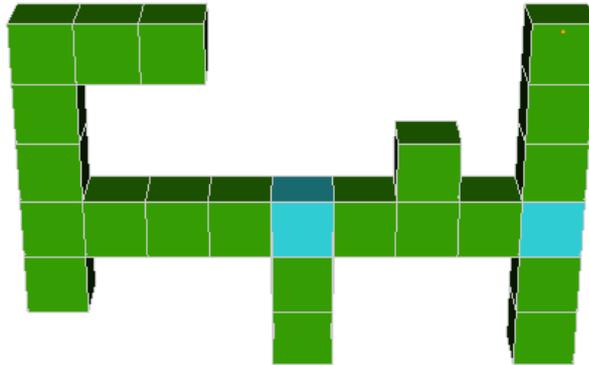


Abbildung 3.3: Beispielskelett nach Reniers Kuvenskelettdefinition

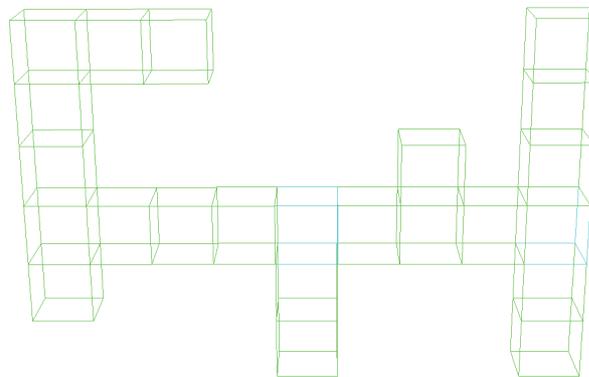


Abbildung 3.4: Beispielskelett in der eigenen VTK-Voxelengine

bilden die Skelettendvoxel, die ebenfalls den Skalar 2 haben und nicht von normalen Skelettvoxeln unterscheidbar sind. Skelettknotenvoxel, die drei Skelettpfade miteinander verbinden, haben den Skalarwert 3, während Skelettknotenvoxel, die vier Skelettpfade miteinander verbinden, den Skalar 4 haben.

Die nachfolgend beschriebenen Algorithmen und Programmteile wurden in Java 1.6 geschrieben und beinhalten das Visualization Toolkit Framework für Java [VTK09]. Mit dem VTK-Framework als Grundlage wurde eine VTK basierte Voxelengine programmiert, die Veränderungen an den Skeletten direkt durch OpenGL visualisiert. Durch die Voxelengine, die Skelette als interaktive 3D-Voxel-Wireframes anzeigt, kann auf einfachem Weg überprüft werden, wie sich Veränderungen der angewandten Algorithmen auf die Skelette auswirken. Beim ersten Ansatz wurde versucht durch die morphologischen Operatoren Dilatation und an-

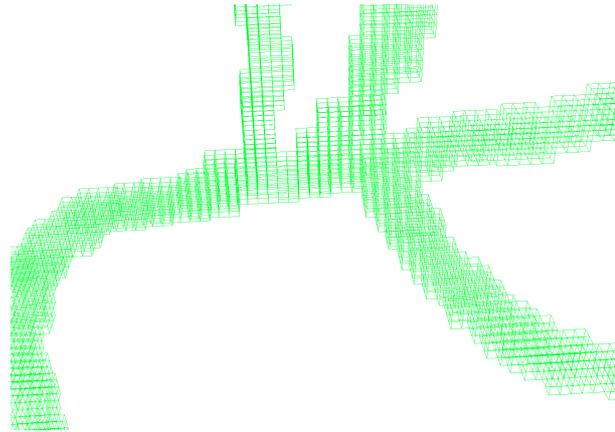


Abbildung 3.5: Ausschnitt eines dilatatierten Kuvenskeletts

schließende Erosion eine verbesserte Skelettstruktur zu erhalten. Bei der Dilatation (lat. für Erweiterung) wird das Skelett S durch ein strukturierendes Element W ausgedehnt. Als strukturierendes Element wurde ein $3 \times 3 \times 3$ Voxel-Würfel genutzt. In einer Iteration wird jeder Skelettvoxel als Mittelpunkt dieses Würfels betrachtet und alle Voxel im $3 \times 3 \times 3$ Würfel werden als Skelettpunkte definiert. Per Definition nach Hermes [Her04] wird dieses Einfügen des Strukturelementes in jeden Skelettvoxel als Operation

$$A = S \cup W \quad (3.7)$$

dargestellt. Die Abbildung 3.5 soll die Wirkung der Dilatation verständlicher machen. Nach der Dilatation erfolgt die Erosion, bei der jeder (auch dilatatierte) Skelettvoxel S_d mit einer Strukturmaske W verglichen wird. Die Strukturmaske stellt in diesem Experiment wieder ein $3 \times 3 \times 3$ Voxelwürfel dar. Bei der Erosion wird geprüft, ob das strukturierende Element vollständig in die Menge der Skelettvoxel passt. Ist das der Fall, bleibt der untersuchte Skelettvoxel in der Menge der erodierten Skelettvoxel erhalten. Die folgende Definition nach Hermes [Her04] lautet daher

$$A = S \setminus W. \quad (3.8)$$

Die Ergebnisse der Dilatation und der anschließenden Erosion fielen positiv aus, so wurden durchschnittlich zwischen 60-70% der falschen Skelettendvoxel eliminiert. Jedoch bleiben manche der falschen Endpunkte erhalten. Zudem verändert sich die Struktur des Skeletts, sodass sich Distanzen von End- zu Knotenpunkten verändern und die Berechnung der Merkmalsvektoren, in die diese Distanzen mit eingerechnet werden, ebenso betroffen sind.

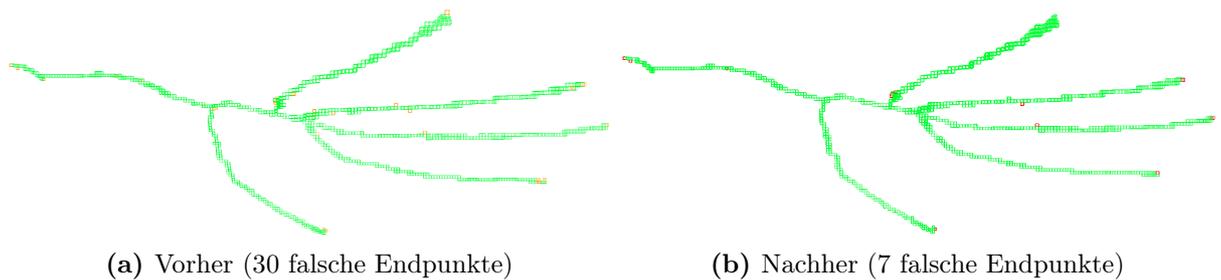


Abbildung 3.6: Anwendung morphologischer Operatoren

In Abbildung 3.6 ist das Kurvenskelett einer menschlichen Hand vor und nach dem Anwenden der beschriebenen morphologischen Operatoren abgebildet. Die grün gefärbten Punkte sind normale Skelettpunkte. Die rot eingefärbten Voxel sind die per Definition erhaltenen Endpunkte des Skeletts.

3.3.3 Dijkstra-Skeleton Algorithmus

Der finale Ansatz zur Eliminierung der nach der Skelettdefinition falschen Endpunkte, der Ermittlung der Knotenpunkte, sowie Berechnung der Merkmalsvektoren hat einen stark modifizierten Dijkstra-Algorithmus als Grundlage.

Zunächst werden die Ziele des Algorithmus definiert:

1. Falsche Endpunkte müssen vollständig eliminiert werden.
2. Richtige Endpunkte müssen erhalten bleiben.
3. Distanzen von End- zu Knotenpunkten müssen definiert und berechnet werden.

Der entwickelte Algorithmus erfüllt diese Ziele und stellt eine Kombination aus Zielerreichung und gleichzeitiger Berechnung aller zur Merkmalsvektorberechnung erforderlichen Maße dar. So ist der mehrstufige Dijkstra-Skeleton-Algorithmus (im Folgenden durch 'DSA' abgekürzt) in der Lage, parallel und vollständig Merkmalsvektoren zu berechnen.

Der DSA besteht aus mehreren Berechnungsschritten, die im Folgenden aufgezählt sind:

1. Transformation des Skeletts in einen zusammenhängenden Graphen
2. Ausdehnung der Knoten in Knotengebiete

3. Berechnung der kürzesten Pfade zwischen allen Endpunkten im Graph nach Dijkstras Algorithmus
 - (a) Speichern des ersten durchquerten Knotengebiets
4. Dijkstra-Pfade zwischen Endpunkten auf zwei Knock-out Kriterien untersuchen:
 - (a) 1. Ausschlusskriterium: Endpunkt liegt zwischen 2 Knotengebieten
 - (b) 2. Ausschlusskriterium: Mehrere Endpunkte haben gemeinsames erstes Knotengebiet und betrachteter Endpunkt hat nicht die längste Distanz zum Knotengebiet
 - (c) Berechnung der fehlenden Werte des Merkmalsvektors
 - (a) Berechnung der Summe aller ausgehenden Pfade pro Knotengebiet
 - (b) Berechnung der durchschnittlichen Distanz aller kürzesten Pfade zwischen Endknoten zu Knotengebieten
 - (c) Berechnung der Varianz der mittleren Distanz
 - (d) Normalisierung
 - (e) Berechnung und Speicherung des Merkmalsvektors

Bei der Graphentransformation in **Schritt 1** findet die Umwandlung der Voxel mit deren Skalarwerten in eine Adjazenzmatrix statt. In der Adjazenzmatrix werden Entfernungen zwischen den einzelnen Graphenknoten gespeichert. Die Graphenknoten werden wiederum als einzelne Objekte (DijkstraNode) angelegt, die notwendige Informationen zur Lage im Raum als auch Informationen über Nachbarknoten enthalten.

Da im darauf folgenden Teil nicht mehr das Skelett in Voxelrepräsentation sondern als Skelettgraph betrachtet wird, werden einzelne Skelettvoxel als (Graph-) Knoten bezeichnet und die Skelettknotenvoxel als Skelettknoten beschrieben.

In **Schritt 2** werden die Skelettknoten zu Skelettknotengebieten (im Folgenden durch 'SKG' abgekürzt) ausgeweitet. Die Notwendigkeit Skelettknoten zu SKG zu erweitern besteht darin, dass kürzeste Pfade zwischen Endpunkten immer das erste SKG speichern müssen. Ist das Skelett an der Skelettknotenstelle aber breiter als ein Voxel, so können die kürzesten Pfade zwischen Endpunkten auch an einem Skelettknoten vorbeigehen, ohne diesen zu berühren. Um das zu vermeiden, werden die SKG berechnet. Es wird ein $3 \times 3 \times 3$ Würfel als strukturierendes Element eingesetzt. Jeder gefundene Nachbarknoten innerhalb des Strukturelements wird

demnach als Teil des SKG markiert. SKG werden als einzelne Objekte (Nodearea) angelegt. Alle SKG werden zudem in einem globalen Objekt (AllNodeAreas) gespeichert, um u.a. eine Überprüfung zu gewährleisten, ob Knoten, bzw. Skelettknoten bereits zu einem SKG gehören.

Im **Schritt 3** werden alle kürzesten Pfade und deren Distanzen von allen End- zu Endpunkten berechnet. Dies geschieht durch einen modifizierten Dijkstra-Algorithmus, der beim Berechnen der kürzesten Pfade das jeweils erste durchquerte Knotengebiet speichert.

Im **4. Schritt** werden die gespeicherten Endknoten und Skelettknoten auf die beiden Ausschlusskriterien in 4 (a) und (b) untersucht, in dem zunächst die Knoten aus der Liste der Endknoten entfernt werden, die auf ihren kürzesten Pfaden zwei oder mehrere unterschiedliche Knotengebiete durchqueren. Danach werden alle Gruppen von Endknoten betrachtet, die das gleiche Knotengebiet durchqueren. In einer solchen Endknotengruppe werden für jeden Endknoten alle kürzesten Distanzen zu Endknoten außerhalb der Gruppe aufsummiert und innerhalb der Gruppe verglichen. Es werden daraufhin alle Endknoten, außer der Knoten mit der höchsten Summe der kürzesten Distanzen, in der Gruppe eliminiert. So bleibt der Endknoten mit der größten Distanz zum Knotengebiet erhalten.

Im **Schritt 5** werden die verbleibenden Teile des Merkmalsvektors berechnet, normalisiert und gespeichert. Danach liegen folgende Informationen vor, aus denen sich der Merkmalsvektor zusammensetzt:

1. Anzahl aller echten Endknoten
2. Anzahl aller Knoten (durch Knotengebiete repräsentiert)
3. Mittlere Anzahl ausgehender Pfade von Knotengebieten
4. Mittlere Distanz zwischen echten Endknoten zu Knotengebieten
5. Varianz der mittleren Distanz von echten Endknoten zu Knotengebieten

3.3.4 Beispiel einer Kurvenskelettumwandlung mit dem DSA

Um den Grundgedanken des Algorithmus verständlicher zu machen, ist in Abbildung 3.7 eine konzeptionelle Darstellung eines Skelettgraphen abgebildet. Skelettknoten werden als gelbe Quadrate und Endpunkte als schwarze Pfeile dargestellt. Die Endpunkte 2 und 3 (e_1, e_2) müssen laut Definition entfernt werden. Dabei werden die beiden Ausschlusskriterien an dem folgenden Beispiel sichtbar. Nachdem alle kürzesten Pfade $g_{e_1, e_2}, g_{e_1, e_3}, g_{e_1, e_4}, \dots, g_{e_5, e_4}$ durch den Dijkstra-

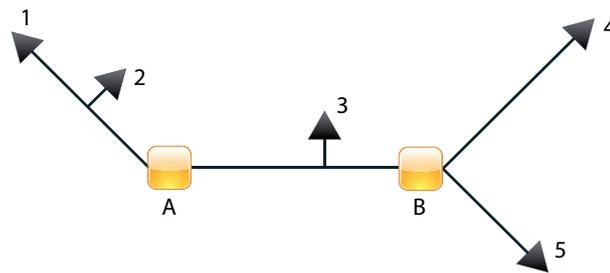


Abbildung 3.7: Konzeptionelle Darstellung eines Skelettgraphen

Algorithmus gebildet wurden, wird im Teil 4 (a) des Algorithmus festgestellt, dass die Endpunkte e_1, e_2 laut Daten, die in Schritt 3 (a) berechnet wurden, das gleiche Knotengebiet A durchquert haben, aber die kürzesten Pfade von e_1 nach e_2 und vice versa kein Knotengebiet durchquert haben. Dies führt zur Berechnung der Summe aller Distanzen $\sum_{i=0}^v \sum_{j=0}^w d(e_v, e_w)$ der berechneten kürzesten Pfade zu allen anderen Endpunkten e_3, e_4, e_5 . In dem in Abbildung 3.7 dargestellten Graph werden die Summen der Distanzen

$$\sum_{i=3}^5 d(e_1, e_i) > \sum_{j=3}^5 d(e_2, e_j) \quad (3.9)$$

von e_1 und e_2 miteinander verglichen. Die Summe der Distanzen aller Pfade ist in e_1 größer, und so wird e_2 aus der Liste der Endpunkte entfernt.

Nach der Kombination aller kürzesten Distanzen von allen End- zu Endpunkten wird bei e_3 festgestellt, dass mehrere Knotengebiete durchquert wurden, somit wird auch der Endpunkt e_3 aus der Liste der Endpunkte entfernt.

Es entsteht somit ein Kurvenskelett nach der Skelettdefinition aus Kapitel 2.1.5. Mit dieser Skelettrepräsentation können die benötigten Skalare $m_1 - m_5$ für den Merkmalsvektor f_v berechnet werden.

3.3.5 Programmstruktur

Die Programmstruktur ist in Form von UML Klassendiagrammen in den Abbildungen 3.8 und 3.9 abgebildet und gibt einen Überblick über die Programmklassen, Funktionen und Funktionssignaturen des Java-Programms. Die wichtigsten Funktionen wurden im Kapitel 3.3.3 erläutert.

3.4 Ähnlichkeitsbestimmung

Im Kapitel 3.4 wird die algorithmische Methode der Ähnlichkeitsbestimmung zwischen Objekten dargestellt. Zusätzlich werden Vergleiche zwischen menschlicher Ähnlichkeitsbestimmung der Ground Truth und algorithmischer Ähnlichkeitsberechnung nach dem DSA anhand standardisierter Methoden aus dem Information Retrieval durchgeführt.

3.4.1 Algorithmische Ähnlichkeitsberechnung

Anhand der in Unterkapitel 3.3.3 berechneten Merkmalsvektoren $\mathbf{f}_{v_1}, \mathbf{f}_{v_2}, \dots, \mathbf{f}_{v_n}$ können die Ähnlichkeiten der Objekte anhand dieser Vektoren berechnet werden. Dazu wird das Winkelmaß des Kosinus zwischen zwei Vektoren genutzt, das durch

$$\cos(\mathbf{f}_{v_a}, \mathbf{f}_{v_b}) = \frac{\sum_{u=1}^s \mathbf{f}_{v_a,u} \mathbf{f}_{v_b,u}}{\sqrt{\sum_{u=1}^s \mathbf{f}_{v_a,u}^2} \sqrt{\sum_{u=1}^s \mathbf{f}_{v_b,u}^2}} \quad (3.10)$$

definiert ist [Sch05]. Wobei s die Anzahl der Elemente der Vektoren ist. Bei diesem Ähnlichkeitsmaß spielen die Längen der Vektoren keine Rolle, da die beiden Wurzeln im Nenner die Euklidischen Distanzen der beiden Vektoren sind und die Vektoren dadurch normiert werden. So kann das Kosinusmaß auch als Skalarprodukt beider normalisierter Vektoren beschrieben werden.

Auf diesem Ähnlichkeitsmaß auf Grundlage der aus den Skeletten berechneten Merkmalsvektoren und den menschlich bewerteten Ähnlichkeiten zwischen Objekten aus der Ground Truth werden im Folgenden bekannte Verfahren aus dem Information Retrieval genutzt, um das Ähnlichkeitsmaß mit den menschlich wahrgenommenen Ähnlichkeiten zu vergleichen.

3.4.2 Ähnlichkeitssuche in Objektdatenbanken

Nach Stock [Sto06] ist das Ziel des Information Retrieval, alle relevanten Objekte zur Deckung des Informationsbedarfs zu finden, und zwar ausschließlich die Relevanten. Die Qualität von Treffermengen auf eine Anfrage geben dabei vor allem die zwei Kennzahlen 'Recall' und 'Precision'. Um diese Kennzahlen zu beschreiben, definiert Stock zuerst folgende drei Variable:

- a - Anzahl der gefundenen relevanten Treffer
- b - Anzahl der nichtrelevanten Objekte in der Treffermenge (Ballast)

- c - Anzahl der relevanten Objekte, die nicht in Treffermenge sind (Verlust)

So lässt sich der Recall als Wert der Vollständigkeit durch

$$\text{Recall} = a/(a + c) \quad (3.11)$$

beschreiben. Der Wert Recall beschreibt das Verhältnis von gefundenen relevanten Objekten zu der Gesamtzahl der relevanten Dokumente, also der Vollständigkeit der gefundenen Objekte. Die Precision als Wert der Genauigkeit berechnet sich durch

$$\text{Precision} = a/(a + b). \quad (3.12)$$

Sie beschreibt demnach das Verhältnis von gefundenen relevanten Objekten im Verhältnis zur gesamten Treffermenge.

So können im Information Retrieval sowohl für Recall und Precision die Werte 1 erreicht werden. Das Bestreben im Information Retrieval beim konkreten Informationsbedarf nach [Sto06] geht bei beiden dieser Werte gegen 1, jedoch sollten die Werte nicht 1 sein. Ist der Recall 1 (Precision \ll 1), so sind eventuell zu viele Objekte in der Treffermenge und die Precision sinkt. Ist die Precision bei 1 (Recall \ll 1), so sind eventuell zu wenige, aber dafür ausschließlich relevante Objekte in der Treffermenge. So ist festzustellen, dass es in vielen Anwendungen des Information Retrieval ein weiteres Ziel ist, Recall und Precision in ähnlicher Größenordnung zu erhalten.

3.4.3 Anfrage-Definition

Im Information Retrieval wird eine Anfrage an das System geformt. Die Ergebnis- oder Treffermenge, wird dann anhand verschiedener Kennzahlen, wie hier dem Recall und Precision bewertet.

Die Anfrage an das System (Programm zur Berechnung der Featurevektoren und Ähnlichkeitsmaß) bilden die in Kapitel 4 ausgewählten und beschriebenen Objekte. Die Werte Recall und Precision werden auf Basis der optimalen Treffermenge aus der Ground Truth 'GT' berechnet. Ein Beispiel für die Berechnung zeigt die Tabelle 3.1. An diesem Beispiel ist zu sehen, wie sich Recall und Precision auf verschiedene Treffermengen der Ground Truth verhalten.

Die Gesamtliste der Ähnlichkeiten der Ground Truth, aus der die Trefferliste H_G berechnet wird, bildet die Grundlage zur Berechnung von Recall und Precision. Die Treffermenge H_G wird durch einen heuristischen Ansatz aus der Gesamtliste der Ähnlichkeiten ermittelt. Dabei werden die aus der Ground Truth ermittelten Ähnlichkeiten $s(O_v)$ aller anderen Objekte zum Anfrage-Objekt O_a absteigend

Anfrageobjekt	Treffermenge GT	Treffermenge Ähnlichkeitsmaß	Recall	Precision
O_a	O_x, O_y, O_z	O_x, O_y, O_z	1	1
O_b	O_i, O_j, O_k, O_l	O_c, O_d	0	0
O_c	O_i, O_j, O_k, O_l	O_i, O_j	0.5	1
O_d	O_i, O_j	O_i, O_j, O_k, O_l	1	0.5
O_e	O_i, O_j, O_k, O_l	O_i, O_j, O_m, O_n	0.5	0.5

Tabelle 3.1: Berechnung von Recall und Precision

sortiert. Im darauf folgenden Berechnungsschritt werden in der sortierten Liste alle Ähnlichkeitsunterschiede $\Delta(d_{O_v}, d_{O_{v+1}})$ gebildet.

Es hat sich durch Versuche aus Kombination unterschiedlicher Anfrage-Objekte herausgestellt, dass die Treffermenge der Ground Truth gut gewählt ist und vergleichbar gute Recall- und Precisionwerte erzielt werden können, wenn nur Objekte vor dem viertgrößten Ähnlichkeitsunterschied mit in die Treffermenge einbezogen werden. Dazu soll die abgebildete Tabelle 3.2 verdeutlichen, wie aus der Gesamtliste von Ähnlichkeiten der Ground Truth die Treffermenge H_G eines Anfrageobjekts berechnet wird.

Gesamtliste	$s(O_h)$	$\Delta(d_{O_v}, d_{O_{v+1}})$	Δ -Position	$O_v \in H_G$
O_i	0.9	-	-	✓
O_j	0.65	0.25	1	✓
O_k	0.64	0.01	7	✓
O_l	0.45	0.2	3	✓
O_m	0.3	0.15	4	x
O_n	0.06	0.24	2	x
O_o	0.02	0.04	5	x
O_p	0.0	0.02	6	x

Tabelle 3.2: Beispielberechnung Treffermenge H_G von Anfrageobjekt O_h

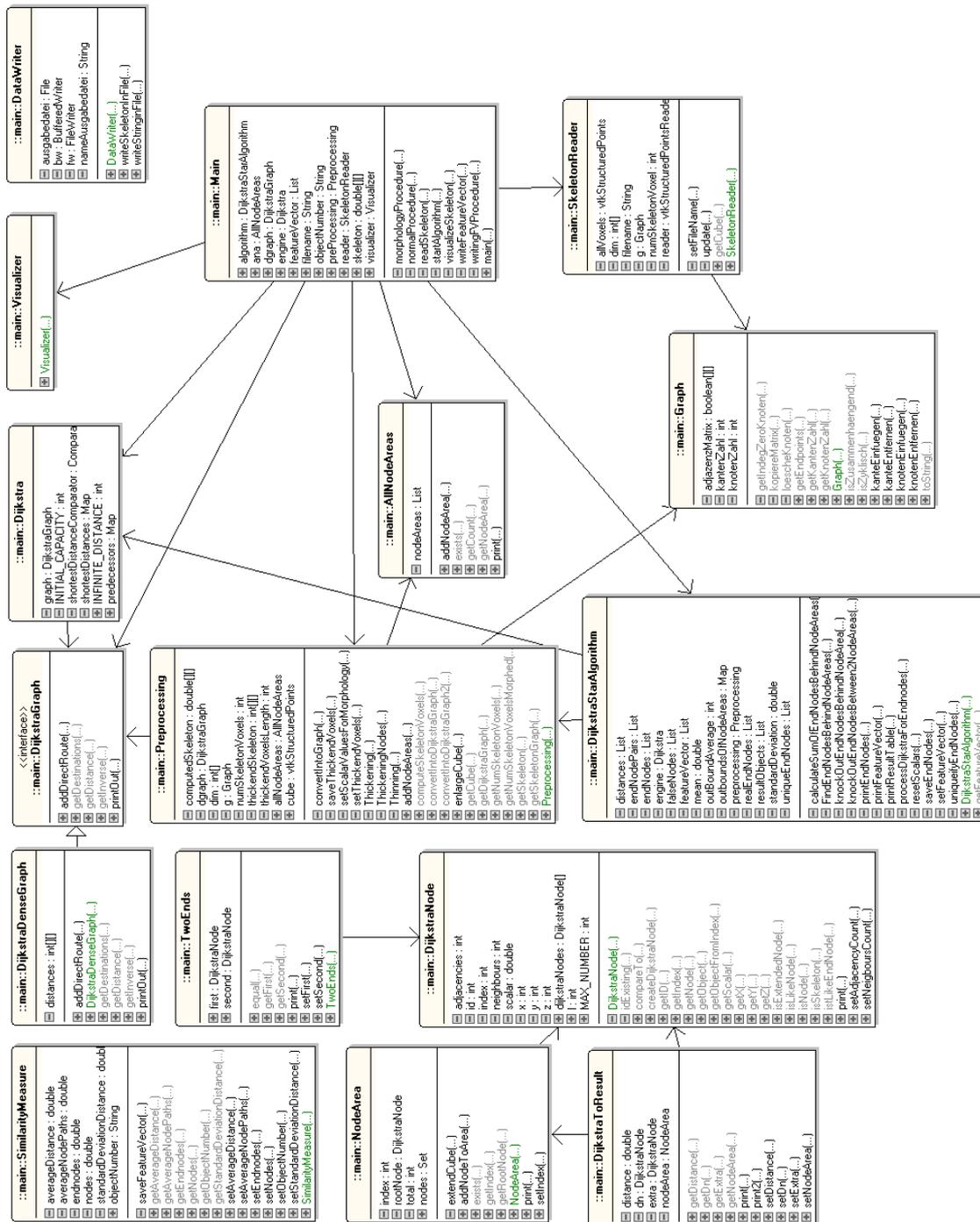


Abbildung 3.8: Klassendiagramm 'Main Package' zur Berechnung von Merkmalsvektoren

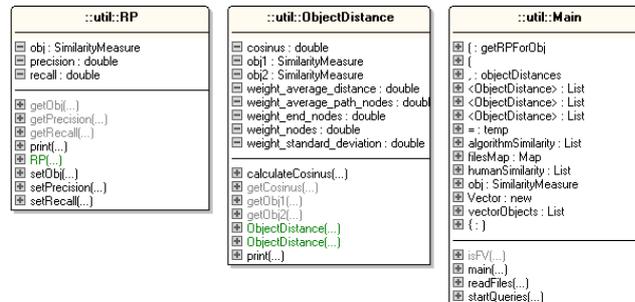


Abbildung 3.9: Klassendiagramm 'Util Package' zur Berechnung der Recall / Precision

Kapitel 4

Experimente und Ergebnisse

4.1 Einführung

In diesem Kapitel werden Experimente mit allen Objekten aus der Ground Truth durchgeführt und die erhaltenen Ergebnisse ausgewertet und visuell dargestellt. Es werden alle in Kapitel 3 beschriebenen Ansätze umgesetzt und die Ergebnisse im letzten Teil dieses Kapitels diskutiert.

Die Ground Truth beinhaltet 14 Objekte, die durch menschliche Einschätzung der Ähnlichkeit der Objekte zueinander bewertet wurden. Dabei wurden folgende Hinweise an die bewertenden Personen gegeben:

1. Ähnlichkeiten werden auf der Skala zwischen 0.00 - 1.00 bewertet
2. Identische Objekte erhalten den Ähnlichkeitswert 1.00 (Beispiel: Menschliche Objekte mit unterschiedlichen Stellungen der Gliedmaßen sind identisch)
3. Vorliegende Abbildungen sind als vollständige 3D-Objekte zu betrachten
4. Wichtige Eigenschaften bei der Ähnlichkeitsbestimmung
 - (a) Geometrie (Formen)
 - (b) Anzahl und Größe von Segmenten, Gliedmaßen
 - (c) Oberflächeneigenschaften wie bspw. Oberflächenverformungen

Die Objekte wurden speziell so ausgewählt, dass sowohl ähnliche Paare, als auch Serien von möglichen ähnlichen Objekten, aber auch einzelne sehr unterschiedliche Objekte in die Ground Truth mit aufgenommen wurden. Tabelle 4.1 beschreibt die Auswahl der Objekte textuell und die Abbildung 4.1 stellt die tatsächlichen Objekte dar.

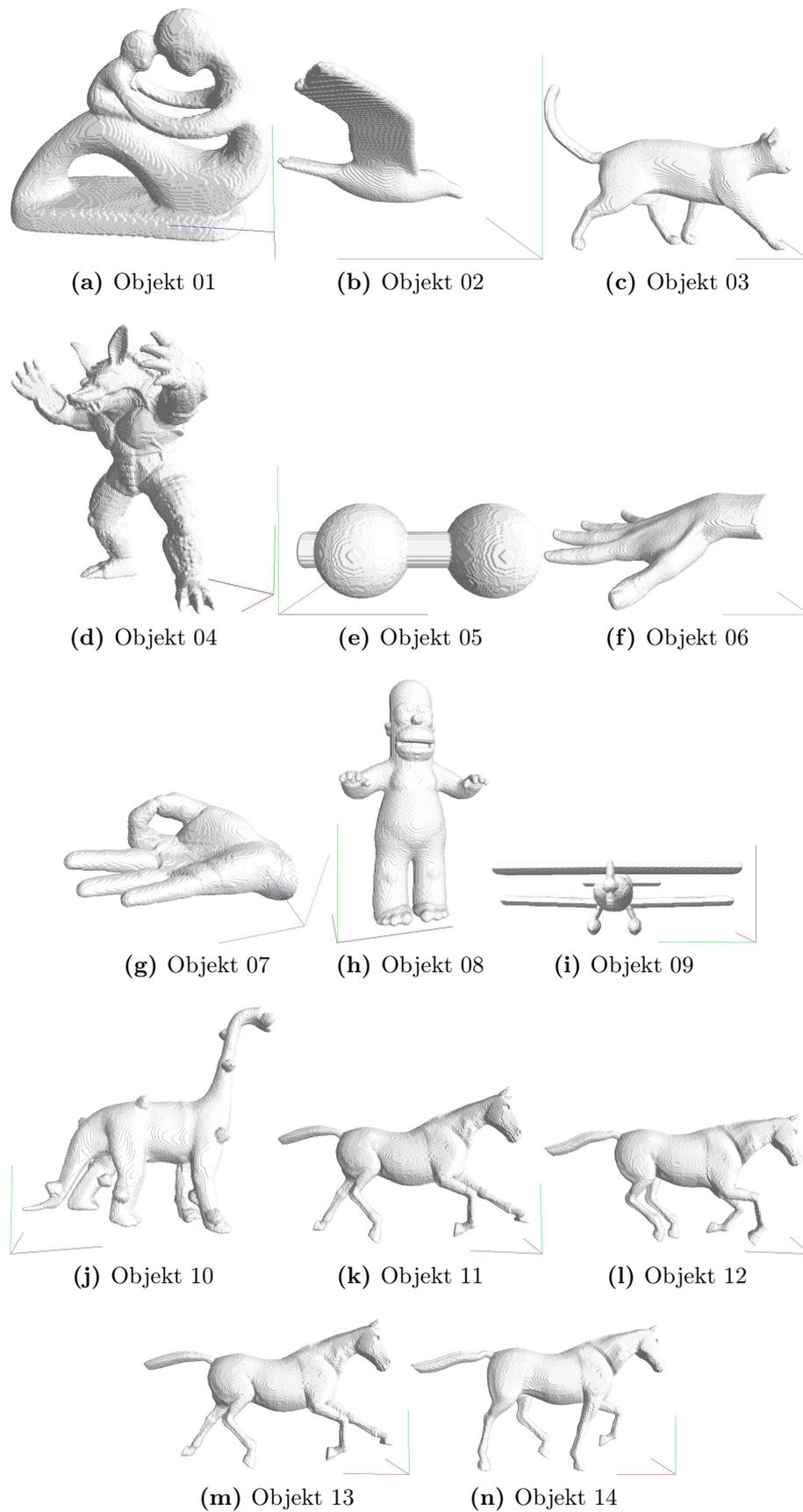


Abbildung 4.1: Objekte aus der Ground Truth

Objektnummer	Objektbeschreibung
O_1	Mutter-Kind-Figur
O_2	Vogel
O_3	Katze
O_4	Armadillo
O_5	Hantel
O_6	Hand ausgesreckt
O_7	Hand Geste
O_8	Homer
O_9	Flugzeug
O_{10}	Kranker Dinosaurier
O_{11}	Pferd 1
O_{12}	Pferd 2
O_{13}	Pferd 3
O_{14}	Pferd 4

Tabelle 4.1: Objektbeschreibungen und Zuordnung

4.2 Experimentenmethodik

Die Erstellung einer Ground Truth, auf deutsch etwa ‘manuell klassifizierte Referenzbilddaten’, ist bei Ähnlichkeitsanalysen und -vergleichen in der Bildverarbeitung ein notwendiges Mittel, um ein neu entwickeltes Ähnlichkeitsmaß einer vorangegangenen menschlichen Ähnlichkeitsanalyse gegenüberzustellen. Durch eine Gegenüberstellung im Kontext der Ähnlichkeitsanalyse zwischen 3D-Modellen können erste, wenn auch nur bedingt präzise Aussagen über die aus dem Verfahren ermittelte und die menschlich ermittelte Ähnlichkeit verschiedener Objekte oder Objektklassen getroffen werden.

Bedingt präzise sind die menschlichen Ähnlichkeitsbewertungen deshalb, da die menschliche Ähnlichkeitswahrnehmung meist nur in zwei Kategorien ‘ähnlich’ und ‘nicht ähnlich’ funktioniert. Zwischen diesen beiden Ausprägungen gibt es keine rationalen Abgrenzungen. In der Statistik bedeutet dies, dass der Mensch die Objekte ordinal skaliert bewertet. Die Ordinalskala hat in dem Fall zwei Ausprägungen mit einem gegenseitigen Verhältnis von ‘ähnlich’ $>$ ‘nicht ähnlich’. Hingegen beruht die vom Algorithmus berechnete Ähnlichkeit auf einer stetig metrischen Intervallskala, in der beliebig viele Ausprägungen bestehen und die Verhältnisse zwischen allen Ausprägungen exakt auf die Art der Berechnung des Ähnlichkeitsmaßes zurückführbar sind. Ähnlichkeitsmaße, die auf unterschiedlichen Skalen beruhen, sind nur schwer und begrenzt miteinander vergleichbar. So können in diesem Fall

Aussagen über 'ähnlich' und 'nicht ähnlich' gemacht werden, jedoch können beide Ähnlichkeitsmaße nicht direkt und absolut miteinander verglichen werden.

Der Vergleich ist vielmehr ausschlaggebend für eine erste Bewertung des neu entwickelten Ähnlichkeitsverfahrens anhand des Vergleichs mit Ähnlichkeitswerten aus menschlichen Klassifikation. Der Vergleich kann wie schon zuvor beschrieben, nur eine grobe Richtung für das weitere Vorgehen geben, da zudem noch menschliche Fehler bei der Ähnlichkeitsanalyse wie Ungenauigkeit oder Unwissenheit über den Sachverhalt und Sachzusammenhänge eine genaue und absolute Ähnlichkeitsanalyse mit den schon unterschiedlichen Skalen der Ähnlichkeitsmaße nicht zulassen.

Die Ground Truth der 14 ausgewählten Objekte wurde durch Ähnlichkeitsanalysen von insgesamt 15 Studenten aus den Fachbereichen Informatik, Informationsmanagement, Computervisualistik und Psychologie gebildet, in dem die Durchschnittswerte der insgesamt 91 Ähnlichkeitswerte berechnet wurden. In Tabelle 4.2 sind die durchschnittlichen Ähnlichkeitswerte abgebildet.

	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8	o_9	o_{10}	o_{11}	o_{12}	o_{13}	o_{14}
o_1	-													
o_2	0,10	-												
o_3	0,12	0,20	-											
o_4	0,20	0,07	0,27	-										
o_5	0,12	0,20	0,13	0,05	-									
o_6	0,13	0,12	0,15	0,13	0,17	-								
o_7	0,13	0,11	0,12	0,10	0,17	0,90	-							
o_8	0,19	0,07	0,40	0,47	0,10	0,20	0,13	-						
o_9	0,08	0,46	0,07	0,07	0,09	0,07	0,10	0,06	-					
o_{10}	0,22	0,07	0,38	0,30	0,07	0,10	0,17	0,17	0,07	-				
o_{11}	0,13	0,10	0,61	0,23	0,07	0,07	0,10	0,19	0,07	0,60	-			
o_{12}	0,13	0,10	0,60	0,23	0,10	0,07	0,10	0,19	0,07	0,59	0,95	-		
o_{13}	0,12	0,10	0,63	0,23	0,10	0,07	0,10	0,19	0,07	0,59	0,95	0,92	-	
o_{14}	0,13	0,10	0,60	0,23	0,10	0,07	0,10	0,19	0,07	0,61	0,96	0,96	0,95	-

Tabelle 4.2: Ähnlichkeitswerte aus der Ground Truth

Um die menschliche Ähnlichkeitsanalyse besser bewerten zu können, wurden die Ähnlichkeitswerte bis 0.10 blau gefärbt ('ähnlich') und die Ähnlichkeitswerte ab 0.90 rot gefärbt ('unähnlich'). Sehr auffällig sind die hohen Ähnlichkeitsbewertungen bei der Objektgruppe $O_{11} - O_{14}$. Diese Objektgruppe besteht aus den unterschiedlichen Pferdedarstellungen. Hohe Ähnlichkeitswerte über 0.90 sind hier also begründet und nachvollziehbar.

In ähnlichem Maße hoch wurde die Objektgruppe O_6, O_7 mit einer durchschnittlichen Ähnlichkeit von 0.9 bewertet. Da beide Objekte Hände mit unterschiedlichen Gesten abbilden, ist die menschliche Ähnlichkeitsbewertung auch in diesem Fall begründet und nachvollziehbar. Ebenfalls ist aus der Ähnlichkeitstabelle 4.2 zu entnehmen, dass beide Objekte O_6, O_7 zu Objekt O_5 mit einer höheren Ähnlichkeit bewertet wurden, als zu den übrigen Objekten. Aus rationaler Sichtweise heraus ist diese höher eingeschätzte Ähnlichkeit nicht begründet. Die Objekte unterscheiden sich in ihrer geometrischen Form stark voneinander. Die höher bewertete Ähnlichkeit kann aber aus dem Sachzusammenhang des bewertenden Menschen hergeleitet werden. So weiß der Bewertende, dass eine Hantel (O_5) mit der Hand (O_6, O_7) benutzt wird und stuft die Ähnlichkeit deshalb höher ein als bei anderen Elementen, bei denen er keinen solch starken und direkten Sachzusammenhang erkennt.

4.3 Merkmalsvektor- und Ähnlichkeitsberechnungen

Für alle Objekte O_1, O_2, \dots, O_{14} wurden die Merkmalsvektoren durch den DSA, der in Kapitel 3.3 vorgestellt wurde, berechnet. Die berechneten Merkmalsvektoren $\mathbf{f}_{v_1} = (s_{v_1}, s_{v_2}, s_{v_3}, s_{v_4}, s_{v_5})^T$ werden in der Tabelle 4.3 vorgestellt.

Objekt	s_{v_1}	s_{v_2}	s_{v_3}	s_{v_4}	s_{v_5}
O_1	25.00	21.00	1.00	16.00	13.90
O_2	5.00	2.00	2.00	58.00	43.80
O_3	11.00	11.00	1.00	60.00	54.40
O_4	6.00	6.00	2.00	2.00	1.10
O_5	5.00	11.00	1.00	9.00	7.60
O_6	6.00	5.00	1.00	60.00	38.80
O_7	5.00	9.00	1.00	61.00	59.00
O_8	18.00	26.00	1.00	9.00	11.10
O_9	18.00	22.00	1.00	37.00	49.40
O_{10}	30.00	49.00	1.00	20.00	36.90
O_{11}	34.00	49.00	1.00	19.00	36.60
O_{12}	25.00	39.00	1.00	26.00	45.30
O_{13}	26.00	27.00	1.00	15.00	14.80
O_{14}	14.00	15.00	1.00	16.00	8.60

Tabelle 4.3: Berechnete Merkmalsvektoren

Nach der Berechnung der Merkmalsvektoren bilden diese die Grundlage zur Ähnlichkeitsberechnung durch das Kosinusmaß. Die Tabelle 4.4 zeigt die Ähnlichkeitswerte nach der Berechnung des Vektor-Kosinusmaßes. Es lassen sich wie bei den Ähnlichkeitswerten der Ground Truth deutlich hohe Ähnlichkeitswerte bei den Objekten 11-14 ($O_{11} - O_{14}$) feststellen. Es wurden dazu alle Werte ab 0.997 rot eingefärbt. Auch die Zweiergruppe der Hände (O_6, O_7) weist eine hohe Ähnlichkeit von 0.995 auf.

4.4 Information Retrieval mit Recall- und Precisionergebnissen

Um die beiden Ähnlichkeitsmaße geeignet vergleichen zu können, wurde auf bekannte Verfahren im Information Retrieval zurückgegriffen. Die aus der Ground Truth geformte Treffermenge ergibt ein optimales Ergebnis anhand dessen die Treffermenge des DSA gemessen werden kann.

Es zeigen sich in vielen Anfragen eindeutig positive Ergebnisse, in denen sowohl der Recall- als auch der Precisionwert in einem Bereich zwischen 0.20 - 0.70 liegen. Gerade bei den Anfragen mit den Objekten $O_1, O_3, O_4, O_6, O_7, O_8$ zeichnen sich sowohl hohe Recall- als auch Precisionwerte ab. Bei den Treffermengen dieser Objekte, die aus dem Vergleich des DSA mit der Ground Truth entstehen, können starke Parallelen zwischen dem menschlichen und dem algorithmischen Ähnlichkeitsmaß abgeleitet werden.

Wiederum sind die Ergebnisse von Recall und Precision bei der Objektgruppe der Pferde ($O_{11} - O_{14}$) nicht so präzise wie erwartet, sondern es ist zu sehen, dass zwar alle anderen Pferdeobjekte pro Anfrage geliefert werden, sich jedoch viele falsche Objekte in der Treffermenge befinden.

Das Diagramm in Abbildung 4.2 zeigt die Recall-Precisionverteilung aller Anfragen mit der in Kapitel 3.4.3 definierten heuristischen Annahme. Alle Objektanfragen, sowie Treffermengen und die daraus berechneten Recall- und Precisionergebnisse sind in Tabelle 4.5 aufgelistet.

4.5 Diskussion

Es lassen sich durch die in dieser Arbeit angewandten Methoden Vergleiche der menschlichen und algorithmischen Ähnlichkeiten durchführen. Unterstützende Verfahren und Kennzahlen des Information Retrieval helfen bei der Analyse und lassen weitere Rückschlüsse beim Ähnlichkeitsvergleich zu.

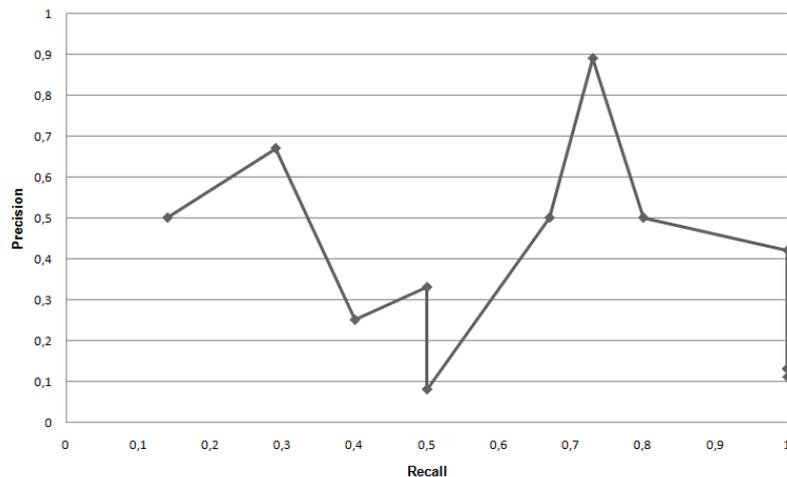


Abbildung 4.2: Recall / Precision im RP-Diagramm

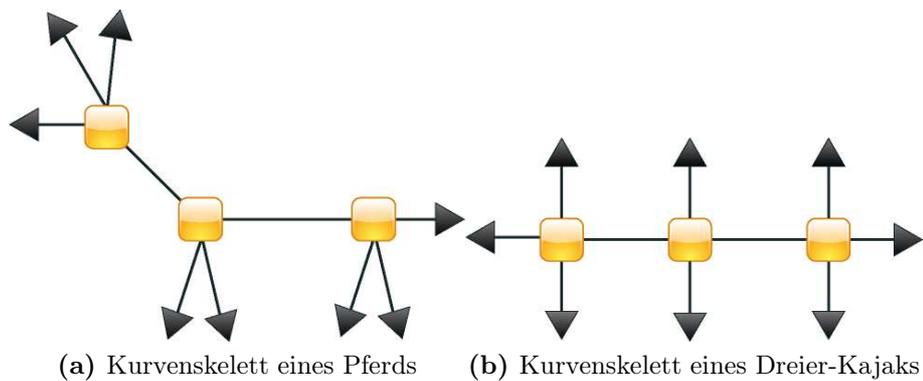


Abbildung 4.3: Topologisch dargestellte Kurvenskelette

Es konnte bestätigt werden, dass die angewandten Methoden der Berechnung von Merkmalsvektoren, die auf 3D-Kurvenskeletten basieren, mit Einschränkungen als Ähnlichkeitsmaß bei Ähnlichkeitsvergleichen genutzt werden können. Einschränkungen bestehen deshalb, da unterschiedliche Objekte sehr ähnliche topologische Strukturen besitzen können. Haben zwei unterschiedliche Objekte sehr ähnliche topologische Strukturen, so werden durch die topologischen Merkmalsvektoren hohe Ähnlichkeitswerte berechnet. Deshalb sollte die Ähnlichkeitsbestimmung durch die vorhandenen Merkmalsvektoren durch weitere Verfahren ergänzt werden. Ergänzende Maßnahmen und existierende Ähnlichkeitsvergleiche auf Grundlage von Kurvenskeletten werden im Kapitel 5 vorgestellt.

In der folgenden Abbildung 4.3 sind die konzeptionellen Skelette eines Pferdes und eines Dreier-Kajaks dargestellt. Zusätzlich sind topologisch wichtige Eigen-

schaften wie End- und Knotenpunkte, sowie Abstände zwischen diesen Punkten dargestellt. Die beiden Skelette sind topologisch fast identisch und ergeben bei dem Ähnlichkeitsvergleich mit den in dieser Arbeit vorgestellten Methoden sehr große Ähnlichkeitswerte. Im Vergleich dazu wird bei einer menschlichen Ähnlichkeitsanalyse mit recht großer Sicherheit keine Ähnlichkeit zwischen diesen beiden Objekten festgestellt werden.

In Tabelle 4.3 ist zu sehen, dass sich einige Skalare der Merkmalsvektoren nicht signifikant voneinander unterscheiden. Es lässt sich darauf schließen, dass eine Gewichtung der einzelnen Skalare der Merkmalsvektoren sinnvoll sein kann. Möglichkeiten zur optimalen Gewichtung bieten neuronale Netze, auf die im folgenden Kapitel als weitere Verbesserung der hier vorgestellten Ähnlichkeitsberechnung eingegangen wird.

	o_1	o_2	o_3	o_4	o_5	o_6	o_7	o_8	o_9	o_{10}	o_{11}	o_{12}	o_{13}	o_{14}
o_1	-													
o_2	0,704	-												
o_3	0,966	0,789	-											
o_4	0,981	0,682	0,946	-										
o_5	0,476	0,699	0,401	0,511	-									
o_6	0,928	0,811	0,991	0,896	0,349	-								
o_7	0,944	0,798	0,992	0,900	0,351	0,994	-							
o_8	0,980	0,644	0,922	0,995	0,522	0,863	0,876	-						
o_9	0,987	0,693	0,954	0,997	0,506	0,912	0,914	0,898	-					
o_{10}	0,982	0,670	0,939	0,999	0,515	0,887	0,894	0,998	0,995	-				
o_{11}	0,988	0,749	0,989	0,980	0,450	0,961	0,967	0,968	0,982	0,978	-			
o_{12}	0,989	0,683	0,952	0,994	0,496	0,902	0,915	0,995	0,989	0,996	0,986	-		
o_{13}	0,986	0,679	0,948	0,995	0,499	0,897	0,908	0,996	0,989	0,997	0,984	0,999	-	
o_{14}	0,992	0,710	0,969	0,991	0,479	0,923	0,938	0,988	0,989	0,991	0,995	0,999	0,997	-

Tabelle 4.4: Ähnlichkeiten auf Grundlage der Merkmalsvektoren nach dem Kosinusmaß

Anfrage-Objekt	Treffermenge GT	Treffermenge Ähnlichkeitsmaß	Recall	Precision
O_1	$O_2, O_3, O_4, O_5, O_6, O_8, O_{10}, O_{11}, O_{12}, O_{13}, O_{14}$	$O_3, O_4, O_8, O_9, O_{10}, O_{11}, O_{12}, O_{13}, O_{14}$	0.73	0.89
O_2	O_3, O_5, O_6, O_7, O_9	$O_1, O_3, O_4, O_5, O_6, O_7, O_9, O_{10}, O_{11}, O_{12}, O_{13}, O_{14}$	1.00	0.42
O_3	O_{11}, O_{13}	O_7, O_9, O_{11}	0.50	0.33
O_4	$O_1, O_3, O_8, O_{10}, O_{11}$	$O_1, O_8, O_9, O_{10}, O_{11}, O_{12}, O_{13}, O_{14}$	0.80	0.50
O_5	O_6, O_7	$O_1, O_2, O_3, O_4, O_7, O_8, O_9, O_{10}, O_{11}, O_{12}, O_{13}, O_{14}$	0.50	0.08
O_6	$O_1, O_2, O_3, O_5, O_7, O_8, O_{10}$	O_3, O_7, O_{11}	0.29	0.67
O_7	$O_1, O_2, O_5, O_6, O_8, O_9, O_{10}$	O_3, O_6	0.14	0.50
O_8	$O_1, O_3, O_4, O_6, O_{10}, O_{11}$	$O_1, O_4, O_9, O_{10}, O_{11}, O_{12}, O_{13}, O_{14}$	0.67	0.50
O_9	$O_2, O_5, O_7, O_8, O_{11}$	$O_1, O_4, O_8, O_{10}, O_{11}, O_{12}, O_{13}, O_{14}$	0.40	0.25
O_{10}	O_{11}	$O_1, O_4, O_8, O_9, O_{11}, O_{12}, O_{13}, O_{14}$	1.00	0.13
O_{11}	O_{13}	$O_1, O_3, O_4, O_9, O_{11}, O_{12}, O_{13}, O_{14}$	1.00	0.13
O_{12}	O_{14}	$O_1, O_4, O_8, O_9, O_{10}, O_{11}, O_{13}, O_{14}$	1.00	0.13
O_{13}	O_{11}	$O_1, O_4, O_8, O_9, O_{10}, O_{11}, O_{12}, O_{14}$	1.00	0.13
O_{14}	O_{12}	$O_1, O_3, O_4, O_8, O_9, O_{10}, O_{11}, O_{12}, O_{13}$	1.00	0.11

Tabelle 4.5: Anfrage- sowie Recall und Precision Ergebnisse

Kapitel 5

Zusammenfassung und Ausblick

5.1 Zusammenfassung

Die aufgenommene Ground Truth zeigt eine recht gute Annäherung von Ähnlichkeitsbewertungen, die durch die Probanden abgeschätzt wurden. Es lassen sich allerdings bei den Ähnlichkeitswerten in der Ground Truth die Ähnlichkeitsbereiche zwischen 'ähnlich' und 'unähnlich' nicht definieren, sodass eine eher duale Ordinalskala zum Einsatz kommt. Hingegen werden die Ähnlichkeiten bei der Ähnlichkeitsberechnung des angewandten Algorithmus auf einer stetig metrischen Intervallskala gemessen.

Aus dieser Tatsache heraus werden die Ähnlichkeitsmaße über die Definition von Anfragen und Treffermengen aus dem Information Retrieval miteinander verglichen. Die Recall- und Precisionergebnisse lassen auf eine gute Annäherung zwischen menschlichem und algorithmischem Ähnlichkeitsvergleich schließen.

Ein wichtiges Anwendungsgebiet, in das der in dieser Arbeit vorgestellte Algorithmus angewendet werden kann, ist die Nachbearbeitung zur exakten Erstellung von Kurvenskeletten (nach der Kurvenskelettdefinition in Kapitel 2.1.5), die durch das Distanztransformationsverfahren (siehe Kapitel 2.3.3) berechnet wurden.

Ein weiteres Anwendungsgebiet, in dem das neu definierte topologisch orientierte Ähnlichkeitsmaß auf Grundlage der fünfdimensionalen Merkmalsvektoren genutzt werden kann, ist die Ähnlichkeitssuche zwischen Objekten, die sich topologisch deutlich voneinander unterscheiden. Bei einer Objektmenge, die jeweils eine sehr unterschiedliche Anzahl von Extremitäten mit unterschiedlichen Längen vorweisen, erzielt das vorgestellte Ähnlichkeitsmaß gute Ergebnisse.

Ein Problem beim Ähnlichkeitsvergleich stellen demnach topologisch sehr ähnliche Objekte und Kurvenskelette dar, die durch fehlende Pruning-Methoden sehr

viele Skelettpfade aufweisen, welche aufgrund von Oberflächenveränderungen des Objektes entstanden sind. Durch die zusätzlichen Pfade wird das Ähnlichkeitsmaß verfälscht. Als Grundlage für den Ähnlichkeitsvergleich müssen also korrekt berechnete Kurvenskelette vorliegen.

Die Genauigkeiten der Berechnung von Kurvenskeletten mit der Distanztransformationsmethode sind nur bedingt präzise, sodass im Fall des Skelettierungsalgorithmus von Reniers [Ren08] eine Modifikation der Skelette erfolgen muss, bevor der Ähnlichkeitsvergleich durchgeführt werden kann.

5.2 Ausblick

Das vorgestellte Ähnlichkeitsmaß kann durch weitere Methoden erweitert und verbessert werden. Ein wichtiger Ansatz dazu ist, die einzelnen Dimensionen der Merkmalsvektoren $\mathbf{f}_v = (s_{v_1}, s_{v_2}, s_{v_3}, s_{v_4}, s_{v_5})^T$ durch eine Gewichtung $\mathbf{f}_{v_g} = (g_1 s_{v_1}, g_2 s_{v_2}, g_3 s_{v_3}, g_4 s_{v_4}, g_5 s_{v_5})^T$ zu präzisieren und auf bestimmte Anwendungsgebiete anpassbar zu machen. Eine Gewichtung der Merkmalsvektoren kann durch automatisierte Methoden berechnet werden. Dort kommt dann ein sogenanntes künstliches neuronales Netz zum Einsatz.

Neuronale Netze bilden einen Teilbereich der künstlichen Intelligenz und können durch mehrere Eingabewerte $s_{v_1}, s_{v_2}, s_{v_3}, s_{v_4}, s_{v_5}$ und einer gegebenen Funktion (hier: Kosinusmaß) die optimalen Gewichtungen g_1, g_2, g_3, g_4, g_5 für die einzelnen Dimensionen der Merkmalsvektoren berechnen.

Eine Veränderung des Ähnlichkeitsmaßes auf Grundlage der Veröffentlichung von Bai und Latecki [BL08] kann weitere Verbesserungen bei der Ähnlichkeitsbestimmung zwischen Objekten auf Grundlage derer Kurvenskelette mit sich bringen. Die beiden Wissenschaftler beschreiben ein innovatives Ähnlichkeitsmaß auf Grundlage von Kurvenskeletten von 2D-Objekten. Die Idee basiert vor allem auf der Überlegung, dass sich bewegliche Objekte zwar visuell voneinander unterscheiden, das Skelett sich jedoch in seiner Struktur nicht verändert. Aus dieser Tatsache heraus definieren die Autoren das Ähnlichkeitsmaß anhand der kürzesten Distanzen auf den Skelettpfaden zwischen Endpunkten.

In Abbildung 5.1 ist die Funktionsweise des Algorithmus ansatzweise dargestellt. Der Ähnlichkeitsvergleichsalgorithmus von Bai und Latecki findet zuerst die korrespondierenden Endpunkte zwischen den zu vergleichenden Objekten und berechnet dann die Merkmalsvektoren anhand der Distanzen der kürzesten Pfade zwischen Endpunkten.

Diesen Ansatz gilt es auf 3D-Kurvenskelette zu übertragen. Dabei spielt die Analyse der Unterschiede zwischen 2D- und 3D-Objekten eine wichtige Rolle. Auch

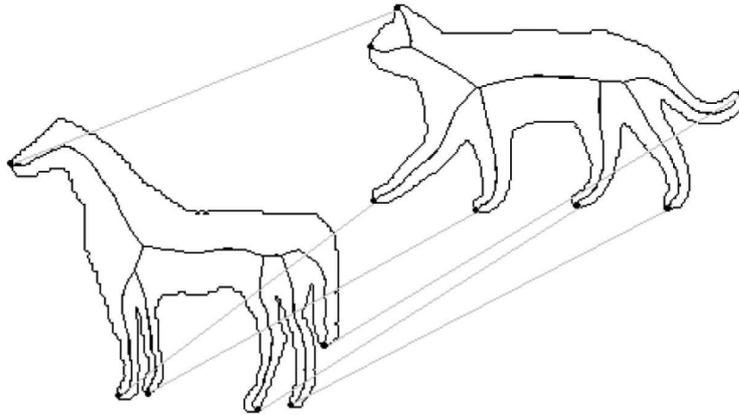


Abbildung 5.1: Algorithmus von Bai und Latecki für den Ähnlichkeitsvergleich zwischen 2D-Objekten

Effizienzüberlegungen, wie beispielsweise die Suche nach korrespondierenden Endpunkten zwischen komplexen 3D-Objekten, können, anders als im 2D-Raum, einen entscheidenden Unterschied bedeuten. Weiterführende Arbeiten müssen zeigen, ob sich der Ansatz von Bai und Latecki auch auf 3D-Objekte im diskreten dreidimensionalen Raum anwenden lässt.

Literaturverzeichnis

- [Ber94] BERTRAND, Aktouf Z. Gilles: A 3D thinning algorithms using subfields. In: *Proc. SPIE Conf. on Vision Geometry III, Vol.2356* (1994), S. 113–124
- [BL08] BAI, Xiang ; LATECKI, Longin J.: Path Similarity Skeleton Graph Matching. In: *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 30, NO. 7, JULY 2008* (2008)
- [Blu67] In: BLUM, H.: *A transformation for extracting new descriptors of shape*. Bd. Models for the Perception of Speech and Visual For. MIT Press, 1967, S. pp. 362–380
- [CSYB05] CORNEA, Nicu D. ; SILVER, Deborah ; YUAN, Xiaosong ; BALASUBRAMANIAN, Raman: Computing hierarchical curve-skeletons of 3D objects. In: *The Visual Computer Volume 21* (2005)
- [GF96] GE, Yaorong ; FITZPATRICK, Michael J.: On the generation of skeletons from discrete Euclidean distance maps. In: *Pattern Analysis and Machine Intelligence, Vol.18* (1996), S. 1055–1066
- [Goh07] In: GOHOUT, Wolfgang: *Mathematik für Wirtschaft und Technik*. Oldenbourg Wissenschaftsverlag, 2007, S. 8,9
- [Her04] In: HERMES, Thorsten: *Digitale Bildverarbeitung*. Hanser Verlag, 2004, S. 28
- [HF05] HASSOUNA, Sabry M. ; FARAG, Aly A.: Robust Centerline Extraction Framework Using Level Sets. In: *Computer Vision and Pattern Recognition* (2005)
- [Hom07] HOMANN, Hanno: *Implementation of a 3D thinning algorithm*. <http://www.sci.utah.edu/devbuilds/biomesh3d/FEMesher/references/ITKbinaryThinningImageFilter3D.pdf>. Version: 2007
- [ITK10] ITK, Kitware: *ITK Framework*. <http://www.itk.org>. Version: 2010

- [KS93] KIRYATI, Nahum ; SZÉKELY, Gabor: Estimating shortest paths and minimal distances on digitized three-dimensional surfaces. In: *Pattern Recognition Vol. 26* (1993), S. 1623–1637
- [Law06] In: LAWSON, Terry: *Topology: a Geometric Approach*. Oxford University Press, 2006, S. 44–46
- [LKC94] LEE, Ta-Chih ; KASHYAP, Rangasami L. ; CHU, Chong-Nam: Building Skeleton Models via 3-D Medial Surface Axis Thinning Algorithms. In: *CVGIP: Graphical Models and Image Processing Vol. 56* (1994), S. 462–478
- [LLS92] LAM, Louisa ; LEE, Seong-Whang ; SUEN, Ching Y.: Thinning methodologies-a comprehensive survey. In: *Pattern Analysis and Machine Intelligence* (1992), S. 869–885
- [MS96] MA, Min C. ; SONKA, Milan: A Fully Parallel 3D Thinning Algorithm and Its Applications. In: *Computer Vision and Image Understanding, Vol. 64* (1996), S. 420–433
- [OK95] OGNIIEWICZ, Robert L. ; KÜBLER, Olaf: Hierarchic Voronoi skeletons. In: *Pattern Recognition, Vol. 28* (1995), S. 343–359
- [PK98] PALÁGYI, Kálmán ; KUBA, Attila: A 3D 6-subiteration thinning algorithm for extracting medial lines. In: *Pattern Recognition Letters 19* (1998), S. 613–627
- [PM09] In: PATRIKALAKIS, Nicholas M. ; MAEKAWA, Takashi: *Shape Interrogation for Computer Aided Design and Manufacturing*. Springer Verlag, 2009, S. 266
- [Ren08] RENIERS, Dennie: *Skeletonization and Segmentation of Binary Voxel Shapes*, Technische Universiteit Eindhoven, Diss., 2008
- [RT06] RENIERS, Dennie ; TELEA, Alexandru: Quantitative comparison of tolerance-based feature transforms. In: *First International Conference on Computer Vision Theory and Applications (VISAPP)* (2006), S. 107–114
- [Sch05] In: SCHMITT, Ingo: *Ähnlichkeitssuche in Multimedia-Datenbanken*. Oldenbourg Wissenschaftsverlag, 2005, S. 231,232
- [Sch07] In: SCHECK, Florian: *Theoretische Physik 1: Mechanik*. Springer Verlag, 2007, S. 11,12

- [SLSK07] SHARF, Andrei ; LEWINER, Thomas ; SHAMIR, Ariel ; KOBBELT, Leif: On-the-fly curve-skeleton computation for 3d shapes. In: *Eurographics 2007 (Computer Graphics Forum)* Bd. 26. Prague : Eurographics, 2007, S. 323–328
- [SS09] In: SCRIBA, Christoph J. ; SCHREIBER, Peter: *5000 Jahre Geometrie: Geschichte, Kulturen, Menschen*. Springer Verlag, 2009, S. 49 ff.
- [Sto06] In: STOCK, Wolfgang G.: *Information Retrieval*. Oldenbourg Wissenschaftsverlag, 2006, S. 63–65
- [TS06] TAMAL, Dey K. ; SUN, Jian: Defining and Computing Curve-skeletons with Medial Geodesic Function. In: *Eurographics Symposium on Geometry Processing* (2006)
- [VTK09] VTK, Kitware: *Java Wrapping*. http://www.vtk.org/Wiki/VTK/Java_Wrapping. Version: 2009