FAKULTÄT FÜR
INFORMATIK

Diplomarbeit

Interaktionstechniken zur Nachkorrektur medizinischer Segmentierungen

---

Daniel Proksch

Betreuer:

Dipl.-Ing. Jana Dornheim
Prof. Dr.-Ing. Bernhard Preim

Proksch, Daniel:
Interaktionstechniken zur Nachkorrektur medizinischer Segmentierungen
Diplomarbeit, Otto-von-Guericke-Universität
Magdeburg 2009.

Selbständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbständig und nur unter Zuhilfenahme der angegebenen Quellen erstellt habe.

Magdeburg, den 11. April 2009

_____

Daniel Proksch

Danksagung

Allem voran geht ein herzliches Dankeschön an Eva-Maria und Gerhard Proksch sowie Magdalena Buch, die durch ihre ausdauernde Unterstützung mein Studium überhaupt erst ermöglicht haben. Ohne Euch wäre ich nicht, wo ich heute bin. Weiterhin danke ich Jana Dornheim für die Betreuung meiner Diplomarbeit und Peter Hahn für die Hilfestellung bei Softwareproblemen. Ein ganz besonderes Dankeschön geht an Juliane Malisch für das fleißige Korrekturlesen der Arbeit ... und überhaupt. Außerdem danke ich allen, die sich die Zeit genommen haben an der Nutzerstudie teilzunehmen und die entwickelten Interaktionstechniken zu testen.

Finally, a very special 'thank you' goes to John Whaley for waiting across the pond.

Inhaltsverzeichnis

1 Einleitung **1**
- 1.1 Motivation . . . 1
- 1.2 Aufgabenstellung . . . 2
- 1.3 Aufbau der Arbeit . . . 3

2 Medizinischer Hintergrund **5**
- 2.1 Anatomische Grundlagen . . . 5
  - 2.1.1 Schildknorpel . . . 5
  - 2.1.2 Halslymphknoten . . . 7
  - 2.1.3 Halstumoren . . . 9
- 2.2 Medizinische Bildgebung . . . 10
  - 2.2.1 Röntgen-Computertomographie . . . 11
  - 2.2.2 Darstellung der Halsstrukturen in CT-Aufnahmen . . . 14
- 2.3 Zusammenfassung . . . 17

3 Stand der Forschung **19**
- 3.1 Mensch-Computer-Interaktion . . . 19
  - 3.1.1 Terminologie . . . 20
  - 3.1.2 Direkte Manipulation . . . 21
- 3.2 Segmentierung medizinischer Bilddaten . . . 22
  - 3.2.1 Deformierbare Modelle . . . 23
  - 3.2.2 Stabile Feder-Masse-Modelle . . . 24
  - 3.2.3 Interaktion in Segmentierungsverfahren . . . 25
- 3.3 Manipulation von Polygonmodellen . . . 32
  - 3.3.1 Intuitive Interaktionstechniken . . . 33
  - 3.3.2 Laplacian Modeling Framework . . . 37
- 3.4 Zusammenfassung . . . 40

4 Analyse der Ausgangsituation 41
4.1 Ausgangsituation . . . 41
4.2 Schlussfolgerungen . . . 44
4.3 Typische Korrekturfälle . . . 46
4.3.1 Allgemeine Beobachtungen . . . 46
4.3.2 Segmentierung der Halsstrukturen . . . 47
4.4 Anforderungen . . . 49
4.5 Zusammenfassung . . . 50

5 Entwurf der Interaktionstechniken 51
5.1 Aufgegriffene Ideen . . . 51
5.2 Allgemeine Konzepte . . . 52
5.2.1 Visualisierung . . . 52
5.2.2 Begrenzung des Einflussbereiches . . . 56
5.2.3 Laplacian Deformation . . . 57
5.2.4 Detailgrad der Modelle . . . 62
5.3 Entwickelte Interaktionstechniken . . . 64
5.3.1 Bulge Tool . . . 64
5.3.2 Traction Tool . . . 69
5.3.3 Sketch Tool . . . 71
5.3.4 Bending Tool . . . 74
5.4 Zusammenfassung . . . 78

6 Implementierung 81
6.1 Verwendete Software . . . 81
6.2 Entwickelte Komponenten . . . 82

7 Evaluierung 87
7.1 Eignung der Interaktionstechniken . . . 87
7.2 Nutzertests . . . 91

8 Zusammenfassung und Ausblick 93
8.1 Zusammenfassung . . . 93
8.2 Ausblick . . . 95

Literaturverzeichnis 97

Abbildungsverzeichnis 103

1 Einleitung

1.1 Motivation

Digitale Bilder spielen eine bedeutende Rolle in der medizinischen Diagnostik und Therapieplanung. Durch verschiedene Bildgebungsverfahren ist es heute möglich, das Körperinnere von Patienten ohne schädigende, operative Eingriffe sichtbar zu machen. Die ständige Weiterentwicklung der verwendeten Techniken ermöglicht es zudem, dass Aufnahmen von immer höherer Auflösung und besserer Qualität, in großen Mengen zur Verfügung stehen. Für die effiziente Auswertung dieser Bilder ist eine computergestützte Verarbeitung unerlässlich.

Eine häufig benötigte Bildverarbeitungsaufgabe ist die Segmentierung der medizinischen Aufnahmen. Das Ziel ist es dabei, solche Bildregionen zu isolieren, die interessierende, anatomische Strukturen repräsentieren. Für das geschulte Auge des Mediziners stellt dies im Allgemeinen kein größeres Problem dar, bringt aber bei manueller Durchführung einen erheblichen zeitlichen Aufwand mit sich. Daher ist die möglichst automatische Segmentierung ein klassischer Einsatzbereich medizinischer Bildverarbeitungssoftware.

Die Identifizierung einzelner Objekte in Bildern ist für das visuelle System des Menschen eine einfach zu lösende Aufgabe. In der computergestützten Bildverarbeitung dagegen, zählt es zu den kompliziertesten Problemen. Vor allem im medizinischen Bereich aber, wo auf Grundlage der digitalen Bilder mitunter lebenswichtige Entscheidungen getroffen werden, kommt der Genauigkeit und Korrektheit der verarbeitenden Algorithmen eine besondere Bedeutung zu.

Prinzipiell bestehen gute Voraussetzungen für die Entwicklung robuster Segmentierungsverfahren auf medizinischen Bildern. Die Aufnahmen werden unter kontrollierten, gleichbleibenden Bedingungen gewonnen und zusammen mit allen wichtigen Parametern in standardisierten Dateiformaten abgelegt. So ist es möglich, bestimmte Annahmen über die zu erwartende charakteristische Darstellung anatomischer Strukturen in den Bildern zu treffen und in die Segmentierungsalgorithmen zu integrieren. Da es im menschlichen Körper eine Vielzahl unterschiedlichster Organe

und Strukturen gibt, ist es in der Regel notwendig, hochspezialisierte Methoden zu entwickeln, mit denen jeweils eine ganz bestimmte Objektklasse segmentiert wird.

Häufig werden dazu spezifische Formeigenschaften herangezogen. Diese durchaus naheliegende Herangehensweise sieht sich allerdings mit einem ernsten Problem konfrontiert: Die Objekte, die im Rahmen der medizinischen Bildverarbeitung betrachtet werden, unterliegen einer besonders hohen Variabilität. Form und Größe einzelner Organe variieren mitunter stark zwischen verschiedenen Patienten. Sofern es sich um im Körper mehrfach auftretende Strukturen (z.B. Lymphknoten) handelt, findet sich eine derartige Variabilität sogar innerhalb ein und desselben Patienten. Noch problematischer wird es, wenn die abweichende Form auf pathologischen Organveränderungen beruht. Abhängig vom Ausmaß der Deformierung, verringert sich dann die Wahrscheinlichkeit, dass ein solches Objekt vom automatischen Verfahren korrekt segmentiert werden kann. Gleichzeitig besteht aber gerade in solch einem Fall die erhöhte Notwendigkeit, ein optimales Segmentierungsergebnis zu erzielen, damit dem Mediziner bestmögliche Bedingungen für die Befundung und Behandlungsplanung zur Verfügung stehen.

Da die Segmentierungen medizinischer Bilder besonders hohen Ansprüchen genügen müssen, ist eine Überprüfung der Ergebnisse durch ärztliches Personal unumgänglich. Obwohl für viele Objektklassen bereits leistungsfähige, weitgehend automatisierte Segmentierungsverfahren verfügbar sind, ist aus o.g. Gründen häufig eine Korrektur der Ergebnisse erforderlich. Weil in den allermeisten Fällen dreidimensionale Bilddaten verarbeitet werden, müssen etwaige Ausbesserungen nicht selten in vielen Schichten eines Datensatzes vorgenommen werden. Mit den herkömmlichen Korrekturmechanismen kann dies schnell zur zeitraubenden Arbeit werden. Wünschenswert wären deshalb solche Interaktionstechniken, die eine effizientere Korrektur von Segmentierungsergebnissen ermöglichen als bisher, damit dem Mediziner mehr Zeit für wichtigere Aufgaben bleibt.

1.2 Aufgabenstellung

Im Rahmen der vorliegenden Diplomarbeit sollen alternative Ansätze für Interaktionstechniken zur möglichst effizienten und intuitiven Nachkorrektur medizinischer Segmentierungen entwickelt und implementiert werden, die über das Einzeichnen der korrekten Objektkonturen in jeder einzelnen Schicht eines tomographischen Datensatzes hinausgehen. Dies soll in erster Linie im Hinblick auf modellbasierte Segmentierungen aus der HNO-Medizin geschehen. Dazu stehen CT-Datensätze mit Segmentierungen des Schildknorpels und der Halslymphknoten sowie die dazu verwendeten Modelle zur Verfügung. Außerdem soll geprüft werden, inwieweit sich eine Anwendung der entwickelten Techniken zur Nachbearbeitung manueller

Segmentierungen von Halstumoren eignet. Dies ist etwa dann sinnvoll, wenn durch neue Erkenntnisse über die individuelle Patientenanatonmie, Anpassungen an den Segmentierungsergebnissen erforderlich werden.

Zu Beginn der Arbeit soll eine Analyse der speziellen Anforderungen an die Interaktionen zur Nachkorrektur durchgeführt werden. Darauf aufbauend sollen mehrere alternative Interaktionstechniken entworfen und implementiert werden. Für die Anwendung bei modellbasierten Segmentierungen soll versucht werden das vorhandene Formwissen zu nutzen, um den Korrekturaufwand weiter zu reduzieren. Schließlich sollen die entwickelten Verfahren evaluiert und miteinander verglichen werden.

1.3 Aufbau der Arbeit

Kapitel 2 gibt einen Überblick über die medizinischen Grundlagen dieser Diplomarbeit. Dabei werden sowohl Anatomie und Darstellung der segmentierten Beispielstrukturen in CT-Datensätzen, als auch die CT-Bildgebung selbst behandelt.

Kapitel 3 widmet sich dem derzeitigen Stand der Forschung im Umfeld der Aufgabenstellung. Hier werden neben grundlegenden Betrachtungen zur Mensch-Computer-Interaktion, vor allem die Rolle von Interaktion in Segmentierungsverfahren, sowie Forschungsergebnisse zur intuitiven Manipulation von 3D-Modellen diskutiert.

Kapitel 4 beschäftigt sich mit der Analyse der Ausgangssituation und Schlussfolgerungen für die Lösung der Aufgabenstellung.

Kapitel 5 erläutert den Entwurf der entwickelten Interaktionstechniken zur Nachkorrektur medizinischer Segmentierungen.

Kapitel 6 beschreibt die prototypische Implementierung der Interaktionstechniken im Projekt *SMSMCorrectionToolbox*.

Kapitel 7 widmet sich der Evaluierung. Hier werden die Erkenntnisse aus Untersuchungen zur Eignung und Nutzbarkeit der entwickelten Interaktionstechniken besprochen.

Kapitel 8 gibt schließlich eine Zusammenfassung und diskutiert die erzielten Ergebnisse der Arbeit. Abschließend werden offene Fragen benannt und zukünftig zu lösende Aufgaben aufgezeigt.

2 Medizinischer Hintergrund

Dieses Kapitel erläutert die medizinischen Grundlagen der vorliegenden Arbeit. Da die Entwicklung der interaktiven Korrekturmechanismen im Rahmen dieser Diplomarbeit anhand von Segmentierungsergebnissen aus der HNO-Medizin gezeigt wird, folgt zunächst eine Betrachtung der Anatomie der segmentierten Strukturen. Die radiologischen Bilder die den Beispielsegmentierungen zu Grunde liegen, wurden allesamt mittels der Röntgen-Computertomographie (CT) gewonnen. Deshalb wird dieses Bildgebungsverfahren anschließend näher behandelt. Das Kapitel schließt mit Abbildungen und Erläuterungen zur charakteristischen Darstellung der hier besprochenen anatomischen Strukturen in CT-Aufnahmen

2.1 Anatomische Grundlagen

Die HNO-Heilkunde ist das Fachgebiet der Humanmedizin, das sich mit der Behandlung von Erkrankungen des Kopf- und Halsbereiches auseinandersetzt. Gerade im Bereich des Halses befindet sich auf engstem Raum, eine Vielzahl anatomischer Strukturen, die der HNO-Mediziner begutachten können muss. Werden dazu tomographische Aufnahmen angefertigt, ist es z.B. als Voraussetzung für eine dreidimensionale Visualisierung häufig erforderlich, die einzelnen Strukturen zu segmentieren. Für die Lösung der Aufgabenstellung der vorliegenden Arbeit steht eine Auswahl von Segmentierungen aus der Halsregion zur Verfügung. Im Folgenden sollen die betreffenden anatomischen Strukturen anhand von Informationen aus [21], [23], [26] sowie [27] näher vorgestellt werden.

2.1.1 Schildknorpel

Eine wichtige Struktureinheit des Halses ist der Kehlkopf (*Larynx*). Er bildet sowohl die Öffnung, als auch den Abschluss der Luftröhre zum Rachenraum. Die Primärfunktion des Kehlkopfes besteht darin, zu verhindern, dass Nahrung in die Luftröhre gelangen kann. Dies wird erreicht, indem beim Schluckakt der Kehlkopf nach vorne gezogen und so die Luftröhre vom Kehldeckel verschlossen wird.

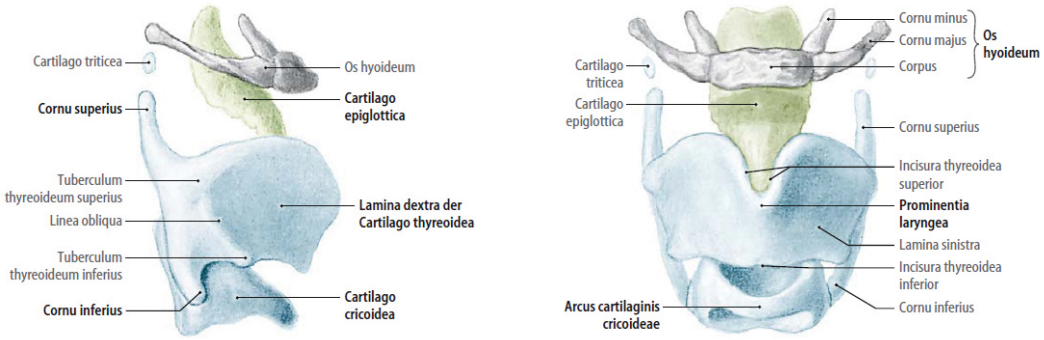


Abbildung 2.1: Kehlkopfskelett mit Zungenbein (*Os hyoideum*), Kehldeckelknorpel (*Cartilago epiglottica*), Schildknorpel (*Cartilago thyreoidea*) und Ringknorpel (*Cartilago crioidea*). (a) Ansicht von rechts, (b) Ansicht von vorn. Quelle: [60]

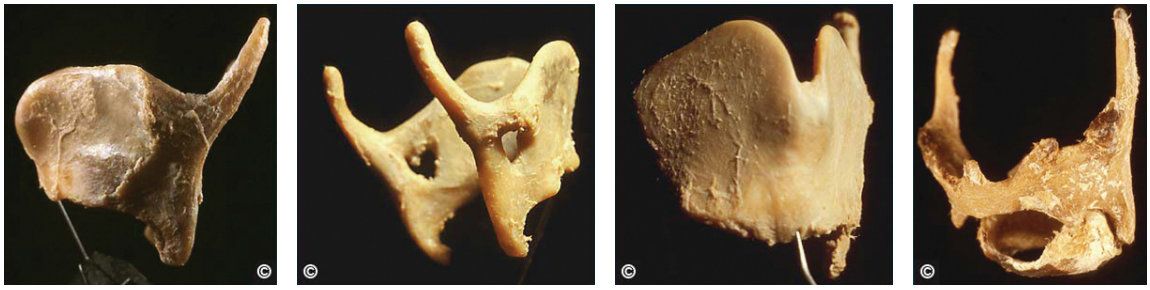
Darüber hinaus ist der Kehlkopf auch für die Stimmgebung verantwortlich. Dazu befinden sich in dessen Inneren zwei mit Schleimhaut umkleidete Gewebelappen, die Stimmlippen[1] genannt werden. Bei der Lautbildung werden die Stimmlippen durch Ausatemluft in Schwingung versetzt, wodurch Töne entstehen. Das Gerüst des Kehlkopfes bildet das aus verschiedenen Knorpeln zusammengesetzte Kehlkopfskelett (Abb. 2.1). Es besteht aus dem Kehldeckel, dem Schildknorpel, dem Ringknorpel und den Stellknorpeln.

Der Schildknorpel (*Cartilago thyreoidea*) ist aufgrund vorliegender Beispielsegmentierungen für diese Arbeit von besonderem Interesse. Er ist der größte Knorpel des Kehlkopfes und umschließt dessen Innenraum. Er besteht aus zwei an der Vorderseite zusammengewachsenen Seitenplatten, die in einem spitzen Winkel zueinander stehen. Am hinteren Ende des Schildknorpels befinden sich insgesamt vier Hörner, wobei je zwei von ihnen oberhalb und unterhalb der Seitenplatten angewachsen sind. Die oberen Hörner sind lang und schmal, während die unteren Hörner eher kurz und dick sind. An der oberen Vorderseite, dort wo beide Platten auseinander laufen, befindet sich der sogenannte Adamsapfel, eine Verdickung, die besonders beim Mann stark ausgeprägt ist.

Abbildung 2.2 zeigt unterschiedliche Ausprägungen der Form bei verschiedenen Schildknorpeln. Hier werden beispielhaft anatomische Unterschiede deutlich, trotz derer ein Segmentierungsverfahren korrekte Ergebnisse produzieren sollte. Alle abgebildeten Exemplare ähneln sich in ihrer grundsätzlichen Form und sind für den Mediziner eindeutig als Schildknorpel zu erkennen. Allerdings zeigen sich gravieren-

[1] umgangssprachlich auch Stimmbänder

de Unterschiede. So kann der Knorpel gelegentlich Löcher aufweisen (Abb. 2.2b) und die Hörner unterschiedlich lang und gar einseitig stark unterentwickelt sein (Abb. 2.2c). Zwar sind extreme Deformierungen wie in Abbildung 2.2d vergleichsweise selten, müssen aber dennoch - gerade auch bei pathologischen Ursachen - korrekt segmentiert werden können.



(a) (b) (c) (d)

Abbildung 2.2: Verschiedene Ausprägungen der Schildknorpelform: (a) unauffällige Anatomie, (b) durchlöcherte Seitenplatten, (c) einseitig unterentwickelte Hörner, (d) besonders ungewöhnliche Anatomie aufgrund pathologischer Einwirkung. Photographien von Dr. Willard R. Zemlin. Mit freundlicher Genehmigung von Eileen Zemlin.

2.1.2 Halslymphknoten

Das lymphatische System des Menschen ist, ähnlich dem Blutkreislauf, ein Geflecht aus Leitungsbahnen, das den gesamten Körper durchzieht. Es dient der Beseitigung von Fremdkörpern, Zelltrümmern und vor allem Krankheitserregern aus dem Körper. Damit ist es ein wichtiger Bestandteil des menschlichen Immunsystems. Die eigentliche Zerstörung, der durch die Lymphflüssigkeit (Lymphe) abtransportierten Viren und Bakterien, findet in den Lymphknoten statt. Dies sind etwa $5\,\mathrm{mm}$ - $20\,\mathrm{mm}$ große Knötchen, in denen die Lymphe gefiltert wird. Die Lymphknoten treten gehäuft in bestimmten Körperregionen wie den Achselhöhlen, der Leistengegend oder der Bauchwand auf. Besonders konzentriert finden sie sich vor allem im Kopf- und Halsbereich (Abb. 2.3). Die Anzahl der dort vorhandenen Lymphknoten wird auf etwa 300 geschätzt. Ein Großteil davon ist jedoch zu klein um radiologisch dargestellt werden zu können. Ist der Organismus erkrankt, kann oftmals ein Anschwellen der Lymphknoten (*Lymphknotenhyperplasie*) beobachtet werden. Vor allem bei unspezifischen Entzündungen, Infektionen oder Systemerkrankungen, kommt es zu solch vorübergehenden Vergrößerungen der Lymphknoten.

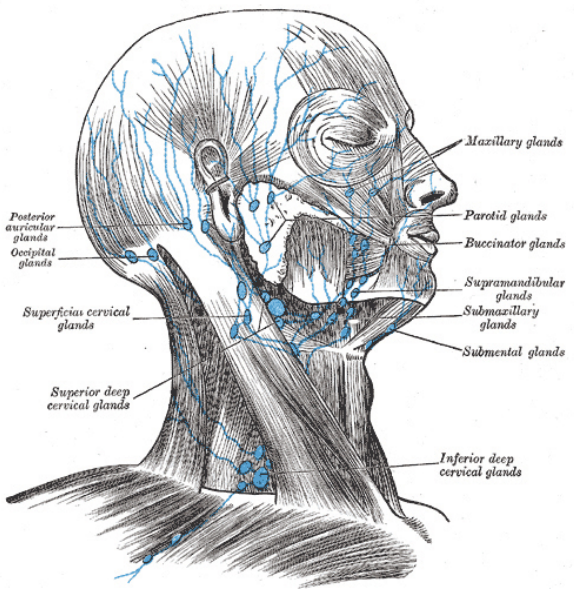


Abbildung 2.3: Lymphknoten im Kopf- und Halsbereich. Quelle: [27]

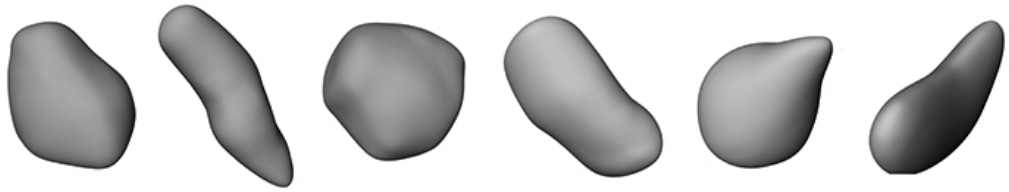


Abbildung 2.4: Beispiele unterschiedlich geformter Lymphknoten. Dreidimensionale Rekonstruktionen von CT-Segmentierungen. Zur besseren Vergleichbarkeit wurden Größenunterschiede in der Darstellung nicht berücksichtigt.

In Abbildung 2.4 ist eine Auswahl verschieden geformter Halslymphknoten dargestellt. Wie zu sehen ist, variiert die Gestalt zwischen länglich ellipsoid und annähernd kugelförmig. Die elliptische Form stellt dabei den klinisch unauffälligen Normalzustand dar. Das Verhältnis von Längs- zu Transversalachse hat in diesem Fall meist einen Wert von ≤ 2 [22]. Bei pathologisch bedingten Vergrößerungen nehmen die Lymphknoten eine kugelähnliche Gestalt an. Wie groß sie dabei werden, kann nicht allgemein beantwortet werden und hängt unter anderem vom umliegenden Gewebe ab. Benachbarter Knochen etwa, gibt aufgrund seiner Festigkeit nicht nach und begrenzt somit den Raum, der dem Lymphknoten zur Verfügung steht. Liegt ein vergrößerter Lymphknoten dagegen in der Nähe von Weichteilgewebe, kann er ungehindert wachsen, da benachbarte Organe und Gefäße keine Barriere darstellen und gegebenenfalls einfach verdrängt werden.

2.1.3 Halstumoren

Unter dem Begriff Tumor wird zunächst jegliche Neubildung von Gewebe verstanden, die auf erhöhter Zellteilung beruht. Dabei unterscheidet man zwischen gutartigen (*benignen*) und bösartigen (*malignen*) Veränderungen. Zur Klasse der gutartigen Tumoren zählen z.B. Warzen, Adenome, aber auch einfache, durch Entzündungen hervorgerufene Schwellungen. Bösartige Tumoren[2] werden unterteilt in Karzinome, Sarkome und Lymphome. Karzinome wachsen aus Deck- und Drüsengewebe, Sarkome aus Binde- und Stützgewebe. Lymphome sind bösartige Veränderungen des Lymphgewebes.

Ob ein Tumor gutartig oder bösartig ist, hängt von dessen Wachstumsverhalten ab. Während benigne Tumoren benachbartes Gewebe lediglich verdrängen, greifen maligne Geschwülste darauf über und zerstören es. Außerdem kommt es bei bösartigen Tumoren zur Bildung von Metastasen, also Absiedlungen von Tumorzellen in andere Körperregionen und Lymphknoten. Die Einteilung bösartiger Tumoren wird anhand der sogenannten TNM-Klassifikation vorgenommen. Dabei wird jeder Tumor durch drei Zahlenwerte charakterisiert. Der T-Wert (0..4) bestimmt die Größe und Ausdehnung des Primärtumors, der N-Wert (0..3) steht für die Stärke des Lymphknotenbefalls und der M-Wert (0/1) gibt an, ob Fernmetastasen vorhanden sind.

In der HNO-Medizin ist Kehlkopfkrebs (*Larynxkarzinom*) der am häufigsten diagnostizierte maligne Tumor. Die bösartigen Wucherungen entstehen dabei zumeist aus dem Plattenepithel, der Schleimhaut, die den Kehlkopf von innen auskleidet. Bei allen malignen Tumoren der Kopf-Hals-Region kommt es insbesondere zum Metastasenbefall der Halslymphknoten. Es ist sogar möglich, dass solche Lymphknotenmetastasen zum ersten Symptom einer Krebserkrankung werden, wenn der Primärtumor sehr klein ist und versteckt, z.B. am Zungengrund oder den Rachenmandeln liegt und bisher keine spürbaren Beschwerden verursacht. Oftmals können bösartige Tumoren im Halsbereich auf starken Tabak- und Alkoholkonsum zurückgeführt werden. [23]

Da sich tumoröse Gewebsveränderungen durch unkontrolliertes Wachstum auszeichnen, lassen sich keine allgemeinen Aussagen zur Gestalt der Tumoren machen. Abbildung 2.5 verdeutlicht, wie unterschiedlich geformt die Geschwülste erscheinen können. Da sich zudem gerade Halstumoren im CT-Bild nicht durch ihren Grauwert von der Umgebung abheben, existieren bis heute keine befriedigenden, automatischen Verfahren für deren Segmentierung in medizinischen Bildern. Deshalb muss diese Aufgabe in den meisten Fällen manuell durch den Mediziner erledigt werden.

[2] umgangssprachlich auch Krebs

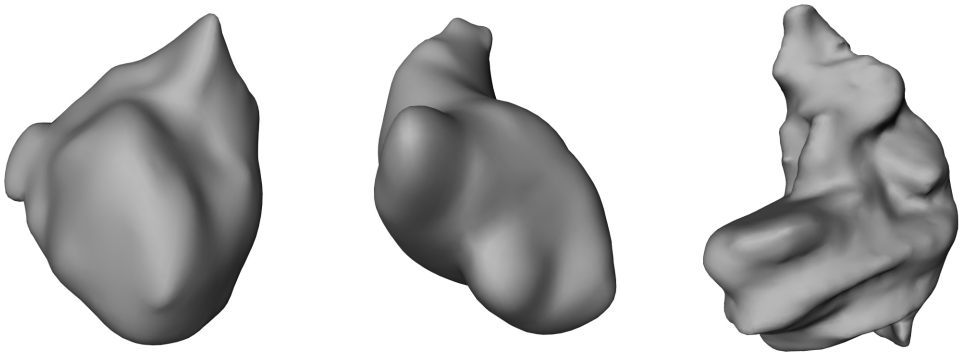


Abbildung 2.5: Beispiele verschiedenartiger Halstumoren. Dreidimensionale Rekonstruktion von CT-Segmentierungen.

2.2 Medizinische Bildgebung

Im klinisch medizinischen Umfeld kommen heute oftmals tomographische Bildgebungsverfahren zur Anwendung. Diese ermöglichen es, im Gegensatz zum konventionellen planaren Röntgen, ein dreidimensionales Abbild vom Inneren der Patienten zu erzeugen. Vertreter dieser Aufnahmeverfahren sind die Magnet-Resonanz-Tomographie (MRT), die Positronen-Emissions-Tomographie (PET) und die Röntgen-Computertomographie (CT). Die genannten Verfahren unterscheiden sich jeweils hinsichtlich der physikalischen Effekte, auf denen die Bildgewinnung basiert. Während beim MRT die unterschiedlichen magnetischen Eigenschaften verschiedener Gewebe ausgenutzt werden, misst man beim PET die Emission einer in den Körper eingebrachten, schwach radioaktiven Substanz. Für CT dagegen, wird Röntgenstrahlung benutzt. Außerdem werden PET und seltener auch MRT, zur Abbildung funktioneller Prozesse verwendet, während CT-Aufnahmen die Anatomie darstellen.

Für die HNO-Heilkunde sind besonders CT und MRT von Bedeutung. Beide Verfahren haben unterschiedliche Vor- und Nachteile. CT-Aufnahmen bieten einen vergleichsweise schlechten Weichteilkontrast. Außerdem muss der Patient für die Bildgewinnung schädlicher Strahlung ausgesetzt werden. MRT-Bilder dagegen, haben einen wesentlich höheren Weichteilkontrast, die Interpretation erfordert jedoch besondere, radiologische Erfahrung. Aus Gesprächen mit unserem klinischen Partner ging hervor, dass aus chirurgischer Sicht deshalb oftmals CT bevorzugt wird. Außerdem sind MRT-Aufnahmen zeitaufwändiger und bedeuten aufgrund der beengenden Aufnahmesituation in der Tomographenröhre zusätzlichen Stress für die oftmals geschwächten Patienten. Da auch alle im Rahmen dieser Arbeit verwendeten Beispielsegmentierungen auf CT-Bildern generiert wurden, wird im Folgenden näher auf dieses Bildgebungsverfahren eingegangen.

Abbildung 2.6: Modernes CT-Aufnahmesystem (Siemens Sensation 16).

2.2.1 Röntgen-Computertomographie

Wie bereits erwähnt, beruht die Röntgen-Computertomographie auf der Durchleuchtung des Körpers mit Röntgenstrahlung. Dass sich verschiedene anatomische Strukturen im CT-Bild durch unterschiedliche Grauwerte darstellen, liegt in der jeweils gewebespezifischen Fähigkeit zur Absorption der Strahlung begründet.

Bildgewinnung. Bei modernen CT-Systemen (Abb. 2.6) läuft die Bildaufnahme wie folgt ab: Der Patient befindet sich auf einer beweglichen Liege und wird langsam in Richtung seiner Längsachse durch die Aufnahmeeinheit geschoben. Diese besteht im Wesentlichen aus einer Röntgenquelle und einem gegenüberliegenden Detektorarray. Während fächerförmig Röntgenstrahlen ausgesandt werden, rotiert die Aufnahmeeinheit um den Patienten, der so allmählich aus allen Richtungen durchstrahlt wird. Die Detektoren registrieren dabei die Energie der Strahlung (bzw. den linearen Schwächungskoeffizienten), die den Körper des Patienten auf der gegenüberliegenden Seite wieder verlässt. Eine vollständige Rotation der Aufnahmeeinheit resultiert somit in einer Vielzahl verschiedener Einzelprojektionen der betreffenden axialen Schnittebene des gescannten Körpers. Durch komplexe mathematische Verfahren wird mittels gefilterter Rückprojektion ein zweidimensionales Abbild dieser Körperschicht berechnet. Da der Patient im Laufe der Aufnahme durch die Strahlenebene geschoben wird, entsteht eine Serie solcher Schichtbilder. Aneinandergereiht ergeben diese schließlich einen dreidimensionalen Datensatz, der die räumliche Rekonstruktion des Körperinneren ermöglicht.

Aufnahmeparameter. Die einzelnen Datensatzelemente repräsentieren diskrete Volumen des gescannten Körpers und werden Voxel (*volumetric pixel*) genannt. Die Kantenlänge der Voxel in x-y-Richtung bestimmt dabei die räumliche Auflösung innerhalb eines Schichtbildes. Die Ausdehnung in z-Richtung wird durch die Breite des verwendeten Röntgenstrahls definiert und entspricht der Schichtdicke. Neben den Voxelabmessungen, ist der Schichtabstand ein weiterer wichtiger Parameter computertomographischer Bilder. Typische Werte für die x-y-Ausdehnung der Voxel liegen im Bereich von 0.1 mm - 2 mm. Die Schichtdicke beträgt in der Regel etwa 0.5 mm - 10 mm, kann aber vor allem zur Verringerung der Strahlenbelastung auch wesentlich größere Werte annehmen[3]. Haben alle Kanten der Voxel die gleiche Länge und gilt Schichtabstand $\leq$ Schichtdicke, ist es möglich, ohne Verlust an Auflösung beliebige Bildebenen und qualitativ hochwertige 3D-Darstellungen zu rekonstruieren.

Darstellung. Die Grauwerte G der Voxel sind vom linearen Schwächungskoeffizienten μ_{Gewebe} des Gewebes im repräsentierten Körpervolumen abhängig und werden zur besseren Vergleichbarkeit von Aufnahmen aus unterschiedlichen CT-Systemen in *Hounsfield-Einheiten* (HE) angegeben. Die Umrechnung erfolgt dabei gemäß

$$G = \frac{(\mu_{Gewebe} - \mu_{Wasser})}{\mu_{Wasser}} \cdot 1000\,\text{HE}. \tag{2.1}$$

Durch die Normierung der Grauwerte ist es möglich, verschiedenen organischen Geweben feste Wertebereiche in den CT-Bildern zuzuordnen. So wird Wasser stets mit 0 HE und Luft mit -1000 HE dargestellt. Der Grauwert von Weichteilgewebe liegt in CT-Aufnahmen im Bereich von etwa 20 HE bis 70 HE. Knochen dagegen, weist die stärkste Röntgenschwächung auf und ist daher durch Werte bis zu 2000 HE repräsentiert. Der gesamte Wertebereich in CT-Datensätzen umfasst bei üblicherweise 12Bit eine Skala von -1024 HE bis +3071 HE. Demgegenüber steht ein Umfang von maximal 60 bis 80 vom Menschen unterscheidbaren Grauabstufungen [30]. Deshalb wird bei der Darstellung der CT-Daten meist nur einem begrenzten Wertebereich die gesamte Grauskala zugeordnet. Werte unterhalb des gewählten Bereichs werden schwarz, Werte oberhalb dagegen weiß dargestellt. Dieser *Fensterung* genannte Vorgang ermöglicht es dem Mediziner, nur die jeweils diagnostisch relevanten Informationen zu betrachten. Definiert wird das Fenster durch den mittleren CT-Wert (*Center*) und die Differenz zwischen oberer und unterer Grenze (*Width*) des darzustellenden Wertebereichs. Letzterer Parameter bestimmt dabei den Kontrast des Bildes. Tabelle 2.1 zeigt Beispiele für Grauwertfenster zur optimalen Darstellung verschiedener Organe in CT-Aufnahmen [22].

[3] Von unserem klinischen Partner liegen CT-Datensätze mit bis zu 50 mm Schichtdicke vor.

Organ	Center	Width
Lunge	-650	1500
Weichteile	40	400
Hals	50	300
Knochen	500	2000

Tabelle 2.1: Beispiele verschiedener Grauwertfenster.

Zur Darstellung von CT-Datensätzen werden hauptsächlich drei orthogonale Schnittebenen (Abb. 2.7) verwendet. Am häufigsten erfolgt die Darstellung von *axialen* Schnittbildern. Diese liegen senkrecht zur Körperachse und entsprechen damit der Schnittführung, in der die Aufnahmen gewonnen werden. Darüber hinaus werden oftmals auch *koronare*, also parallel zur Stirn verlaufende und *sagittale*, parallel zur Medialebene des Patienten liegende Schnittbilder betrachtet.

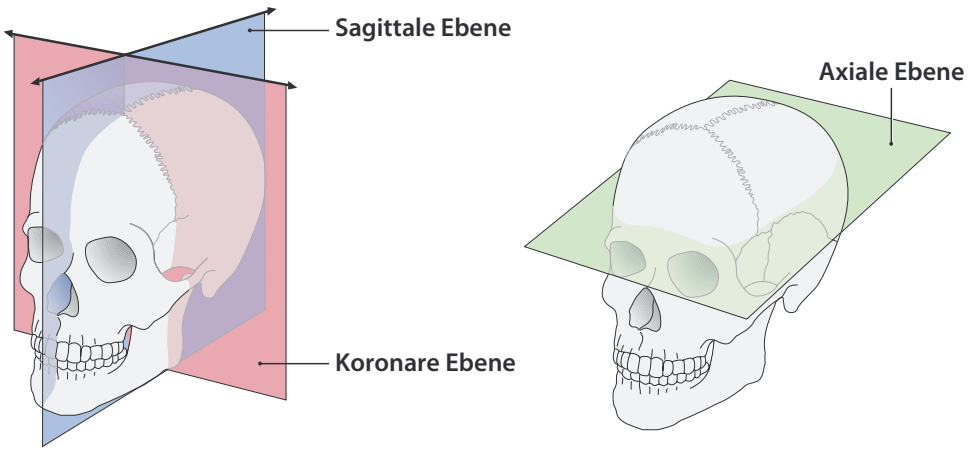


Abbildung 2.7: Radiologische Schnittebenen. Quelle: [55]

Artefakte. Entscheidend für den Erfolg automatisierter Segmentierungsverfahren auf CTs - und damit auch für die Frage, ob und wie stark der Nutzer korrigierend eingreifen muss - ist besonders auch die Qualität der Eingabedaten. CT-Aufnahmen sind dann von schlechter Qualität, wenn sie eine hohe Artefaktbelastung aufweisen. Unter Artefakten versteht man dabei solche Bestandteile der Bilder, die keine Entsprechung in der Verteilung der abzubildenden physikalischen Größe eines Objektes haben [9].

Vor allem die räumliche Auflösung der Aufnahmen ist hier von Bedeutung. Je größer das einem Voxel entsprechende Volumen, umso stärker wird der Einfluss sogenannter Partialvolumeneffekte. Die Folge ist, dass einzelne Voxel mitunter mehrere verschiedene Gewebearten repräsentieren und somit die Grenzen zwischen anatomischen Strukturen im CT-Bild verwischen. Für kleine Objekte, wie z.B. Lymphknoten, besteht zudem die Gefahr in der Aufnahme wesentlich kleiner oder sogar überhaupt nicht mehr differenziert dargestellt zu werden. Auch metallische Implantate (im HNO-Bereich vor allem Zahnfüllungen) führen zu Artefakten in den Aufnahmen. Sie absorbieren die Röntgenstrahlung stärker als das umliegende Gewebe und führen zu inkonsistenten Schwächungswerten. Dies zeigt sich im CT-Bild häufig als sternförmig von den Implantaten ausgehende Streifen, die das benachbarte Gewebe überlagern. Weitere Einflüsse, die sich negativ auf die Qualität der Aufnahmen auswirken, sind etwa Streustrahlung oder Bewegungen des Patienten. Die resultierenden Artefakte verschlechtern die Bedingungen für automatisierte Segmentierungsverfahren und erhöhen ebenso den Aufwand für etwaige Nutzereingriffe. So muss der Mediziner mehr Zeit für die genaue Betrachtung der Bilder aufwenden und benötigt viel Erfahrung, wenn sich die zu segmentierenden Objekte im CT-Bild nicht ohne Weiteres vom umliegenden Gewebe abgrenzen lassen.

2.2.2 Darstellung der Halsstrukturen in CT-Aufnahmen

In Abschnitt 2.1 wurden bereits die anatomischen Grundlagen für die, im Rahmen dieser Arbeit relevanten Strukturen des Halses behandelt. Für die Segmentierung in CT-Aufnahmen ist es besonders wichtig zu wissen, wie sich diese Strukturen in den Bildern überhaupt darstellen. Dies wird im Folgenden näher betrachtet.

Schildknorpel. Da der Schildknorpel aus sehr dichtem Gewebe besteht, ist er im CT-Bild gut von den umliegenden Weichteilstrukturen abzugrenzen. Der Grauwertbereich des Knorpels erinnert eher an den von Knochen und liegt zwischen etwa 1000 HE und 1500 HE. Allerdings erscheint nur die äußere Kontur derart hell. Das Innere dagegen, ist eher durch niedrige Grauwerte charakterisiert. Häufig kommt es auch zu starken Signalschwankungen, so dass die Kontur unterbrochen erscheint. In axialen Schichtbildern stellt sich die Form des Schildknorpels (von unten nach oben) wie folgt dar (Abb. 2.8). Zunächst sind die unteren Hörner als kleine, kreisförmige Gebilde zu erkennen. Dann werden zunehmend die Seitenplatten sichtbar, die schließlich miteinander verschmelzen, so dass der Schildknorpel in etwa die Gestalt eines nach unten geöffneten Hufeisens annimmt. Betrachtet man die höher gelegenen Schichten, sieht man wie sich die Seitenplatten wieder entzweien. Am oberen Ende des Schildknorpels befinden sich die oberen Hörner, die sich schließlich wieder als kreisförmige Gebilde darstellen.

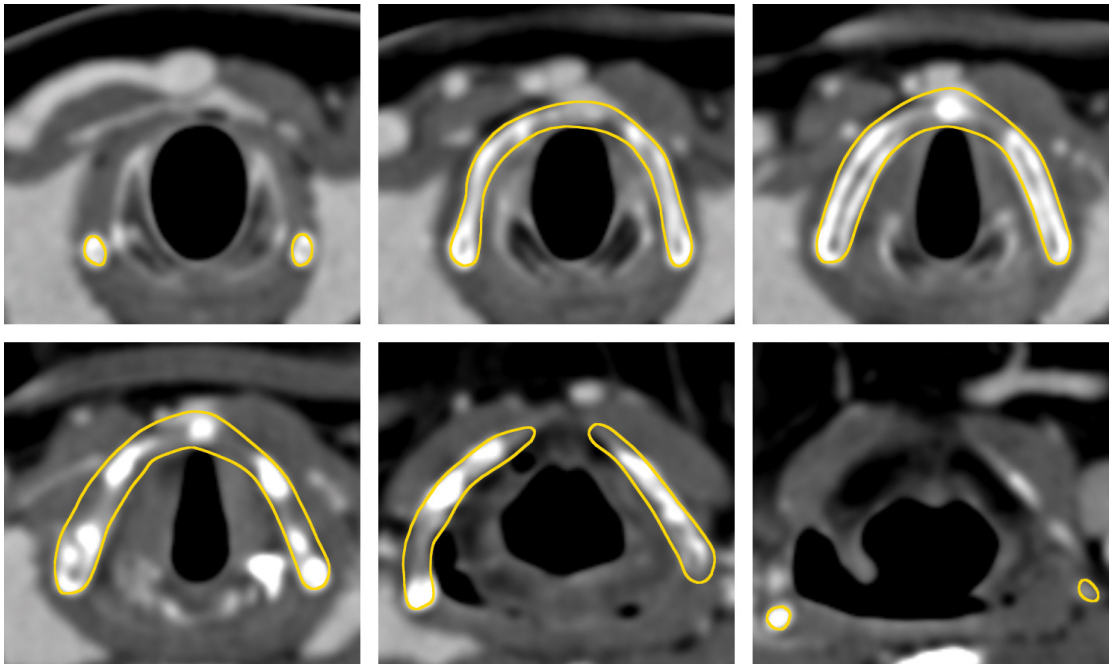


Abbildung 2.8: Schildknorpel im axialen CT-Bild.

Halslymphknoten. Wie bereits erwähnt, haben Lymphknoten in den meisten Fällen eine länglich ellipsoide bis kugelähnliche Form. In der Halsregion orientiert sich zudem die Ausrichtung ihrer Längsachse vorwiegend an parallel zur Körperachse verlaufenden Leitstrukturen, wie den großen Blutgefäßen. Daraus ergibt sich, dass Halslymphknoten in axialen CT-Schichten oftmals eine kreisförmige Gestalt annehmen. Der Grauwertbereich, in dem sich Lymphgewebe in CT-Bildern typischerweise abbildet, wurde in [18] erstmals näher untersucht. Dabei konnten die Werte auf ein Intervall zwischen -60 HE und 100 HE eingegrenzt werden. Innerhalb einzelner Datensätze wurden Schwankungen von bis zu 130 HE gemessen. (Abb. 2.9)

Eine eindeutige Identifizierung von Lymphknoten in CT-Datensätzen, nur auf Grundlage der genannten Eigenschaften, ist jedoch nicht möglich. Zu viele andere anatomische Strukturen des Halses stellen sich in den Schichtbildern ähnlich dar. Anders als diese, zeichnen sich Lymphknoten allerdings durch ihre geringe Ausdehnung in z-Richtung aus. Daher ist es sinnvoll, statt lediglich einzelne Schichtbilder isoliert zu betrachten, den gesamten Datensatz heranzuziehen. Lymphknoten lassen sich dann daran erkennen, dass sie immer nur in wenigen, aufeinander folgenden Schichten abgebildet sind. Um außerdem die Abgrenzung zu benachbarten Blutgefäßen zu erleichtern, kann durch Verabreichung von Kontrastmittel erreicht werden, dass sich diese wesentlich heller darstellen als das angrenzende Lymphgewebe.

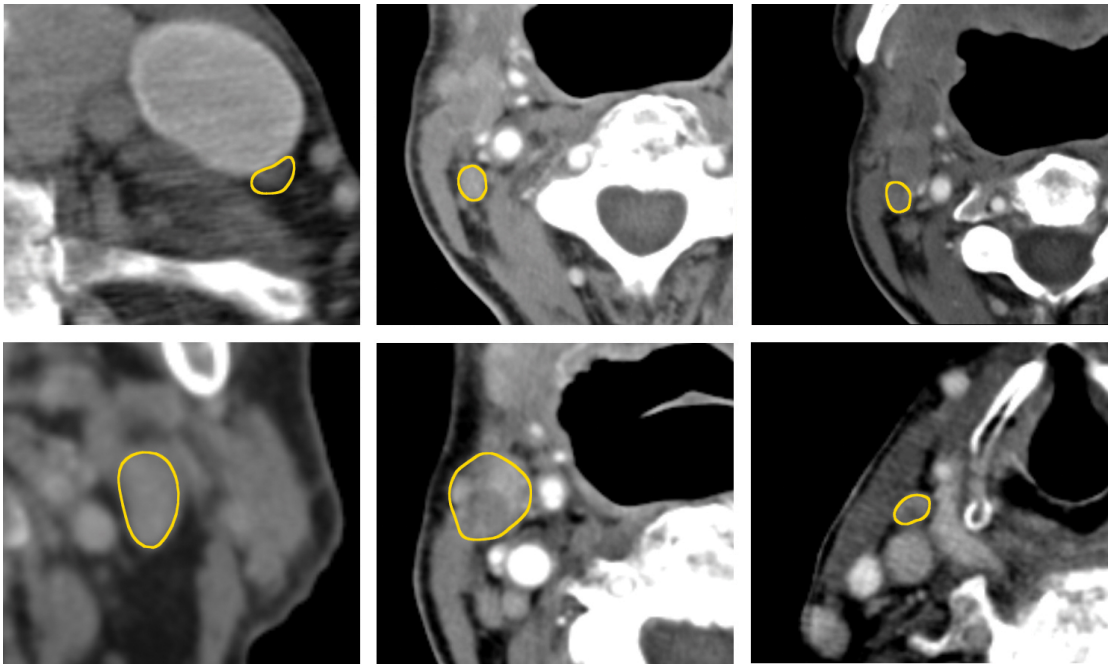


Abbildung 2.9: Verschiedene Lymphknoten im axialen CT-Bild.

Halstumoren. Da Tumoren durch unkontrolliertes Gewebewachstum entstehen, liegen sie in CT-Aufnahmen häufig im selben Grauwertbereich, wie auch das Gewebe, aus dem sie entwachsen. Zudem befindet sich gerade im Halsbereich eine Vielzahl von Weichteilstrukturen, die allesamt vergleichbare Grauwerte aufweisen. Auch Fettgewebe, das zwischen den Strukturen gelegen, sichtbare Grenzen im CT-Bild hervorrufen würde, findet sich dort kaum. Für die Segmentierung von Halstumoren in CT-Aufnahmen, müssen also zusätzliche Informationen herangezogen werden.

Tatsächlich ist die Computertomographie bei der Tumordiagnostik auch nicht die geeignetste Untersuchungsmethode. Primärtumoren werden im HNO-Bereich häufiger durch endoskopische Verfahren begutachtet [21]. Diese ermöglichen dem Mediziner eine direkte Betrachtung der pathologischen Strukturen mit Hilfe von Kameratechnik. Für eine dreidimensionale Visualisierung der Patientenanatomie zur Operationsplanung oder als Datenbasis für chirurgische Roboter-Assistenzsysteme, sind aber dennoch tomographische Aufnahmen und deren anschließende Segmentierung erforderlich. Dies geschieht in den meisten Fällen manuell, wobei der Mediziner auf das durch Endoskopien oder bereits intraoperativ gewonnene Wissen über die Gestalt des Tumors zurückgreift.

Abbildung 2.10 zeigt den Ausschnitt eines Hals-CTs, in dem ein Kehlkopfkarzinom diagnostiziert wurde. Gut zu erkennen ist, dass sich der Tumor nur minimal vom umliegenden Gewebe abgrenzt. Allerdings liefert die Form der Luftröhre einen ent-

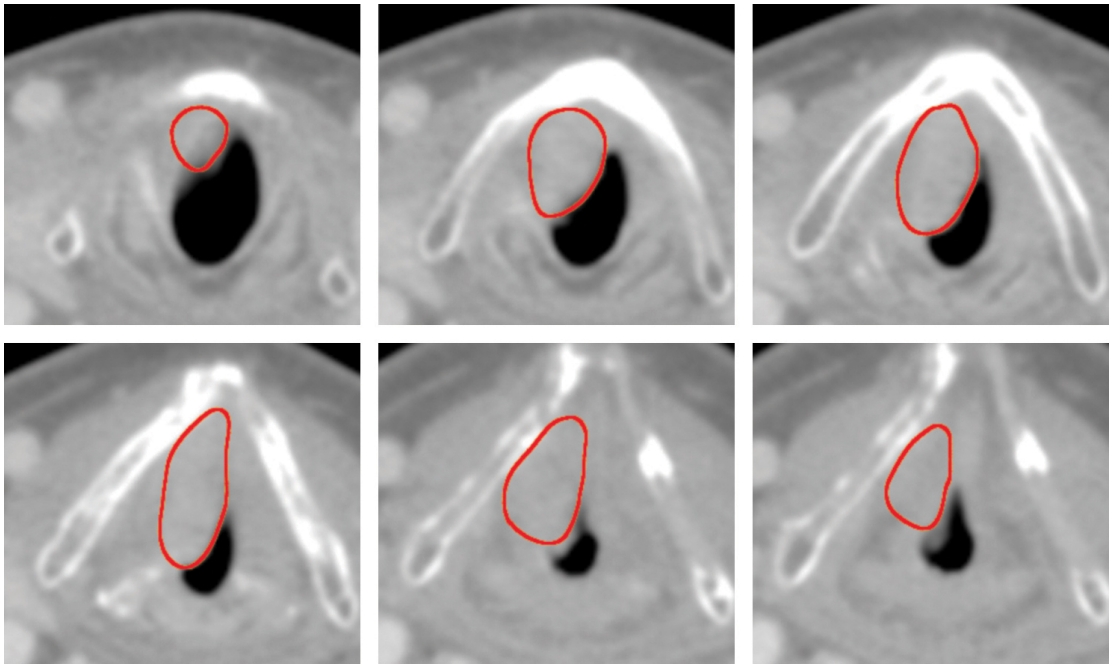


Abbildung 2.10: Kehlkopfkarzinom im axialen CT-Bild. Wie zu erkennen ist, unterscheidet sich der Tumor im Grauwertbereich nur minimal vom umliegenden Weichteilgewebe.

scheidenden Hinweis auf die Existenz und Gestalt des Karzinoms. Normalerweise als symmetrisches, schwarzes Oval im CT-Bild dargestellt, erkennt man deutlich, wie hier das Tumorwachstum bereits zur einseitigen Verengung der Luftröhre geführt hat. Der Schildknorpel dagegen erscheint unbeeinflusst, wodurch die Ausdehnung des Tumors auf der gegenüberliegenden Seite abgeschätzt werden kann.

2.3 Zusammenfassung

Die Entwicklung der Interaktionstechniken zur Segmentierungskorrektur wird in der vorliegenden Arbeit anhand von Beispielen aus der HNO-Heilkunde erläutert. Im Einzelnen werden dazu Segmentierungen von Halslymphknoten, dem Schildknorpel und verschiedener Tumoren aus dem Halsbereich verwendet. Zum besseren Verständnis wurden in diesem Kapitel deren anatomische Eigenschaften betrachtet. Außerdem wurde das Aufnahmeverfahren der Röntgen-Computertomographie erläutert und die Darstellung der genannten Strukturen in CT-Bildern geschildert. Insgesamt wurde deutlich, vor welch schwieriger Aufgabe automatisierte Segmentierungsverfahren in vielen Fällen stehen. Im nächsten Kapitel wird der Stand der Forschung in den für diese Arbeit relevanten Bereichen vorgestellt.

3 Stand der Forschung

In diesem Kapitel wird der aktuelle Stand der Forschung diskutiert. Zu Beginn folgt ein kurzer Exkurs in das Gebiet der Mensch-Computer-Interaktion. Hier wird auf die Terminologie rund um den Begriff der Interaktion und das Konzept der direkten Manipulation eingegangen. Anschließend werden Segmentierungsverfahren vorgestellt, die in der medizinischen Bildverarbeitung zum Einsatz kommen. Dabei soll jedoch kein Anspruch auf Vollständigkeit erhoben werden. Vielmehr liegt das Hauptaugenmerk auf den stabilen Feder-Masse-Modellen, die besonders für Segmentierungen im HNO-Bereich mit Erfolg eingesetzt werden. Danach wird anhand von Beispielen diskutiert, auf welche Weise Nutzerinteraktion in verschiedenen Segmentierungsverfahren realisiert ist. Im letzten Teil des Kapitels geht es schließlich um intuitive Ansätze zur Manipulation dreidimensionaler Modelle.

3.1 Mensch-Computer-Interaktion

Mensch-Computer-Interaktion (MCI) ist der Teilbereich der Informatik, der sich mit der Erforschung und Entwicklung interaktiver Systeme beschäftigt. Gegenstand der Untersuchungen sind also die Schnittstellen, über die Nutzer und System miteinander kommunizieren. Da neben dem Computer eben auch der Mensch im Mittelpunkt steht, handelt es sich um ein stark interdisziplinäres Gebiet, in das vor allem auch Erkenntnisse aus Psychologie, Ergonomie, Soziologie und Design einfließen.

Die technische Grundlage der Interaktion stellen Eingabegeräte dar. Sie ermöglichen es dem Nutzer, Daten und Informationen in das System einzugeben. Am stärksten verbreitet sind dabei, nach wie vor, Tastatur und Maus. Aber auch Graphiktabletts und vor allem berührungssensitive Bildschirme (Touch-Screens) werden immer häufiger verwendet. Für Virtual Reality Anwendungen etwa, ist der Datenhandschuh das Eingabegerät der Wahl.

Von diesen *physischen* Eingabegeräten, wird auf Programmebene allerdings abstrahiert und stattdessen mit *logischen* Eingabegeräten gearbeitet. So bleiben dem Programmierer unnötige, technische Details verborgen und Software kann ohne Änderungen im Programmcode, mit verschiedenen physischen Eingabegeräten be-

dient werden, sofern diese dasselbe logische Konzept realisieren. Ein Beispiel dafür ist das Zeigegerät. Als logisches Eingabegerät kann es als Positionsgeber verstanden werden. Physische Zeigegeräte sind z.B. Touch-Screens, bei denen eine Positionseingabe direkt auf dem Bildschirm vorgenommen wird, oder die Computermaus, deren Position indirekt auf den Bildschirm umgerechnet werden muss. Aufgrund der Abstraktion wird auf Applikationsebene in beiden Fällen lediglich eine Positionsänderung registriert, jedoch nicht *wie* diese zustande kommt. [52]

3.1.1 Terminologie

Um verschiedene Varianten von Nutzerinteraktionen zu untersuchen, existieren in der MCI einige grundlegende Begriffe, die im Folgenden näher erläutert werden sollen. In der Literatur tauchen dabei zum Teil unterschiedliche, bedeutungsgleiche Bezeichnungen auf. Für die vorliegende Arbeit wird die Terminologie verwendet, wie sie in [8] angelehnt an [20] eingeführt wird. Die begrifflichen Zusammenhänge sind in Abbildung 3.1 zusätzlich schematisch dargestellt.

- **Interaktionsaufgaben** sind die Aktionen, die ein Nutzer durchführen muss, um ein bestimmtes Interaktionsziel zu erreichen. Elementare Interaktionsaufgaben sind die Positionierung, die Auswahl, die Texteingabe und die Eingabe numerischer Werte. Durch verschiedene Kombinationen lassen sich diese zu komplexeren Aufgaben wie etwa der Konstruktion und Manipulation geometrischer Objekte zusammenfassen.
- **Interaktionsformen** werden die grundlegenden Arten der Informationseingabe in den Computer genannt. Man unterscheidet dabei grob zwischen sprachbasierten und graphikbasierten Interaktionsformen. Kommandosprachen sind beispielsweise sprachbasiert, während GUI-Komponenten, wie Buttons oder Schieberegler, graphikbasierte Interaktionsformen sind. Eine weitere wichtige graphikbasierte Form ist die direkte Manipulation, die gleich noch gesondert betrachtet wird.
- **Interaktionstechniken** sind die konkreten Möglichkeiten wie Eingabegeräte für Interaktionsaufgaben eingesetzt werden können. Sie werden nach der Interaktionsform klassifiziert die sie verwenden. Die Interaktionsaufgabe Konstruktion eines geometrischen Objektes kann z.B. mit der graphikbasierten, direkt manipulativen Interaktionstechnik *Rubberbanding* durchgeführt werden.
- **Interaktionsauswirkungen** bezeichnen die unmittelbaren Effekte einer Interaktion. Im besten Fall entsprechen diese dem Interaktionsziel des Nutzers, also dem, was dieser tatsächlich durch die Interaktion erreichen wollte. Das Gegenteil geschieht etwa dann, wenn dem Nutzer das Prinzip der angewandten Interaktionstechnik unklar ist und er deshalb die Konsequenzen falsch einschätzt.

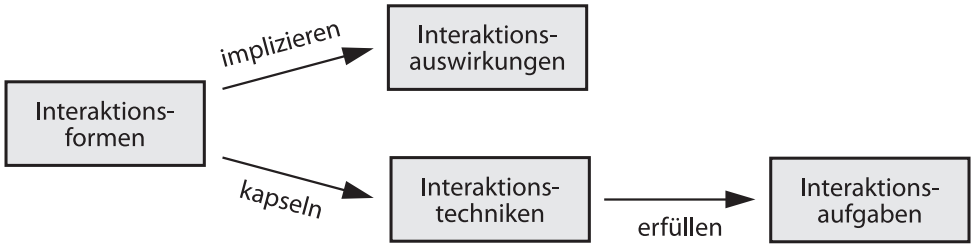


Abbildung 3.1: Terminologische Zusammenhänge der Interaktionsbegriffe nach [8].

3.1.2 Direkte Manipulation

Die direkte Manipulation ist eine spezielle Variante der graphikbasierten Interaktionsformen, bei der die Interaktionsaufgaben durch direkte Zeigehandlungen auf graphische Objekte realisiert werden. Nach Ben Shneiderman, einem Pionier auf dem Bereich der MCI, beruht die direkte Manipulation auf folgenden Hauptprinzipien [58]:

- permanente Sichtbarkeit der jeweils interessierenden Objekte
- schnelle, umkehrbare, einstufige Benutzeraktionen
- unmittelbare, visuelle Rückmeldung
- Ersetzung komplexer Kommandos durch physische Aktionen.

Die direkte Manipulation erlaubt es dem Nutzer auf besonders intuitive Weise mit der Software zu interagieren, da starke Ähnlichkeit zur Interaktion in der „realen“ Welt besteht. Shneiderman berichtet gar von enthuisiastischen Anwendern, die Freude an der Benutzung haben, die von der leichten Erlernbarkeit schwärmen und die das Gefühl haben, direkt manipulative Systeme vollständig zu beherrschen [58].

Demgegenüber ergeben sich bei dieser Interaktionsform allerdings auch einige Nachteile. So besticht die direkte Manipulation zwar durch ihre intuitive Bedienbarkeit, erlaubt aber lediglich elementare Operationen, so dass auch bei routinierter Anwendung kein Effizienzgewinn zu erwarten ist. Zudem sind exakte Manipulationen nur schwer zu realisieren und aus Entwicklersicht besteht ein wesentlich höherer Aufwand als bei anderen Interaktionsformen [52].

Die positiven Eigenschaften überwiegen jedoch vor allem für weniger computererfahrene und gelegentliche Nutzer. Gerade dann wenn Interaktionstechniken gefragt sind, die ohne Lernaufwand sofort einsetzbar sein sollen, ist die Akzeptanz für direkt manipulative Methoden besonders hoch. Für die Entwicklung der Interaktionstech-

niken zur Nachkorrektur von Segmentierungsergebnissen gilt es zudem, eine Verbesserung im Vergleich zur bisherigen, manuellen Korrekturmethode zu erreichen. Dass hierfür bislang trotz aller Nachteile auf eine Variante der direkten Manipulation zurückgegriffen wird, zeigt wie wichtig intuitive Bedienbarkeit für diese Anwendung ist. Für die Lösung der Aufgabenstellung dieser Arbeit bietet sich daher die Verwendung direkt manipulativer Techniken in besonderem Maße an.

3.2 Segmentierung medizinischer Bilddaten

Unter dem Begriff Segmentierung versteht man in der Bildverarbeitung den Vorgang, die atomaren Elemente eines Bildes[1] zu semantischen Einheiten zusammenzufassen. Wird dabei das gesamte Bild zerlegt, so dass jedes Element genau einer solchen Einheit zugeordnet wird, spricht man von vollständiger Segmentierung. In medizinischen Bildern bezieht sich die Semantik in der Regel auf anatomische bzw. funktionelle Eigenschaften, so dass die einzelnen Segmente unterschiedliche biologische Strukturen repräsentieren. Zudem wird für medizinische Anwendungen seltener eine vollständige Zerlegung, als vielmehr die gezielte Segmentierung einzelner, interessierender Objekte benötigt. So besteht eine häufige Aufgabe etwa darin, verdächtige Halslymphknoten in CT-Aufnahmen von Krebspatienten zu segmentieren. [51]

Die Segmentierung spielt im Verarbeitungsprozess medizinischer Bilddaten eine bedeutende Rolle, da sie wichtige Voraussetzung für weitere Anwendungen ist. So können von segmentierten Organen zum Beispiel dreidimensionale Modelle generiert und in interaktive Visualisierungen integriert werden. Dadurch lässt sich die Morphologie eines Patienten wesentlich besser darstellen und vom Mediziner begutachten. Sogar virtuelle Endoskopien werden auf diese Weise möglich. Darüber hinaus lassen sich mit Hilfe von Segmentierungen medizinischer Aufnahmen chirurgische Roboter-Assistenzsysteme steuern, mit denen Operationen in sensiblen Bereichen durchgeführt werden können. Die Vermessung von Organen und die möglichst automatische Diagnoseunterstützung sind weitere Anwendungen, die ohne vorherige Segmentierung der Bilddaten nicht zu realisieren wären.

Die Algorithmen, die zur Segmentierung verwendet werden[2], unterteilt man üblicherweise in *Low-Level* und *High-Level* Verfahren [51]. Für die Low-Level Segmentierung wird lediglich die Grauwertinformation der Bildpunkte herangezogen. Die wahrscheinlich einfachste Vorgehensweise ein Bild zu segmentieren, besteht dabei

[1] Pixel oder Voxel

[2] Hier erfolgt lediglich eine oberflächliche Beschreibung der Verfahren. Weitergehende Informationen zur Segmentierung im Allgemeinen sind z.B. [61] zu entnehmen. Die Segmentierung medizinischer Bilddaten wird ausführlich in [51] behandelt.

in der Setzung eines Schwellwertes. Dadurch wird das Bild in Regionen zerlegt, deren Grauwert entweder unterhalb oder oberhalb dieser Schwelle liegt. Der Schwellwert kann dabei manuell durch den Nutzer oder automatisch aus dem Histogramm des Bildes bestimmt werden. Anwendbar ist diese Methode allerdings nur dann, wenn sich die zu segmentierende Struktur auch tatsächlich durch ihren charakteristischen Grauwertbereich vom übrigen Bild separieren lässt.

Ein weiteres Low-Level Verfahren ist die Anwendung von Filtern zur Kantenextraktion. Kanten sind in der Sprache der Bildverarbeitung Diskontinuitäten in den Grauwerten benachbarter Bildelemente. Durch die Faltung eines Bildes mit geeigneten Filtermasken können die partiellen Ableitungen der Bildfunktion berechnet werden. Die Länge der Gradienten entspricht dabei der Stärke der Diskontinuitäten. Problematisch ist hierbei die Relevanz der so entdeckten Kanten, da beispielsweise auch durch Rauschen lokal starke Grauwertdifferenzen im Bild verursacht werden können, die keine Objektkontur repräsentieren. Außerdem setzt diese Art der Segmentierung voraus, dass die interessierenden Objekte durch geschlossene Kanten im Bild abgegrenzt sind.

In den allermeisten Fällen sind Low-Level Verfahren allein für die Segmentierung medizinischer Bilddaten nicht geeignet. Häufig ist bei diesen Aufgaben komplexeres Modellwissen über die betreffenden Objekte erforderlich, das über die einfache Verarbeitung von Grauwertinformation hinaus geht. Hier setzt die Idee der High-Level oder auch *modellbasiert* genannten Verfahren an. Derartige Segmentierungsmethoden verfügen über eine geeignete Repräsentation von apriori-Wissen von den zu segmentierenden Strukturen. Da für die erfolgreiche Segmentierung in der Regel sehr spezielles Modellwissen benötigt wird, werden die Verfahren gezielt für bestimmte Problemstellungen entwickelt, auf die ihre Anwendung dann beschränkt ist. Eine häufig zur Segmentierung medizinischer Bilddaten verwendete Klasse von High-Level Verfahren sind die deformierbaren Modelle, auf die im Folgenden näher eingegangen wird.

3.2.1 Deformierbare Modelle

Das Modellwissen deformierbarer Modelle besteht in erster Linie aus geometrischen bzw. analytischen Beschreibungen der gesuchten Objektform. Nach [2] können die deformierbaren Modelle in drei Kategorien unterteilt werden:

- **Statistische Modelle** modellieren die Formeigenschaften der Zielobjekte über statistische Verteilungen. Dies erfordert, dass im Vorfeld aus einer Menge von Trainingsdaten möglichst repräsentative Informationen über die Formvarianz gewonnen werden können.

- **Strukturelle Modelle** segmentieren Objekte durch Zerlegung in wesentlich einfacher zu erkennende, funktionelle Untereinheiten. Dies ist dann sinnvoll, wenn die Zielobjekte sehr komplex geformt und deren einzelne Teile stark in Lage, Rotation oder Skalierung variieren.
- **Dynamische Modelle** sind inspiriert durch physikalische Vorgänge der realen Welt. Dabei werden implizit oder explizit beschriebene geometrische Modelle im Bild platziert, die sich dem zu segmentierenden Objekt dynamisch anpassen.

3.2.2 Stabile Feder-Masse-Modelle

Der für diese Arbeit wichtigste Vertreter deformierbarer Modelle ist das stabile Feder-Masse-Modell (SMSM[3]). Dabei handelt es sich um ein physikalisch-basiertes, dynamisches Modell, das aus einer Toplogie von *Massen*, *Sensoren* und *Federn* aufgebaut ist. Die Massen sind untereinander durch die elastischen Federn verbunden, während die Sensoren an die Massen gekoppelt sind. Bilden die vernetzten Massen ein Dreiecksnetz, ergibt sich als zusätzliches Strukturmerkmal das Vorhandensein von Konturflächen. Daher kann die Geometrie eines SMSM als Polygonmodell (Abb. 3.2) aufgefasst werden.

Die Segmentierung mit SMSM erfolgt als bildbasierte Verformung durch die Einwirkung *interner* und *externer* Kräfte auf das Modell. Die externen Kräfte werden dabei durch Bildeigenschaften an den Sensorpositionen berechnet. Die internen Kräfte dagegen, ergeben sich aus den Feder- und Torsionskräften innerhalb des Modells. Durch die externen Kräfte wird also eine Verformung aufgrund von Low-Level Bildinformation bewirkt, während die internen Kräfte das Modell zusammenhalten und charakteristische Formeigenschaften kodieren. Auf diese Weise ist es möglich das Modell einem Datensatz anzupassen, wobei fehlende Bildinformation durch das Modellwissen überbrückt werden kann. Die dynamische Segmentierung kommt dann zum Stillstand, wenn ein (annäherndes) Gleichgewicht der wirkenden Kräfte erreicht wurde.[4]

Neben der ersten Einführung *stabiler* Feder-Masse-Modelle zur Segmentierung des linken Herzventrikels in SPECT-Daten [16], konnten SMSM bisher erfolgreich auf CT-Datensätzen des Halses angewandt werden. So sind auch die im Rahmen der vorliegenden Arbeit verwendeten Segmentierungen des Schildknorpels [15] und der Halslymphknoten [57] aus der Anwendung stabiler Feder-Masse-Modelle hervorgegangen. Darüber hinaus wurden SMSM zur Segmentierung von Halsblutgefäßen [38] und zur Segmentierung des Kopfwendemuskel [40] verwendet.

[3] von engl. stable mass-spring model

[4] Ausführlichere Erläuterungen zu stabilen Feder-Masse-Modellen sind in [16] zu finden.

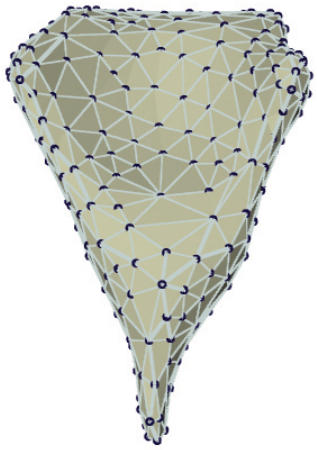


Abbildung 3.2: Stabiles Feder-Masse-Modell des Kehldeckels.

3.2.3 Interaktion in Segmentierungsverfahren

In diesem Abschnitt soll die Rolle der Interaktion in medizinischen Segmentierungsverfahren diskutiert werden. Grundsätzlich gibt es bei automatischen bzw. halbautomatischen Methoden drei verschiedene Zeitpunkte, an denen der Nutzer das Segmentierungsergebnis beeinflussen kann: Vor Beginn, währenddessen und schließlich nach Beendigung des Segmentierungsvorgangs sind interaktive Eingriffe denkbar. Bevor diese Situationen nun anhand von Beispielen näher untersucht werden, soll zunächst die rein manuelle Segmentierung betrachtet werden.

Manuelle Segmentierung. Aus algorithmischer Sicht ist die manuelle Methode die einfachste und zudem flexibelste Variante zur Segmentierung medizinischer Bilder. Prinzipiell wird hierbei dem Nutzer lediglich ein Werkzeug bereitgestellt, mit dem dieser dann selbständig den Umriss des gesuchten Objekts einzeichnet. Im einfachsten Fall geschieht dies dadurch, dass mit gedrückter Maustaste jeder einzelne Bildpunkt der Kontur markiert wird. Dieses Verfahren führt aber in der Regel zu unbefriedigenden Ergebnissen, da die Eingabe mit der Maus sehr ungenau ist und sich bereits leichtes Zittern negativ auf das Resultat auswirkt. Besser ist es, die Eingabe auf nur einige wenige Punkte zu beschränken und die vollständige Kontur linear zu interpolieren. Optisch ansprechendere, glattere Kurven können durch die die Verwendung von Spline-Interpolation erreicht werden. Im Vergleich zur linearen Interpolation lässt sich oftmals die Anzahl der einzugebenden Stützpunkte noch verringern und darüber hinaus ein natürlicher wirkendes Ergebnis erzielen [51].

Da der Software bei der manuellen Methode kein Wissen über die eigentliche Segmentierungsaufgabe zur Verfügung steht, ist es auch die für den Nutzer bei Weitem aufwändigste Methode. So wird beispielsweise in [53] beschrieben, dass für die manuelle Segmentierung der Nasennebenhöhlen in einem hochaufgelösten CT-Datensatz ganze acht bis zehn Stunden aufgewandt werden müssen. Ein weiterer Nachteil ergibt sich auch daraus, dass bei diesem Verfahren keine objektiven Ergebnisse produziert werden können. Kaus *et al.* berichten in [32] von bis zu 22% Unterschied im Voxelvolumen bei der Segmentierung ein und desselben Hirntumors durch verschiedene Experten.

Insbesondere dann aber, wenn keine befriedigenden automatisierten Lösungen verfügbar sind oder erst eine Vielzahl komplizierter, abstrakter Parametereinstellungen vorgenommen werden müssen, kommt die manuelle Segmentierung noch häufig zum Einsatz. Auch für die Vorbereitung von Operationen mit chirurgischen Roboter-Assistenzsystemen wird eher auf die Expertise erfahrener Mediziner bei der Erstellung kritischer Segmentierungen vertraut. Spätestens aber zur nachträglichen Korrektur automatisch erzeugter Segmentierungsergebnisse wird in der Regel auf die manuelle Methode zurückgegriffen. Trotz seiner Defizite zeichnet sich dieses Verfahren nämlich durch eine außerordentlich intuitive Bedienbarkeit aus. Darüber hinaus gestattet es dem Nutzer uneingeschränkt, sämtliches Expertenwissen in den Segmentierungsprozess einzubringen. Dies ist etwa dann von Vorteil, wenn dem Mediziner zusätzliche Informationen über die individuelle Morphologie[5] vorliegen, die dem Computerprogramm nicht zugänglich sind.

Initialisierung. Da es sich bei der Segmentierung medizinischer Bilddaten um sehr komplexe Problemstellungen handelt, muss ein automatisiertes Segmentierungsverfahren oftmals vom Nutzer der konkreten Situation entsprechend initialisiert werden. Das kann beispielsweise wie beim *Connected Thresholding* [17] durch die Eingabe von oberer und unterer Grenze eines Grauwertbereichs geschehen. Am häufigsten aber werden vom Nutzer Informationen über die Lage des gesuchten Objektes im Datensatz verlangt. Dies ist etwa beim klassischen *Region Growing* der Fall, für das ein Saatpunkt im Inneren der gesuchten Struktur bestimmt werden muss [1]. Um Übersegmentierung zu verhindern, wird bei der Segmentierung von Lymphknoten mittels *Fast Marching* [64] zur Begrenzung des Suchraumes zusätzlich die Definition eines Kreises um den Lymphknoten benötigt.

Vor allem deformierbare Modelle müssen zur Initialisierung häufig manuell in der Nähe des gesuchten Objektes platziert werden. Um zweidimensionale Daten mittels *Snakes* [31] zu segmentieren, muss der Nutzer zunächst ein grobes Hüllpolygon um

[5] z.B. aus einer vorangegangenen Endoskopie

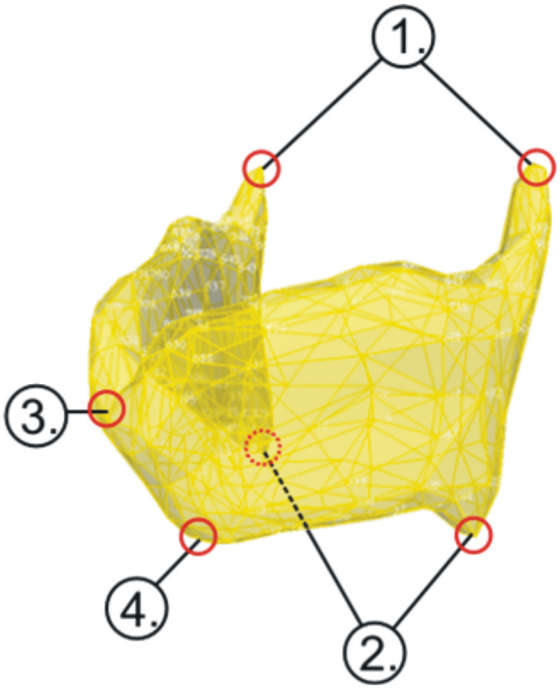


Abbildung 3.3: Die Initialisierung der SMSM Segmentierung des Schildknorpels erfolgt durch die Positionierung von sechs Landmarken im Datensatz. Quelle: [15]

das Zielobjekt einzeichnen. Für dreidimensionale Daten bietet sich etwa die Verwendung von SMSM an. Bei dieser Variante zur Segmentierung von Lymphknoten ist es notwendig einen Punkt im Inneren oder besser noch zwei gegenüberliegende Punkte der Kontur zu bestimmen [57]. Auf ähnliche Weise erfolgt die Initialisierung der Segmentierung von Halsblutgefäßen in [38]. Hier besteht die Aufgabe des Nutzers darin die Lage des gesuchten Blutgefäßes, einschließlich eventueller Verzweigungen, in einigen Schichten durch möglichst mittig gelegene Punkte anzugeben. Für die Schildknorpel-Segmentierung mittels SMSM [15] muss das Modell anhand der Positionierung sechs markanter Objektpunkte initialisiert werden (Abb. 3.3). In [24] werden zur Segmentierung des muskoloskeletalen Systems gar zehn solcher Landmarken benötigt. Bei der Segmentierung des Kopfwendemuskels mit Hilfe von SMSM [40] initialisiert der Nutzer den Vorgang, indem er in einigen Schichten des Datensatzes durch Einzeichnen eines Striches die Mittelachse des Muskels approximiert. Ähnlich verläuft auch die Initialisierung von *SketchSurfaces* [5], einem halbautomatischen Verfahren für die Segmentierung beliebiger Strukturen in medizinischen Datensätzen. Hier muss der Nutzer zunächst die Ausdehnung des Zielobjektes grob, mit wenigen Strichen skizzieren. Dadurch wird bereits eine gute Approximation der Objektkontur erreicht, mit der die automatische Anpassung des so definierten Oberflächenmodells initialisiert wird.

Steuerung. Neben der Initialisierung ist bei vielen Verfahren auch ein Eingreifen des Nutzers während des Segmentierungsprozesses erforderlich. Im Gegensatz zur automatischen Segmentierung nennt man solche Verfahren interaktiv oder halbautomatisch. Ein klassisches Beispiel dafür ist *Live-Wire* [7]. Bei diesem Verfahren werden zweidimensionale Bilder als Graphen interpretiert. Die Pixel werden dazu als Knoten aufgefasst, die über (in Abhängigkeit vom Grauwert) gewichtete Kanten mit den Bildpunkten ihrer Nachbarschaft verbunden sind. Zur Initialisierung des Verfahrens bestimmt der Nutzer einen Startpunkt auf der Kontur der gesuchten Struktur. Bewegt er nun die Maus, wird mit Hilfe des Dijkstra-Algorithmus [13] automatisch der günstigste Pfad zwischen Startpunkt und aktueller Cursor-Position berechnet und im Bild angezeigt. Durch wiederholte Mausklicks wird die jeweils aktuelle Position als Ausgangspunkt für die nächste Pfadberechnung verwendet und so schließlich die gesamte Kontur segmentiert. Eine ähnliche Methode wird in [36] beschrieben, wobei hier jedoch der Pfad zwischen zwei Punkten mit Hilfe von aktiven Konturen generiert wird. Andere Variationen sind etwa *Live-Lane* [19] oder *Phase-Wire* [47]. Die Interaktionstechnik bei all diesen Verfahren ist vergleichbar mit der interpolierten, manuellen Segmentierung, jedoch mit weniger Aufwand verbunden. Außerdem wird die Kontur hier auf Grundlage von Kanteninformation des Bildes berechnet. Daher ist bereits geringes Modellwissen in den Algorithmus integriert, wodurch ein wesentlich exakteres und objektiveres Ergebnis als bei manuellen Segmentierungen erzielt werden kann. Erfolgversprechend ist die Anwendung von *Live-Wire* und dessen Abwandlungen allerdings nur dann, wenn sich das zu segmentierende Objekt aufgrund seiner Kanten im Bild von der Umgebung abgrenzen lässt.

Bei den oben genannten Verfahren ist die Steuerung durch den Nutzer zwingend erforderlich. Zwar wird ein Großteil der Segmentierung automatisch generiert, jedoch sind die Verfahren ohne Eingriffe des Anwenders nicht lauffähig. Demgegenüber existiert eine Reihe von Algorithmen die grundsätzlich automatisch ablaufen, aber dem Nutzer Optionen zur Verfügung stellen, das Segmentierungsergebnis entscheidend zu beeinflussen. So wie etwa bei der modellbasierten Segmentierung mit *Snakes* [31]. Über die Initialisierung hinaus, hat der Nutzer hier die Möglichkeit mit sogenannten *Springs* und *Volcanoes* den Segmentierungsprozess zu unterstützen. Damit werden zusätzlich zu den, aus dem Bild abgeleiteten, weitere externe Kräfte eingeführt, mit denen die Kurve von einer Bildregion angezogen (*Springs*) oder davon abgestoßen (*Volcanoes*) wird. Mit Springs ist es außerdem möglich zwei Kurven lokal zusammen zuhalten. Die Nutzereingriffe werden dadurch realisiert, dass in den gewünschten Bildregionen entsprechende grafische Symbole platziert werden (Abb. 3.4). Zwar ist diese Interaktionstechnik leicht verständlich, jedoch ist es mitunter schwierig das genaue Verhalten der Snakes nach solchen Eingriffen vorauszusagen.

Bei der Segmentierung mit Hilfe von *Balloons* [12], der dreidimensionalen Erweiterung der Snakes, kann ebenfalls eine nutzergenerierte Kraft angewandt werden. So wird etwa in [35] beschrieben, wie durch direkte Manipulation des Modells Knoten

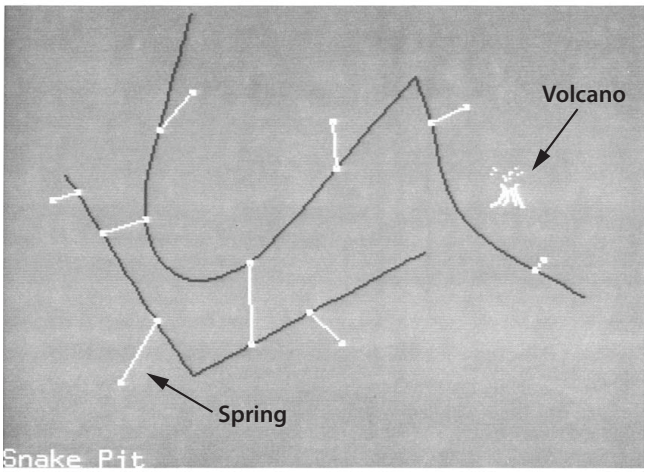


Abbildung 3.4: Interaktive Steuerung der Segmentierung mit *Snakes*. Mit *Springs* wird die Kurve von einer Bildregion (oder einer anderen *Snake*) angezogen, mit *Volcanoes* dagegen, kann sie davon abgestoßen werden. Quelle: [31]

an beliebige Positionen verschoben und dort verankert werden können. Der Nutzer hat so die Möglichkeit eine fehlerhafte Segmentierung zu unterbinden, indem er den Deformierungsprozess in die gewünschte Richtung lenkt. Außerdem ist es möglich, die lokalen Parameter einzelner Knoten zu verändern. Da dies während des laufenden Segmentierungsvorgangs geschieht, passt sich das *Ballon*-Modell sofort entsprechend der veränderten Situation an.

Bei *SketchSurfaces* [5] kann der Nutzer ebenfalls durch geometrische Operationen in den Segmentierungsprozess eingreifen. Auch hier besteht die Möglichkeit, Kontrollpunkte der Kontur (sogar in beliebig orientierten Ebenen des Datensatzes) zu verschieben und zu fixieren. Darüber hinaus können Teile des Modells abgeschnitten und lokale Verfeinerungen der Konturen bewirkt werden.

In allen oben zitierten Aufsätzen werden in erster Linie die jeweiligen Algorithmen erläutert, die zur Steuerung verwendeten Interaktionstechniken dagegen meist nur am Rande diskutiert. Allerdings erschien mit der Dissertation von Olabarriaga [48] auch eine Arbeit, die sich ausschließlich mit der Interaktion im Bereich der medizinischen Segmentierung befasst. Olabarriaga stellt darin fest, dass es gegenwärtig und auch in naher Zukunft, keine vollständig verlässlichen, automatischen Segmentierungsverfahren für medizinische Bilddaten geben wird; es existieren immer Fälle, in denen der Nutzer korrigierend eingreifen muss. Im Sinne der Zuverlässigkeit und Reproduzierbarkeit wird daher für den ausschließlichen Einsatz interaktiv gesteuerter Segmentierungsmethoden plädiert. Allerdings wird auch kritisiert, dass bei den bestehenden Verfahren, ohne die vorherige Untersuchung genereller Aspekte, lediglich auf den Einzelfall zugeschnittene Interaktionstechniken zum Einsatz kommen.

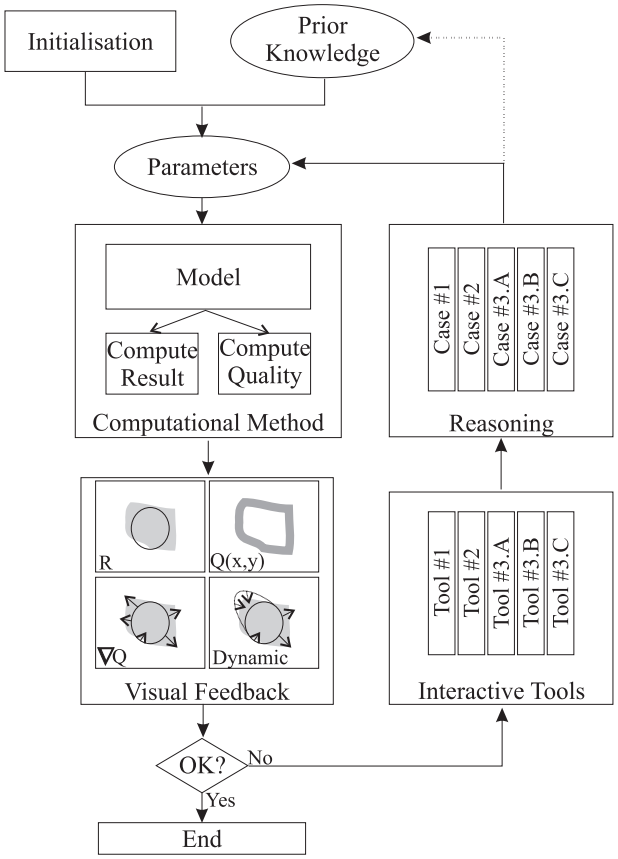


Abbildung 3.5: *Framework for Interactive Segmentation.* Schematische Darstellung. Quelle: [48].

Um dem zu begegnen, wurde von Olabarriaga das *Framework for Interactive Segmentation* (FIS) entwickelt. Die grundlegende Idee besteht dabei darin, automatische Segmentierung und Nutzerinteraktion derart in einem Prozess zu integrieren, dass unzureichendes, algorithmisches Modellwissen effizient vom Anwender vervollständigt werden kann (Abb. 3.5). Das Framework basiert auf der Annahme, dass unabhängig von der konkreten Problemstellung (mindestens) fünf generelle Situationen benannt werden können, in denen eine automatische Segmentierung versagt. Daher sollten nach Olabarriaga spezielle, fallbasierte Interaktionstechniken angeboten werden, mit denen je nach Situation lokale Anpassungen am Segmentierungsmodell vorgenommen werden können. Für den Fall, dass zum Beispiel lokal nur schwache Kanteninformation im Bild vorhanden ist[6], muss es ein Werkzeug geben,

[6] „*low or absent visual evidence*“

dass es ermöglicht den korrekten Verlauf der Kante einzugeben. Damit der Nutzer entscheiden kann, welches Werkzeug anzuwenden ist, muss er in der Lage sein, zu erkennen warum das Modell scheitert. Um dies zu ermöglichen, bedarf es eines Qualitätsmaßes, mit dem die Ähnlichkeit zwischen momentanem Zwischenergebnis und idealisierter Modellannahme berechnet und dem Nutzer angezeigt werden kann. Neben fallbasierten Interaktionstechniken ist im Rahmen des FIS also auch geeignetes visuelles Feedback erforderlich, das über das Einblenden der aktuellen Segmentierung hinausgeht.

Wie Olabarriaga selbst feststellt, ist es jedoch nicht ohne Weiteres möglich, bestehende interaktive Segmentierungsverfahren im Sinne ihres Framework anzupassen. In [48] stellt sie deshalb eine neu entwickelte, modellbasierte Methode vor, die vom FIS Gebrauch macht. Damit ist es effizient möglich verlässliche, reproduzierbare und akkurate Segmentierungen des Sprunggelenkspaltes in planaren Röntgenaufnahmen zu gewinnen. Bei dieser Anwendung handelt es sich allerdings um ein zweidimensionales Problem. Eine Segmentierung dreidimensionaler Datensätze wird von Olabarriaga nicht diskutiert. Der Segmentierungsverlauf sowie die Visualisierungen von Qualitätsfunktion und Modellannahmen lassen sich im Zweidimensionalen problemlos vom Nutzer überblicken. Im dreidimensionalen Fall wird dies jedoch wesentlich schwerer zu bewältigen sein. Zudem ist unklar, ob sich dort gegenüber der manuellen Nachkorrektur tatsächlich ein Zeitgewinn erzielen lässt - im klinischen Alltag ein wichtiger Faktor für die Akzeptanz eines Segmentierungsverfahrens. Richtig ist, dass die Ergebnisse automatisierter Methoden stets überprüft und nicht selten auch korrigiert werden müssen. Dennoch haben sich viele solcher Verfahren, wie zum Beispiel bei der Anwendung von SMSM im HNO-Bereich, bereits bewährt. So erscheint es entgegen Olabarriagas Argumentation dennoch sinnvoll, verbesserte Interaktionstechniken für die nachträgliche Korrektur automatischer Segmentierungsverfahren zu entwickeln.

Nachkorrektur. Nach Beendigung eines automatisierten Segmentierungsvorganges kann eine Korrektur der Ergebnisse auf zweierlei Arten geschehen. Zum einen ist es möglich, das Verfahren neu zu initialisieren um es dann wiederholt auszuführen. So kann beispielsweise ein Schwellwert verändert oder ein Saatpunkt neu positioniert werden. Tatsächlich geeignet ist dieses Vorgehen allerdings nur bei der Verwendung sehr schneller Segmentierungsverfahren oder dann, wenn das Ergebnis aufgrund ungünstiger initialer Parametrisierung als besonders fehlerhaft einzuschätzen und durch den Neustart eine wesentliche Verbesserung zu erwarten ist. Der Vorteil dabei liegt darin, dass der subjektive Einfluss auf das Segmentierungsergebnis verringert werden kann und der automatische Vorgang unter Umständen sogar weniger Zeit beansprucht, als eine vollständige, manuelle Nachkorrektur.

Kommt das erneute Ausführen des gesamten Vorgangs nicht in Frage, bleibt die Möglichkeit der (unabhängig vom verwendeten Segmentierungsverfahren ausfallenden) Nachkorrektur der Ergebnisse. In den meisten Artikeln über automatische Segmentierungsmethoden wird auf diesen Schritt jedoch nicht näher eingegangen.

Ein Vorschlag zur Korrektur von Lymphknotensegmentierungen mittels SMSM findet sich aber in [57]. Hier wird ein Vorgehen beschrieben, das teilweise von den Modellannahmen des Segmentierungsverfahrens Gebrauch macht. Bei der dort vorgeschlagenen Interaktion muss der Nutzer, im Bereich einer Fehlsegmentierung, lediglich auf einen Punkt der tatsächlichen Objektkontur klicken. Die nächstgelegene Masse wird dorthin verschoben und fixiert, worauf sich das Modell aufgrund seiner internen Federkräfte sofort anpasst.

Auch in [54] findet sich ein Vorschlag zur nachträglichen Korrektur. Für die Nachbearbeitung von *Region Growing* basierten Segmentierungen des im Innenohr gelegenen Warzenfortsatzes, wird dort eine *Live-Wire*-Variante verwendet. Die Besonderheit besteht hierbei darin, dass die Interaktion nur in wenigen axialen Schichten stattfindet, zwischen denen die Korrektur interpoliert wird. Dies ist dann erfolgversprechend und führt zur Reduktion des Interaktionsaufwands, wenn Datensätze mit hoher Auflösung in z-Richtung zur Verfügung stehen. In diesem Fall hält die Annahme stand, dass sich die Objektkontur in benachbarten Schichten nur minimal unterscheidet. Außerdem ist für die Anwendung dieser Methode erforderlich, dass sich die Zielstruktur über Kanten im Bild von der Umgebung abgrenzen lässt.

Abgesehen von den genannten Beispielen wird die Nachkorrektur automatischer Segmentierungsergebnisse in der Regel durch einfache, manuelle Bearbeitung der Kontur in jeder Schicht eines Datensatzes vorgenommen [51]. Auf diese Weise wird auch von den Medizinern der HNO-Universitätsklink Leipzig, unserem klinischen Partner, bei der Korrektur von SMSM-basierten Segmentierungen verfahren. Da der Suche nach alternativen Interaktionstechniken zur manuellen Nachkorrektur bisher nur wenig Aufmerksamkeit zuteil wurde - hier aber dennoch Bedarf besteht - zeigt sich erneut die Notwendigkeit für Neuentwicklungen in diesem Bereich.

3.3 Manipulation von Polygonmodellen

Da im Rahmen dieser Arbeit die Segmentierung in dreidimensionalen Datensätzen betrachtet wird, liegt es nahe, die Zielobjekte auch als dreidimensionale Körper aufzufassen. Wie gezeigt wurde, geschieht genau dies bei der Verwendung von SMSM. Die Segmentierungsergebnisse liegen in diesem Fall in Form dreidimensionaler Polygonmodelle vor. Aber auch anderweitig gewonnene Resultate, wie etwa bei der manuellen Tumorsegmentierung, können in eine solche Repräsentation überführt werden.

Um dem dreidimensionalen Charakter der Objekte gerecht zu werden, kann die Nachkorrektur der Segmentierungen sinnvoller Weise als Manipulation polygonaler Oberflächenmodelle verstanden werden. Im Anschluss folgt deshalb ein Überblick über den Stand der Forschung im Bereich intuitiver Polygonmodellierung. Da in diesem Zusammenhang das *Laplacian Modeling Framework* eine wichtige Rolle spielt und auch im weiteren Verlauf dieser Arbeit zur Anwendung kommt, wird es im letzten Abschnitt ausführlicher erläutert.

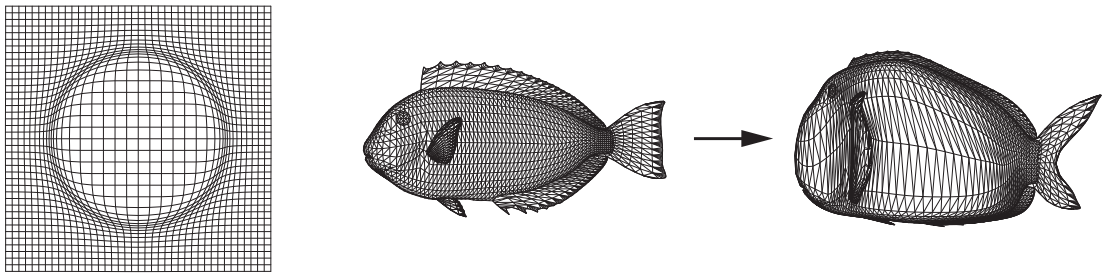
3.3.1 Intuitive Interaktionstechniken

Die Manipulation von Polygonmodellen ist eine klassische Aufgabe in den Bereichen Konstruktion und Industriedesign sowie bei der Herstellung von Kinofilmen und Computerspielen. Die Anwender sind üblicherweise Experten auf dem Gebiet der 3D-Modellierung und besitzen daher entsprechend fortgeschrittene technische Fertigkeiten. In der Regel wird von den verwendeten Softwareprogrammen eine Vielzahl unterschiedlicher Deformierungs- und Modellierungswerkzeuge zur Verfügung gestellt.

Besonders verbreitet ist beispielsweise die Verwendung der *Free-Form Deformation* (FFD) [6, 56]. Dabei werden die zu deformierenden Objekte in den sogenannten FFD-Raum eingebettet, der sich aus trivariaten Béziervolumen (*Hyperpatches*) zusammensetzt. Jedes dieser Subvolumen ist durch 64 Kontrollpunkte definiert, die zusammen eine, das Polygonmodell umhüllende, Gitterstruktur ergeben. Bezüglich dieses Gitters ist jedes Modellvertex durch eine eindeutig, parametrisch beschreibbare Position repräsentiert. Die Deformierung wird nun indirekt, über das Hüllgitter vorgenommen. Dazu werden ein oder mehrere der Kontrollpunkte als *Handle*[7] definiert, und mit der Maus bewegt. Das Resultat ist eine veränderte Vertex-Parametrisierung, aus der dann die Deformation des Polygonmodells berechnet wird. Die Verwendung von *Handles* ist ein sehr eingängiges Konzept, da es an das direkte Verformen des Objektes mit den eigenen Händen erinnert.

FFD-basierte Interaktionstechniken sind mächtige Werkzeuge, die vor allem eine präzise Manipulation der Modellgeometrie erlauben. Der große Nachteil besteht jedoch in der oftmals schweren Vorhersagbarkeit der Interaktionsauswirkung. Für die gezielte Nutzung ist in der Regel ein tieferes Verständnis der zugrunde liegenden Parametrisierung vonnöten, weshalb der Anwenderkreis eher auf erfahrene Nutzer beschränkt ist. Um Anfängern und auch Gelegenheitsnutzern den Zugang zur Polygonmodellierung zu erleichtern, wird seit einigen Jahren gezielt daran geforscht, intuitivere Interaktionstechniken zu entwickeln.

[7] engl. Griff

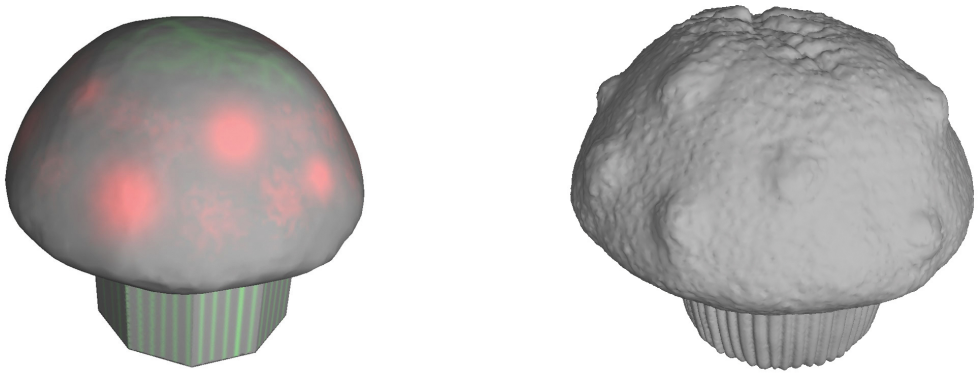


(a) (b)

Abbildung 3.6: Lupenwerkzeug zur Manipulation von Polygonmodellen. (a) Schematische Darstellung des Verformungseffektes, (b) Ergebnis der sukzessiven Anwendung des Werkzeugs auf ein 3D-Modell. Quelle: [46]

Ein Vorschlag dazu, der von Neumann *et al.* in [46] berichtet wird, basiert auf der Verwendung eines lupenähnlichen Werkzeugs. Hauptsächlich für die Anwendung auf zweidimensionalen Strichzeichnungen entwickelt, wird in dem Aufsatz jedoch auch ein möglicher Einsatz zur Manipulation von Polygonmodellen beschrieben. Basierend auf dem *Elastic Presentation Framework* [10] wurde eine Interaktionstechnik implementiert, mit der lokale Deformierungen an Polygonmodellen erreicht werden können. Das Deformationsverhalten erinnert dabei an die Verwendung eines Vergrößerungsglases und kann durch Parameter wie die fokale Länge und den Vegrößerungsfaktor beeinflusst werden. Abbildung 3.6a zeigt eine schematische Darstellung des Lupeneffektes, der dadurch entsteht, dass Vertices innerhalb des Einflussbereiches, in entgegengesetzter Richtung zur *viewing normal*, näher an den Betrachterstandpunkt bewegt werden. In Abbildung 3.6b ist das Resultat der sukzessiven Anwendung des Werkzeugs auf ein Polygonmodell zu sehen. Die erreichbaren Ergebnisse sind relativ beschränkt. Für den Einsatz dieser Interaktionstechnik spricht allerdings, dass sie insbesondere von ungeübten Nutzern ohne großen Lernaufwand angewandt werden kann.

Eine gänzlich andere Herangehensweise stammt von Lawrence und Funkhouser [37]. Hier wird die Deformation durch die Simulation physikalischer Prozesse (ähnlich den deformierbaren Modellen zur Segmentierung) erreicht. Das Besondere bei diesem Ansatz ist die Art und Weise der interaktiven Steuerung des Deformationsprozesses. Die Einstellung der Parameter wird durch Verwendung einer Farbpinsel-Metapher lokal am Polygonmodell vorgenommen. Durch die Wahl der Farbe bestimmt der Nutzer den Wert der Geschwindigkeit und die Richtung der Oberflächenverformung. Mit der Pinselform wird die räumliche Verteilung der Parametrisierung auf dem Modell beeinflusst. In Abbildung 3.7 ist das Vorgehen illustriert. Zunächst übermalt der Nutzer die zu deformierenden Bereiche des Modells in der gewünschten Farbe (Abb. 3.7a). Rottöne führen zur „Aufblähung“ in Richtung der Oberflächennormale,



(a) (b)

Abbildung 3.7: Farbpinsel-Metapher zur Manipulation von Polygonmodellen. (a) Lokal eingefärbtes Ausgangsmodell, (b) Ergebnis der Deformierung. Quelle: [37]

Grüntöne bewirken eine entgegengesetzte Verformung. Anschließend erfolgt die eigentliche Deformierung, indem der Simulationsprozess solange schrittweise ausgeführt wird, bis das gewünschte Ergebnis erreicht ist (Abb. 3.7b). Zwar sind auch bei diesem Ansatz die Möglichkeiten zur Modellmanipulation begrenzt, jedoch wird berichtet, dass sich besonders organische Objekte gut modellieren lassen. Dies würde für den Einsatz im medizinischen Bereich sprechen. Allerdings ist zu erwarten, dass eine direkte Verformung der Modelle von unerfahrenen Nutzern eher angenommen wird, als der Umweg über die physikalisch simulierte Deformation.

Einen besonders beliebten Ansatz zur intuitiven Polygonmodellierung stellt die Verwendung skizzenbasierter Verfahren dar. Auf diesem Bereich wird besonders ausgiebig geforscht und so existiert bereits eine Vielzahl relevanter Veröffentlichungen. Eine detaillierte Übersicht über den Stand der Forschung in diesem Spezialgebiet, mit weit über einhundert zitierten Aufsätzen, findet sich in [49]. Da eine genauere Betrachtung den Rahmen sprengen würde, werden an dieser Stelle nur einige wichtige Vertreter genannt.

Als Vorreiter im Bereich der skizzenbasierten 3D-Modellierung ist *SKETCH* von Zeleznik *et al.* [66] zu nennen. Hier wurde erstmals zweidimensionale Skizzierung als Interaktionstechnik zur dreidimensionalen Modellierung verwendet. Der Nutzer von *SKETCH* skizziert charakteristische Merkmale geometrischer Primitive, die daraufhin vom System erkannt, instantiiert und in der 3D-Szene platziert werden. Einige Jahre darauf wurde *Teddy* von Igarashi *et al.* [29] entwickelt, das schließlich zu *SmoothTeddy* [28] erweitert wurde und als Grundlage für die Entwicklung von *FiberMesh* [43] diente. Bei *Teddy* werden zweidimensionale Silhouetten gezeichnet, die dann automatisch zu dreidimensionalen Freiformobjekten „aufgebläht“ werden. Eine interessante Option ist zudem, bestehende Modelle zu manipulieren, indem ganz

einfach deren Umrisse überzeichnet werden. Diese Interaktionstechnik erinnert an die manuelle Nachkorrektur von Segmentierungsergebnissen. Im Unterschied dazu, resultiert die Interaktion bei *Teddy* allerdings in einer dreidimensionalen Deformation (Abb. 3.8). Für den zeitraubenden Prozess der Nachkorrektur dreidimensionaler Segmentierungen, könnte ein ähnlicher Ansatz zur Verminderung des Interaktionsaufwands sinnvoll eingesetzt werden.

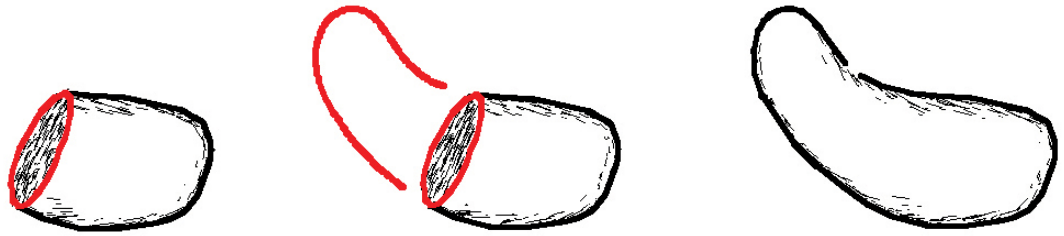
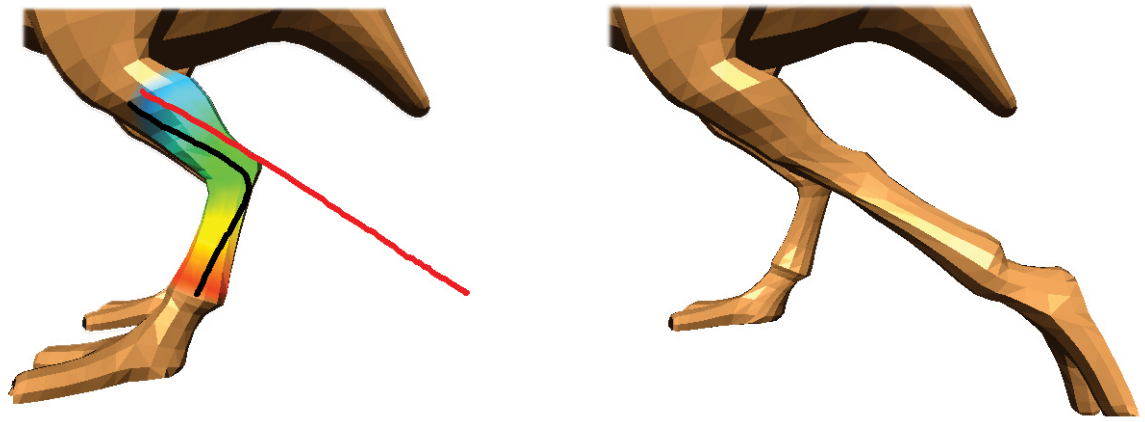


Abbildung 3.8: Beispiel einer möglichen Modellmanipulation bei Teddy. Die Deformierung wird durch einfaches Überzeichnen der gewünschten Kontur erzielt. Quelle: [29]

Alle genannten Systeme beruhen auf der Idee die Skizzierung mit Stift und Papier nachzuahmen, bei der es weniger um Exaktheit, als vielmehr das schnellstmögliche Erzielen von Ergebnissen geht. Zwar ist es möglich, die Interaktion mit der Computermaus durchzuführen, wesentlich besser geeignet sind für diese Verfahren jedoch integrierte Display-Tabletts. Dies spricht gegen den Einsatz im klinischen Bereich, da dort in den meisten Fällen lediglich Maus und Tastatur zur Verfügung stehen.

Inspiriert durch die skizzenbasierten Ansätze wird von Kho *et al.* in [33] ein weiteres, interessantes Verfahren zur Manipulation von Polygonmodellen vorgestellt. Auch hier wird die Deformation durch Skizzierung vorgenommen. Dabei wird zunächst auf dem zu deformierenden Bereich des Modells eine Referenzkurve gezeichnet. Daraus folgt einerseits die implizite Definition der *Region of Interest* (ROI) und andererseits eine stark vereinfachte Abstraktion der Ausgangssituation. Üblicherweise entspricht die Referenzkurve annähernd der Hauptskelettlinie des Modells im interessierenden Bereich, ist aber nicht darauf beschränkt. Die eigentliche Deformation wird nun durch einzeichnen einer zweiten Kurve initiiert. Dabei werden die Koordinaten der Vertices innerhalb der ROI, entsprechend der Abbildung von Referenz- auf Zielkurve transformiert (Abb. 3.9). Ähnlich wie bei den anderen, auf Skizzen basierenden Modellierungsmethoden, ist auch hier eine exakte Manipulation der Modellgeometrie nur schwer zu erreichen. Allerdings ist die vorgeschlagene Interaktionstechnik intuitiv und kann leicht auch von unerfahrenen Nutzern nachvollzogen werden. Für die Grobkorrektur medizinischer Segmentierungen scheint ein auf dieser Technik basierendes Verfahren gut geeignet.



(a) (b)

Abbildung 3.9: Skizzenbasierte Modellmanipulation nach Kho *et al.* (a) Ausgangsmodell mit eingezeichneter Referenzkurve (schwarz) und Zielkurve (rot). (b) Ergebnis der Deformation. Quelle: [33]

3.3.2 Laplacian Modeling Framework

Nachdem im vorangegangenen Abschnitt verschiedene Interaktionstechniken zur Deformation von Polygonmodellen vorgestellt wurden, geht es nun um das *Laplacian Modeling Framework* (LMF), einen Ansatz zur internen Realisierung von Modellmanipulationen. Dieses Framework ermöglicht es auf elegante Weise die Deformierung von Polygonmodellen mathematisch zu beschreiben, weshalb es zum Beispiel in [44, 4, 65, 45] und auch im Rahmen der vorliegenden Arbeit zum Einsatz kommt. Die grundlegende Idee des LMF beruht auf der Repräsentation polygonaler Netze mittels diskreter Differentialkoordinaten. In [3] wurde von dieser Idee erstmals für das Morphing von dreidimensionalen Modellen Gebrauch gemacht. Darüber hinaus wurde dort auch der mögliche Einsatz zur Freiform-Modellierung diskutiert. Das gesamte Framework zur Manipulation von Polygonmodellen wurde schließlich in [59] und [39] beschrieben. Erweiterungen dazu sind in [42] und [41] zu finden.

Sei $\mathcal{M} = (\mathcal{V}, \mathcal{E}, \mathcal{F})$ ein Polygonmodell bestehend aus einer Menge von n Vertices $\mathcal{V}$, einer Menge von Kanten $\mathcal{E}$ und einer Menge von Dreiecken $\mathcal{F}$. Jeder Vertex i hat eine Position $\vec{v}_i = (v_{xi}, v_{yi}, v_{zi})^T$ bezüglich des globalen Koordinatensystems. Die Menge $\mathcal{N}(i) = \{j \mid (j, i) \in \mathcal{E}\}$ bezeichnet dabei die direkte Nachbarschaft eines Vertex i und $deg(i)$ dessen Grad[8].

[8] also die Anzahl der Elemente von $\mathcal{N}(i)$

Die Geometrie des Modells wird somit über die Konnektivität und die Koordinaten der Vertices beschrieben. Neben der Darstellung des Netzes mittels absoluter Koordinaten $\vec{v}_i$, kann aber auch eine intrinsische Repräsentation $\vec{\delta}_i$ erreicht werden, so dass sich mit den Modellkoordinaten lokale Forminformation kodieren lässt. Im LMF geschieht dies durch Anwendung des diskreten Laplace-Operators, einer linearen Abbildung, die einem Vertex i den Differenzvektor zwischen $\vec{v}_i$ und den mittleren Koordinaten seiner Nachbarschaft zuordnet:

$$\vec{\delta}_i = \vec{v}_i - \frac{1}{deg(i)} \sum_{j \in \mathcal{N}(i)} \vec{v}_j. \tag{3.1}$$

Für die Laplace-Koordinaten des gesamten Polygonnetzes $\Delta(i) = \vec{\delta}_i^T$ ergibt sich mit

$$L(i,j) = \begin{cases} -\frac{1}{deg(i)} & (i,\ j) \in \mathcal{N}(i) \\ 1 & i = j \\ 0 & sonst \end{cases} \tag{3.2}$$

das lineare Gleichungssystem

$$LV' = \Delta. \tag{3.3}$$

Ist die absolute Position eines beliebigen Vertex i bekannt, ist das Gleichungssystem eindeutig lösbar, wobei die Lösung gerade den absoluten Koordinaten des Modells entspricht. Somit können beide Repräsentationen beliebig ineinander überführt werden.

Für die Deformation eines Polygonmodells nach dem LMF wird nun zunächst für dessen Vertices die Laplace-Repräsentation berechnet. Ziel der Manipulation ist es, bestimmten Vertices auf wohldefinierte Weise neue Positionen zuzuordnen. Wenn die Manipulation etwa durch Bewegung eines einzigen Vertex durchgeführt wird, möchte man oftmals erreichen, dass sich die Vertices in der Umgebung derart mit bewegen, dass lokale Formeigenschaften so weit wie möglich erhalten bleiben. Dies kann auch als die Forderung danach verstanden werden, eine möglichst große Übereinstimmung der Laplace-Koordinaten vor der Deformation mit denen danach zu erreichen.

Der große Vorteil des LMF besteht darin, dass Modellmanipulationen auf besonders einfache Weise beschrieben werden können. Durch das simple Hinzufügen weiterer Zeilen zum LGS (3.3) können beliebige, lineare Deformationsbedingungen aufgestellt werden. Im einfachsten Fall ist das die Zuweisung einer konstanten Position zu einem Vertex i, um diesen neu zu positionieren oder aber zu verankern.

$$\vec{v}_i = \vec{c}_1 \tag{3.4}$$

Zudem ist es möglich, für beliebige Punkte des Polygonnetzes positionale Bedingungen anzugeben. So kann jeder Punkt auf einer Kante durch Linearkombination von zwei (3.5), beliebige Dreieckspunkte durch Linearkombination von drei Vertices (3.6) dargestellt werden.

$$t\,\vec{v}_i + (1-t)\,\vec{v}_j = \vec{c}_2 \tag{3.5}$$

$$s\,\vec{v}_i + t\,\vec{v}_j + (1-s-t)\,\vec{v}_k = \vec{c}_3 \tag{3.6}$$

Darüber hinaus wurden in [65] vektorielle Bedingungen eingeführt, mit denen die relative Position zweier Vertices festgelegt werden kann.

$$\vec{v}_i - \vec{v}_j = \vec{c}_4 \tag{3.7}$$

Die eigentliche Deformation des Polygonmodells ergibt sich schließlich aus der Lösung V' des linearen Gleichungsystems

$$\tilde{L}V' = \tilde{\Delta} \tag{3.8}$$

mit der erweiterten Koeffizientenmatrix $\tilde{L}$ und dem, um die Positionskonstanten ergänzten Laplace-Vektor $\tilde{\Delta}$. In der Regel ist (3.8) aufgrund der Zusatzbedingungen allerdings überbestimmt, so dass für dieses System keine exakte Lösung existiert. Der Ausweg besteht in einer näherungsweisen Lösung im Sinne der *Methode der kleinsten Quadrate* [63]. Gesucht ist demnach

$$\min_{V'} \|\tilde{L}V' - \tilde{\Delta}\|_2 \tag{3.9}$$

Die näherungsweise Lösung kann durch Anwendung der Normalengleichungen (3.10) bzw. (3.11) gefunden werden.

$$\tilde{L}^T\tilde{L}\,V' = \tilde{L}^T\,\tilde{\Delta} \tag{3.10}$$

$$V' = (\tilde{L}^T\tilde{L})^{-1}\,\tilde{L}^T\,\tilde{\Delta} \tag{3.11}$$

Die Deformation eines Polygonmodells mit Hilfe des *Laplacian Modeling Framework* resultiert also in optimalen Vertexkoordinaten bezüglich der Berücksichtigung positionaler Bedingungen, bei gleichzeitiger Aufrechterhaltung lokaler Formeigenschaften. Interessant ist zudem die Möglichkeit, die verschiedenen Zeilen des Gleichungssystems unterschiedlich zu gewichten, so dass die Relevanz einzelner Bedingungen hervorgehoben werden kann.

In der Regel ist die Lösung linearer Gleichungssysteme ein besonders teurer Rechenschritt. Allerdings ist die hier verwendete Koeffizientenmatrix $\tilde{L}$ extrem dünn besetzt, so dass die Berechnung mit Hilfe spezieller numerischer Methoden effizient durchgeführt werden kann. Erst dadurch wird der Einsatz des LMF für direkt manipulative Interaktionstechniken möglich.

3.4 Zusammenfassung

In diesem Kapitel wurde der Stand der Forschung betrachtet. Zu Beginn wurden wichtige Begriffe aus dem Bereich der MCI eingeführt und gezeigt, dass die Wahl der Interaktionsform für die Segmentierungskorrektur auf die direkte Manipulation fallen sollte. Im Anschluss folgte eine Betrachtung von medizinischen Segmentierungsverfahren, wobei das Hauptaugenmerk auf den dort zum Einsatz kommenden Interaktionstechniken lag. Hier zeigte sich, dass für die nachträgliche Korrektur kaum alternative Techniken existieren. Da jedoch häufig automatisierte Segmentierungsverfahren genutzt werden, besteht entsprechender Bedarf. Schließlich wurden intuitive Interaktionstechniken zur Manipulation polygonaler Modelle vorgestellt, an denen sich die Korrekturwerkzeuge orientieren könnten. Da im Verlauf der Arbeit darauf zurückgegriffen wird, folgte abschließend eine Erläuterung des *Laplacian Modeling Framework*. Im nächsten Kapitel wird nun die Ausgangsituation für die Aufgabenstellung dieser Arbeit untersucht, typische Korrekturfälle beschrieben sowie die Anforderungen an die zu entwickleden Interaktionstechniken herausgestellt.

4 Analyse der Ausgangsituation

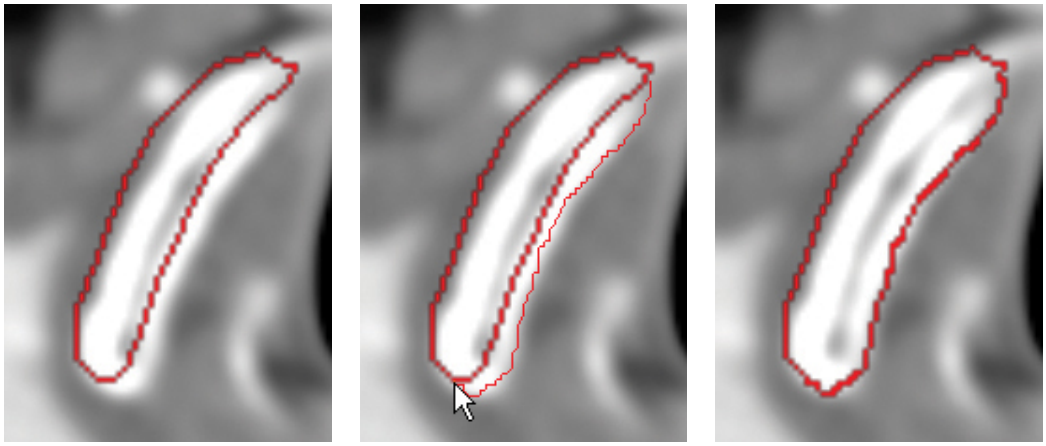
In diesem Kapitel wird die Ausgangssituation der behandelten Problemstellung untersucht. Dazu wird zunächst die gegenwärtige Vorgehensweise bei der manuellen Nachkorrektur SMSM basierter Segmentierungen diskutiert. Im Anschluss daran werden typische, zu korrigierende Segmentierungsfehler, vor allem am Beispiel der Halsstrukturen[1] betrachtet. Schließlich wird aus den vorangegangenen Überlegungen gefolgert, welchen Anforderungen die zu entwickelnden Interaktionstechniken idealerweise genügen sollten, um einen optimalen Korrekturvorgang zu ermöglichen.

4.1 Ausgangsituation

Das Szenario der Segmentierung mittels SMSM sieht gegenwärtig wie folgt aus. Der Nutzer initialisiert das Verfahren durch Positionierung des Modells in den axialen Schichtbildern des Datensatzes. Anschließend wird der automatische Vorgang gestartet und das Modell beginnt sich aufgrund der vorliegenden Bildinformation selbständig der Patientenanatomie anzupassen. Der Nutzer kann dabei jederzeit den Fortgang der Segmentierung überwachen. Das momentane Segmentierungsergebnis wird sowohl als farbige, dem Schichtbild überlagerte Kontur, als auch durch die dreidimensionale Ansicht des Modells in einem zusätzlichen Fenster visualisiert. Zudem ist es möglich, während des laufenden Vorgangs durch die CT-Schichten zu navigieren und die 3D-Darstellung aus beliebigen Blickwinkeln zu betrachten.

Je nach Segmentierungsaufgabe kann es einige Minuten dauern, bis sich schließlich ein Kräftegleichgewicht einstellt und die dynamische Verformung zum Stillstand kommt. An dieser Stelle ist die automatische Segmentierung beendet und das vorläufige Ergebnis wird einer Überprüfung durch den Anwender unterzogen. Werden dabei Mängel entdeckt, ist eine manuelle Nachkorrektur erforderlich (Abb. 4.1). Um dies zu ermöglichen, muss die Segmentierung zunächst voxelisiert und in eine editierbare Konturrepräsentation überführt werden. Der Nutzer kann nun mit einem

[1] Schildknorpel, Lymphknoten, Halstumoren



(a) (b) (c)

Abbildung 4.1: Manuelle Nachkorrektur einer Segmentierung, wie sie bisher durchgeführt wird. (a) Fehlerhafte Segmentierung, (b) nachgezeichnete Objektkontur, (c) korrigierte Segmentierung.

Zeichenwerkzeug die gewünschten Anpassungen in den axialen Schichtbildern vornehmen. Dies geschieht wie bei der manuellen Segmentierung, indem die korrekte Kontur der gesuchten Struktur (an den fehlerhaften Stellen) einfach nachgezeichnet wird[2]. So können im Falle von Untersegmentierung die bestehende Kontur ergänzt und bei Übersegmentierung entsprechende Teile abgeschnitten werden.

Positiv an dieser Vorgehensweise ist die Flexibilität, die es dem Nutzer erlaubt sein gesamtes Expertenwissen in den Korrekturvorgang einfließen zu lassen. Zwar bedeutet dies ebenfalls, dass ein subjektiv beeinflusstes Ergebnis zu erwarten ist, andererseits können auf diese Weise aber sämtliche Unzulänglichkeiten der Modellannahmen ausgeglichen werden. Das Problem der Subjektivität kann prinzipiell dadurch gemindert werden, dass korrigierte Segmentierungen je nach benötigter Exaktheit im konkreten Fall von weiteren Experten begutachtet werden.

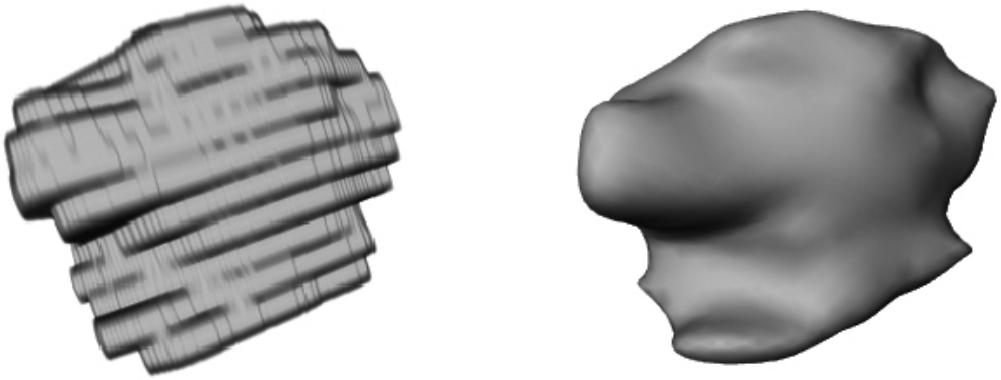
Der größte Nachteil bei der bisherigen Art der manuellen Nachkorrektur ist der erforderliche Zeitaufwand. Dies ist der kritische Faktor im klinischen Alltag, den es so gering wie möglich zu halten gilt. Da die Korrekturen in jeder einzelnen axialen Schicht vorgenommen werden müssen, steigt der Aufwand vor allem bei anatomischen Strukturen mit großer Ausdehnung in z-Richtung sowie bei höher aufgelösten Datensätzen. In letzterem Fall könnte zwar das bereits erwähnte schichtinterpolierte Live-Wire [54] verwendet werden, allerdings müssen die Zielobjekte dazu ausreichend durch Kanten von ihrer Umgebung abgrenzbar sein.

[2]Die derzeit verwendete Nachkorrektur erlaubt jedoch lediglich Freihandzeichnungen ohne die Möglichkeit interpolierter Kurveneingabe.

Nachteilig ist zudem, dass durch die bisherige Vorgehensweise zur Nachkorrektur die Dreidimensionalität der stabilen Feder-Masse-Modelle weitestgehend außer Acht gelassen wird. So kann zwar während der Segmentierung die Verformung des Modells in der 3D-Ansicht verfolgt werden, manuelle Nachkorrekturen allerdings haben keinerlei Auswirkung auf diese Darstellung. Durch die erforderliche Voxelisierung werden nur jene Teile des Modells beibehalten, die sich als Konturen aus dem Durchschnitt mit den axialen Schichtebenen ergeben. Zusätzlich zur Einblendung der Segmentierung im CT-Bild ist eine dreidimensionale Darstellung allerdings überaus sinnvoll. Zwar liefert der Datensatz die wichtigste Information für die Nachkorrektur, da hier der Bezug zur Patientenanatomie vorhanden ist, allerdings kann eine CT-Aufnahme nie als identische Abbildung der tatsächlichen, anatomischen Situation verstanden werden. Die Bildinformation ist immer gestört und mit Ungenauigkeiten behaftet, die vor allem auf den Partialvolumeneffekt zurückzuführen sind. Besonders bei hoher Schichtdicke und großem Schichtabstand - Parametereinstellungen, die oft zur Verringerung der Strahlenbelastung der Patienten gewählt werden - muss die Bildinformation kritisch betrachtet werden.

Im Allgemeinen wird der Nutzer bei der Korrektur also zusätzliches Wissen einfließen lassen. Dies geschieht zumeist durch Anwendung seiner medizinischen Fachkenntnisse, aufgrund derer er stets auch eine Vorstellung von der dreidimensionalen Gestalt des Zielobjektes vor Augen hat. In machen Fällen, wie etwa bei der Segmentierung von Halstumoren, kann dieses Wissen zum Beispiel auch aus einer vorangegangenen Endoskopie stammen. Die Nachkorrektur stützt sich daher nicht nur starr auf die vorliegende Bildinformation, sondern verfolgt vor allem das Ziel, ein anatomisch plausibles Ergebnis zu erreichen.

Um dies zu erleichtern, wäre die dreidimensionale Visualisierung der Segmentierung auch während des Korrekturvorgangs eine wertvolle Hilfestellung. Zwar ist es möglich, aus der voxelisierten Segmentierung wiederum ein dreidimensionales Polygonmodell zu generieren, in Echtzeit ist dies allerdings nicht realisierbar. Um trotzdem ohne größeren Aufwand die gesamte Gestalt der segmentierten Struktur zu betrachten, könnte bestenfalls auf ein einfaches direktes Volumenrendering (Abb. 4.2a) zurückgegriffen werden. Solche Darstellungen sind für CT-Daten jedoch oftmals weniger geeignet, da abhängig von der Schichtdicke, die Visualisierung stark durch Treppenartefakte gestört ist. Stabile Feder-Masse-Modelle dagegen (Abb. 4.2b), sind nicht von diesem Problem betroffen. Sie stellen eine wesentlich natürlichere Approximation der tatsächlichen Gestalt der anatomischen Strukturen dar. Außerdem wird schon bei der Generierung der Modelle das Ziel verfolgt, einen guten Kompromiss zwischen Anzahl der Vertices und möglichst natürlicher Form zu erreichen. Dadurch ist es möglich, auch die dynamische Anpassung des Modells während der Segmentierung effizient dreidimensional zu visualisieren.



(a) (b)

Abbildung 4.2: Dreidimensionale Visualisierung eines segmentierten Halstumors. (a) Direktes Volumenrendering und (b) Polygonmodell.

Ebenso könnte auch bei Nachkorrekturen die 3D-Darstellung in Echtzeit aktualisiert werden, würde man das Modell weiterverwenden. Somit wäre der Nutzer in die Lage versetzt, die Modifikation der Segmentierung bedeutend leichter, dem „Bild in seinem Kopf“ entsprechend durchzuführen.

4.2 Schlussfolgerungen

Insgesamt lässt sich festhalten, dass die bisherige Vorgehensweise zur Nachkorrektur vor allem deshalb ineffizient ist, weil wertvolle Information, die in Form des SMSM eigentlich bereits vorhanden ist, ungenutzt bleibt. Anstatt den dreidimensionalen Charakter beizubehalten, werden lediglich die Konturen des Modells in den axialen Schichtbildern betrachtet und somit jegliches Modellwissen aufgegeben. Als Schlussfolgerung ergibt sich, dass es wesentlich sinnvoller erscheint, die Nachkorrektur der Segmentierungen als geometrische Manipulation des Modells zu realisieren. So kann erreicht werden, dass lokale Korrekturen auch benachbarte Bereiche beeinflussen und somit der Interaktions- und schließlich auch Zeitaufwand reduziert wird. Außerdem können die Interaktionsauswirkungen auch gleichzeitig in der dreidimensionalen Darstellung des Modells betrachtet werden, wodurch eine sofortige Überprüfung der Plausibilität der gesamten Segmentierung ermöglicht wird. Wird die Segmentierung als Visualisierungsgrundlage benötigt, ergibt sich als zusätzlicher Vorteil, dass nun bereits ein verwendbares Polygonmodell zur Verfügung steht. Bis auf einen eventuellen, geringen Nachbearbeitungsschritt sind keine zusätzlichen Berechnungen notwendig, um die segmentierte Struktur angemessen zu visualisieren.

Wenn der Vorgang der Nachkorrektur als Manipulation dreidimensionaler Polygonmodelle aufgefasst wird, ergibt sich unweigerlich die Frage, wie diese Interaktion durchgeführt werden soll. Bisher geschieht die Korrektur als direkte Manipulation der segmentierten Kontur in den zweidimensionalen Schichtbildern des Datensatzes. Wie in Kapitel 3.1.2 bereits diskutiert, sollte die direkte Manipulation als Interaktionsform auch beibehalten werden. So kann eine besonders intuitive Bedienbarkeit gewährleistet und große Akzeptanz beim Nutzer erreicht werden. Sollte allerdings - da nun eine Interaktion mit dreidimensionaler Auswirkung angestrebt wird - künftig besser direkt in der 3D-Ansicht korrigiert werden?

Bei den Anwendern, die Korrekturen an medizinischen Segmentierungen vornehmen, handelt es sich um ärztliches Personal. In der Regel wird davon auszugehen sein, dass diese Nutzer über keine Erfahrung im Bereich der Polygonmodellierung verfügen. Zwar gehört der Computer, wie heutzutage für die meisten Berufsgruppen, auch zum Arbeitsalltag der Mediziner, ihr hauptsächlicher Tätigkeitsbereich ist aber die Arbeit am Patienten. Zudem stehen am typischen Computerarbeitsplatz der Mediziner lediglich die gewöhnlichen Eingabegeräte Maus und Tastatur zur Verfügung. Grundkenntnisse der Computernutzung können angenommen, fortgeschrittene Fähigkeiten wie etwa die Interaktion in virtuellen Welten jedoch nicht vorausgesetzt werden . Die Manipulation dreidimensionaler Darstellungen mit zweidimensionalen Eingabegeräten bedarf aber vor allem entsprechender Erfahrung, damit effizient gearbeitet werden kann. Bei unerfahrenen Nutzern sorgt die ungewohnte Art der Interaktion zunächst oftmals für Verwirrung. Darüber hinaus ist auch mit höherem Berechnungsaufwand zu rechnen, so dass die Interaktion unter Umständen nicht flüssig ablaufen kann und die Benutzbarkeit leidet.

Deshalb wird wie auch in [5] und [35] gefolgert, dass in Hinblick auf die Zielgruppe die Interaktion nach wie vor in den zweidimensionalen Schichtbildern stattfinden sollte. So kann auch die Akzeptanz der neuen Interaktionstechniken von vornherein gefördert werden, da zumindest dieser Aspekt dem Nutzer keine Umgewöhnung abverlangt. Durch die Arbeit mit tomographischen Bilddaten sind die Ärzte mit den Darstellungen der zweidimensionalen Schichtebenen ausreichend vertraut. Außerdem besteht in den CT-Bildern der direkte Bezug zur Anatomie, womit diese die hauptsächliche Informationsquelle für die Segmentierung darstellen. Dass eine zusätzliche 3D-Visualisierung trotzdem besonders hilfreich ist, wurde bereits erläutert. Zusammenfassend wird für die Nachkorrektur medizinischer Segmentierungen im Rahmen der vorliegenden Arbeit folgendes Grundkonzept verfolgt: Das SMSM wird nicht nur zur Segmentierung benutzt, sondern auch für deren Nachkorrektur herangezogen. Die Korrektur wird dabei als direkte, geometrische Manipulation des Modells realisiert. Angewandt werden die Interaktionstechniken wie gewohnt auf die Modellkonturen in den zweidimensionalen Schichtbildern des Datensatzes. Zusätzlich erfolgt eine dreidimensionale Darstellung der gesamten Segmentierung, in der die durchgeführten Korrekturen sofort betrachtet werden können.

4.3 Typische Korrekturfälle

4.3.1 Allgemeine Beobachtungen

Nachdem das Ausgangsszenario diskutiert wurde, sollen nun solche Situationen betrachtet werden, in denen eine Nachkorrektur durch den Nutzer erforderlich wird. Dazu folgen zunächst einige allgemeine Beobachtungen, die der bereits in Kapitel 2.3.3 vorgestellten Dissertation von Olabarriaga [48] entstammen. Das darin entwickelte *Framework for Interactive Segmentation* (FIS) beruht im Wesentlichen auf der Bereitstellung von Interaktionstechniken, die je nach Art des Fehlverhaltens der modellbasierten Segmentierung, entsprechende fallbasierte Korrekturen ermöglichen. Die Interaktionen im Rahmen des FIS werden während des dynamischen Segmentierungsvorgangs durchgeführt. Dass diese Herangehensweise zwar für die von Olabarriaga besprochenen Anwendungsbeispiele plausibel ist, sich für den Einsatz bei dreidimensionalen Segmentierungen mittels SMSM allerdings weniger anbietet, wurde bereits erörtert. Dennoch halten die grundlegenden Überlegungen aus [48] auch für die Problemstellung der vorliegenden Arbeit stand. Olabarriaga stellt fünf generelle Fälle dar, aufgrund derer modellbasierte Segmentierungen scheitern und die das Einbringen von zusätzlichem Wissen durch den Anwender erfordern. Genau diese Situationen sind es grundsätzlich auch, in denen die Interaktionstechniken der vorliegenden Arbeit zur Nachkorrektur angewandt werden müssen. Daher wird Olabarriagas Klassifikation im folgenden kurz erläutert.

K1) Schwache oder fehlende Bildinformation. In diesem Fall ist die zu segmentierende Struktur kaum oder gar nicht durch Kanten von der Umgebung abgrenzbar. Trifft dies nur auf kleinere Abschnitte der Objektkontur zu, kann die fehlende Information oftmals durch das Modellwissen kompensiert werden. Sind größere Bereiche betroffen, erhöht sich die Wahrscheinlichkeit für Fehlsegmentierungen.

K2) Irreführende Bildinformation. Hier wird das Modell aufgrund externer Kräfte von Bildregionen angezogen, deren lokale Informationen der Modellannahme entsprechen, die in der Regel jedoch nicht der zu segmentierenden Struktur angehören. Dabei kann es sich zum Beispiel um starke Kanten benachbarter Objekte handeln.

K3) Abweichung der Modellannahmen. In dieser Situation weicht die Darstellung der gesuchten Struktur im radiologischen Bild soweit von der Modellannahme ab, dass es zu Fehlsegmentierungen kommt. Hierbei können wiederum drei Fälle unterschieden werden:

- **a)** Abweichung der Bildeigenschaften (z.B. Grauwertereich)
- **b)** Lokale Abweichung von der Modellform
- **c)** Globale Abweichung von der Modellform

4.3.2 Segmentierung der Halsstrukturen

Lymphknoten. Als Ausgangsmodell für die Segmentierung der Lymphnkonten mittels SMSM wird eine synthetisch generierte Kugel verwendet. Initialisiert wird die automatische Modellanpassung durch Bestimmung eines oder zweier Punkte im Lymphknoten bzw. auf der Objektkontur.

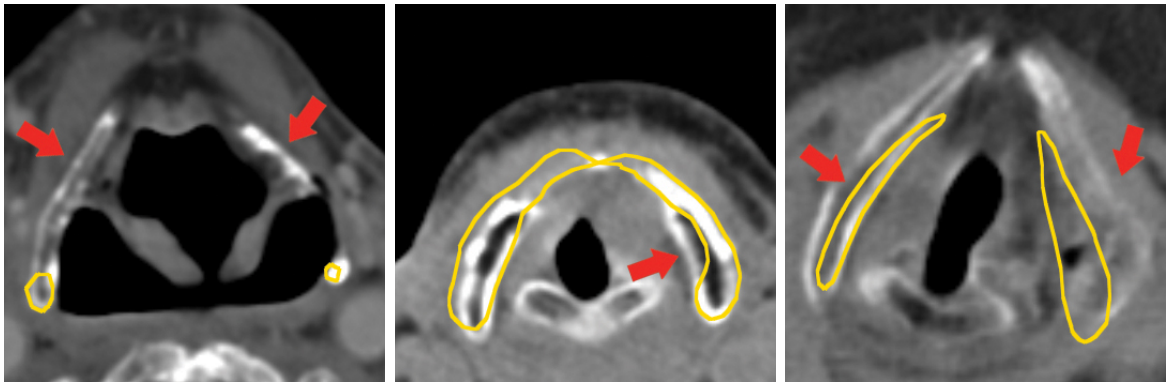
Aus den in [57] angestellten Untersuchungen geht hervor, dass es in vielen Fällen zu einer leichten Untersegmentierung der Lymphknoten kommt. Etwas umfangreichere Korrekturen sind vor allem dann nötig, soll ein Lymphknoten segmentiert werden, der sich in unmittelbarer Nachbarschaft eines Weiteren befindet. Dies ist ein klassisches Beispiel für Fall K1. Durch das lokale Fehlen der Objektkante, lässt sich allein aufgrund der Bildinformation nicht sagen, wo die Kontur zwischen den Strukturen genau verläuft. In der Regel resultiert eine solche Situation in einer Übersegmentierung, so dass sich das Modell zumindest teilweise auf beide Lymphknoten ausweitet.

Insgesamt betrachtet, sind bei der Lymphknotensegmentierung nur selten größere Nachkorrekturen erforderlich, was auf die einfache Gestalt zurückzuführen ist. Kleinere Manipulationen aufgrund von Untersegmentierungen sind aber oftmals erforderlich. Durch die einfache, rundliche Form der Lymphknoten sind die Nachkorrekturen mit Hilfe der neuen, dreidimensionalen Interaktionstechniken vermutlich in kürzester Zeit zu bewältigen.

Schildknorpel. Das zur SMSM Segmentierung des Schildknorpels verwendete Modell wurde aus der Handsegmentierung eines, als durchschnittlich erachteten Schildknorpels generiert [15]. Zur Initialisierung des automatischen Segmentierungsvorgangs, muss das Modell durch Angabe von sechs Landmarken im Datensatz positioniert werden.

Als Grundlage für die Untersuchung der Korrekturfälle im Rahmen dieser Arbeit wurden die Segmentierungen in insgesamt elf CT-Datensätzen mit unterschiedlichen Aufnahmeparametern durchgeführt. Als mit Abstand häufigste Fehlsegmentierung konnte beobachtet werden, dass es zu einer Untersegmentierung im oberen Bereich der Seitenplatten der Schildknorpel kam (Abb. 4.3a). In neun der elf Datensätze ergab sich mindestens einseitig ein derartiger Korrekturbedarf. Der Grund dafür ist wahrscheinlich die vergleichsweise schwache Signalintensität in den oberen Schichten, so dass die Modellsensoren von den unteren Bereichen stärker angezogen werden.

In den Abbildungen 4.3b und 4.3c sind weitere Fehlsegmentierungen zu sehen, bei denen die Sensoren abgelenkt werden. Das Innere der Seitenplatten stellt sich oftmals sehr dunkel dar, das Äußere dagegen besonders hell, so dass an diesen Stellen starke Grauwertgradienten vorliegen. In manchen Fällen sind die Kanten einseitig stärker ausgeprägt und führen somit zu einer fehlerhaften Modellanpassung. Diese Segmen-



(a) (b) (c)

Abbildung 4.3: Fehlsegmentierungen des Schildknorpels. (a) Untersegmentierung im oberen Bereich der Seitenplatten, (b) und (c) (links) Ablenkung der Modellsensoren durch starke Gradienten, (c) (rechts) Fehlsegmentierung aufgrund zu schwacher Bildinformation.

tierungsfehler können als besondere Ausprägung des Falls K2 nach Olabarriagas Klassifikation verstanden werden. Die Modellanpassung wird hier lokal jedoch nicht von benachbarten Bereichen anderer Strukturen, sondern von Teilen des Schildknorpels selbst gestört. Auch Fall K1 konnte beobachtet werden (Abb. 4.3c), wobei das Modell durch extrem schwache Bildinformation nicht von der Seitenplatte angezogen wurde. Eine weitere Segmentierung wies Korrekturfall K3c auf, da ein Teil des Schildknorpels durch einen Tumor zerstört wurde.

Insgesamt ist festzuhalten, dass die Segmentierung des Schildknorpels mittels SMSM eher zur Untersegmentierung tendiert und in den meisten Fällen zumindest geringer Nachkorrektur bedarf.

HNO-Tumoren. Die HNO-Tumoren nehmen innerhalb der hier vorgenommenen Betrachtungen eine Sonderstellung ein, da die vorliegenden Segmentierungen nicht automatisch mit Hilfe von SMSM, sondern manuell generiert worden sind. Der Korrekturbedarf ergibt sich hier vor allem aus der schlechten Abgrenzbarkeit der Tumoren vom umliegenden Gewebe. Oftmals lässt sich allein auf Grundlage der CT-Daten die genaue Objektkontur nicht bestimmen. Trotz dessen werden Segmentierungen der Tumoren in den Computertomographien benötigt.

Das zusätzliche Wissen um die Gestalt wird in der Regel durch endoskopische Verfahren gewonnen. Dies kann im Vorfeld einer Segmentierung, aber auch nachträglich geschehen, so dass eventuelle Korrekturen erforderlich werden können. Da Tumoren keine feste Gestalt haben und stets im Wachstum begriffen sind, oder etwa durch Bestrahlung im Rahmen einer Therapie schrumpfen, ändert sich ihre Form oft in-

nerhalb kurzer Zeiträume. Dennoch kann für gewöhnlich von Gemeinsamkeiten in der Gestalt eines Tumors bei CT-Aufnahmen zu verschiedenen Zeitpunkten ausgegangen werden. Zur Segmentierung der aktuellen Situation kann deshalb ein bereits vorhandenes Segmentierungsergebnis herangezogen werden, das lediglich an den entsprechenden Stellen einer Korrektur bedarf.

Bisher erfolgen solche Nachkorrekturen auf die gleiche Weise wie bei SMSM basierten Segmentierungen. Aber auch im Fall der manuell erstellten Tumorsegmentierungen könnte man von den neu entwickelten Interaktionstechniken profitieren. Auch ließe sich durch die dreidimensionale Visualisierung eine bestehende Segmentierung leichter an die im Rahmen einer Endoskopie gesehenen Bilder anpassen. Notwendig dafür wäre jedoch, dass ein Polygonmodell des segmentierten Tumors generiert wird. Dieser Schritt ist nicht trivial und erfordert einen gewissen Zeitaufwand, wird allerdings zu Visualisierungszwecken ohnehin oftmals durchgeführt. Wenn die neuen Techniken zu einer deutlichen Erleichterung im Vergleich zur bisherigen Nachkorrektur führen, dann ist die Generierung des Polygonmodells ein vermutlich gern in Kauf genommener Mehraufwand.

4.4 Anforderungen

Aus den vorangegangenen Überlegungen sollen nun Anforderungen abgeleitet werden, die von den gesuchten Interaktionstechniken erfüllt werden müssten, um einen möglichst optimalen Korrekturvorgang zu gewährleisten. Besonders wichtig ist die Forderung nach Flexibilität. Die Anwendung der Werkzeuge sollte nicht auf eine bestimmte Segmentierungsaufgabe beschränkt, sondern möglichst universell möglich sein. Ebenfalls ist wünschenswert, dass die Werkzeuge ungeachtet der verschiedenen Ursachen für Fehlsegmentierungen einsetzbar sind.

Darüber hinaus ist die Forderung nach einer Verminderung des aufzubringenden Zeitaufwands bei der Nachkorrektur ein wichtiger Aspekt. Zeit ist ein kritischer Faktor im klinisch medizinischen Alltag und wesentlicher Grund für den Bedarf an neuen Interaktionstechniken. Deshalb sollten die Korrekturen durch möglichst wenige Interaktionsschritte erreicht werden können. Damit insgesamt wenig Zeit aufgewandt werden muss, ist es zudem erforderlich, dass notwendige Berechnungen möglichst schnell realisiert werden können, so dass der Nutzer ein unmittelbares Feedback seiner Interaktion bekommt. Im Sinne der direkten Manipulation sollten außerdem sämtliche Interaktionsauswirkungen auch wieder rückgängig gemacht werden können.

Da bei der Zielgruppe, wie bereits erwähnt, keine fortgeschrittenen, technischen Fertigkeiten vorausgesetzt werden können, ist die intuitive Bedienbarkeit ein weiterer, wesentlicher Aspekt. Um de Akzeptanz der Korrekturwerkzeuge bei den Medizinern

zu erreichen, muss eventuelle Einarbeitungszeit so gering wie möglich gehalten werden. Da die Nachkorrektur von Segmentierungen zwar ein notwendiger, aber vom eigentlichen Aufgabenbereich medizinischen Personals weit entfernter Arbeitsschritt ist, muss angenommen werden, dass nur wenig Zeit für das Erlernen der neuen Interaktionstechniken aufgebracht werden kann.

Eine weitere Voraussetzung für die effiziente und intuitive Nachkorrektur ist, dass alle Interaktionsauswirkungen angemessen visualisiert werden. Nur wenn der Nutzer sieht, welchen Einfluss seine Interaktionen tatsächlich haben, ist er in der Lage die Interaktionstechniken auch optimal einzusetzen. In diesem Zusammenhang ist es außerdem wichtig, dass die Interaktionsauswirkungen möglichst vorhersagbar sind und unerwartetes Verhalten weitestgehend ausgeschlossen werden kann.

Das Hauptargument Olabarriagas aus [48] gegen den Einsatz von Techniken zur nachträglichen Segmentierungskorrektur ist die zu erwartende Subjektivität der Manipulationen. Um diesem Nachteil entgegenzuwirken, sollte ein gewisser Grad an Objektivität gewährleistet sein, um die Korrekturwerkzeuge möglichst robust gegenüber der Anwendung durch verschiedene Nutzer zu machen. Dies kann durch den Bezug auf bereits vorhandene Modellinformation erreicht werden, womit auch die Korrektur von Segmentierungsfehlern aufgrund der Fälle K1 und K2 erleichtert werden kann. Dennoch sollte der Nutzer durch das verwendete Modell nicht zu sehr eingeschränkt werden, so dass auch solche Fälle effizient korrigiert werden können, in denen das Modellwissen unzureichend ist (K3). Schließlich ist es zudem, vor allem zur Zeitersparnis erforderlich, dass die Interaktionsauswirkungen auf die fehlerhaften Bereiche der Segmentierung beschränkt werden und bereits korrekt segmentierte Bereiche unbeeinflusst bleiben.

4.5 Zusammenfassung

In diesem Kapitel wurde die bisherige Vorgehensweise zur Nachkorrektur SMSM basierter Segmentierungen analysiert. Hieraus wurde das grundsätzliche Prinzip der zu entwickelnden Interaktionstechniken geschlussfolgert: Die Korrektur wird als geometrische Manipulation des Segmentierungsmodells konzipiert, wobei die Interaktion in den 2D-Schichtbildern stattfindet und das Modell parallel dazu auch dreidimensional visualisiert wird. Außerdem wurden die typischen Fälle betrachtet, in denen die Interaktionstechniken zur Nachkorrektur angewandt werden müssen. Schließlich wurde aus den gewonnenen Erkenntnissen gefolgert, welche Anforderungen für die optimale Durchführung von Nachkorrekturen erfüllt sein sollten. Das folgende Kapitel behandelt den Entwurf der entwickelten Interaktionstechniken.

5 Entwurf der Interaktionstechniken

In diesem Kapitel soll es nun um den Entwurf der entwickelten Interaktionstechniken zur Nachkorrektur medizinischer Segmentierungen gehen. Zu Beginn werden dazu, die aus anderen wissenschaftlichen Arbeiten aufgegriffenen Ideen benannt. Danach werden einige generelle Konzepte erläutert, die im Rahmen der Interaktionstechniken zum Einsatz kommen. Dies umfasst zum einen unterstützende Techniken zur Visualisierung der Datensätze und zur Begrenzung des Einflussbereiches der Modellmanipulationen, zum anderen algorithmische Konzepte, auf denen die Korrekturwerkzeuge aufbauen. Im Anschluss wird schließlich der Entwurf der entwickelten Interaktionstechniken vorgestellt.

5.1 Aufgegriffene Ideen

Wie im vorigen Kapitel gefolgert, wird die Korrektur der Segmentierungen im Rahmen der vorliegenden Arbeit als interaktive Deformation von Polygonmodellen aufgefasst. Deshalb wurden bereits in Kapitel 3.3 solche wissenschaftlichen Arbeiten betrachtet, die sich mit intuitiven Ansätzen für diese Aufgabenstellung befassen. Die darin angeführten Beispielanwendungen beziehen sich zwar hauptsächlich auf die Manipulation von an Computerspielfiguren erinnernden Modellen, dennoch erscheinen einige der verwendeten Ideen auch sinnvoll für die Nachkorrektur medizinischer Segmentierungen einsetzbar zu sein. Zudem zielen die beschriebenen Interaktionstechniken in erster Linie auf Anfänger bzw. Gelegenheitsnutzer ab, also jenen Anwenderkreis, dem in der Regel auch medizinisches Personal angehört.

Vor allem die auf dem *Laplacian Modeling Framework* (LMF) basierenden Ansätze [44, 4, 65, 45] sind viel versprechend, da hier während der Deformation lokale Formeigenschaften erhalten bleiben. Bei der Verwendung von SMSM zur Segmentierung liegt in Form des verwendeten Modells bereits Wissen über die zu erwartende Gestalt der Zielobjekte vor. In den Fällen, in denen eine zu korrigierende Fehlsegmentierung nicht auf fehlerhafte Gestaltannahmen zurückzuführen ist, kann die Korrektur durch Anwendung des LMF auf sinnvolle Weise erleichtert werden. So werden beliebige Deformierungen zugunsten solcher, die Formmerkmale beibehalten, erschwert.

Als Interaktionstechnik wird das klassische, zur Freiform-Deformierung verwendete Handle-Konzept aufgegriffen, das als Metapher leicht verständlich ist. Im Gegensatz zur mächtigen, aber komplizierten Manipulation eines FFD-Hüllgitters [6, 56], wird im Rahmen dieser Arbeit eine direkte Manipulation des Polygonmodells angestrebt.

Auch die skizzenbasierten Ansätze wie *Teddy* [29] stellen eine interessante Möglichkeit der Modellmanipulation dar. Hier zeigen sich Parallelen zur bisher verwendeten manuellen Segmentierung, weshalb für diese Technik mit großer Akzeptanz seitens der Anwender gerechnet werden kann. Der Vorteil gegenüber der bisherigen Methode besteht zum einen darin, dass die Manipulation nicht auf einzelne Schichten beschränkt, sondern eine dreidimensionale Operation ist. Zum anderen lässt sich bei der Kombination mit dem LMF, ähnlich wie in [44] erreichen, dass die neuberechnete Kontur nicht exakt der vom Nutzer gezeichneten entspricht. Vielmehr wird die Nutzereingabe tatsächlich nur als Skizzierung der gewünschten Manipulation verstanden, während die Deformation auch hier wieder unter Aufrechterhaltung lokaler Formeigenschaften durchgeführt wird. Darüber hinaus wird im Rahmen dieser Arbeit auch ein Werkzeug in Anlehnung an die in [33] vorgestellte Technik entworfen.

5.2 Allgemeine Konzepte

5.2.1 Visualisierung

Als eine Schlussfolgerung der Analyse in Kapitel 4 ergab sich, dass geeignete Visualisierung eine wichtige Voraussetzung für die effiziente und intuitive Korrektur medizinischer Segmentierungen ist. Wie erläutert wurde, soll die Interaktion nach wie vor in der Darstellung der zweidimensionalen Schichtbilder des Datensatzes erfolgen. Zusätzlich wird jedoch auch eine dreidimensionale Visualisierung angeboten, in der zu jedem Zeitpunkt die aktuelle Gestalt der gesamten Segmentierung betrachtet werden kann. Um die Orientierung für den Nutzer noch wesentlich zu erleichtern, wird die im aktuell betrachteten Schichtbild sichtbare Kontur auch als farbige Markierung auf der 3D-Visualisierung hervorgehoben. Auf diese Weise kann die kombinierte Verwendung beider Darstellungen sinnvoll unterstützt werden, da sofort ersichtlich ist, welcher Teil des Modells im Schichtbild zu sehen ist.

Bei der bisherigen Art der Nachkorrektur ist der Nutzer auf die Interaktion in den axialen Schichtbildern beschränkt. Durch die angestrebte dreidimensionale Herangehensweise allerdings, ergibt sich die Möglichkeit auch andere Schnittführungen zu verwenden. Theoretisch könnten dazu beliebige Ebenen des Datensatzes berechnet werden. Dies ist jedoch eher unüblich und resultiert in schwerer zu interpretierenden Bildern. Gewohnt sind die Mediziner hingegen die Betrachtung der orthogonalen

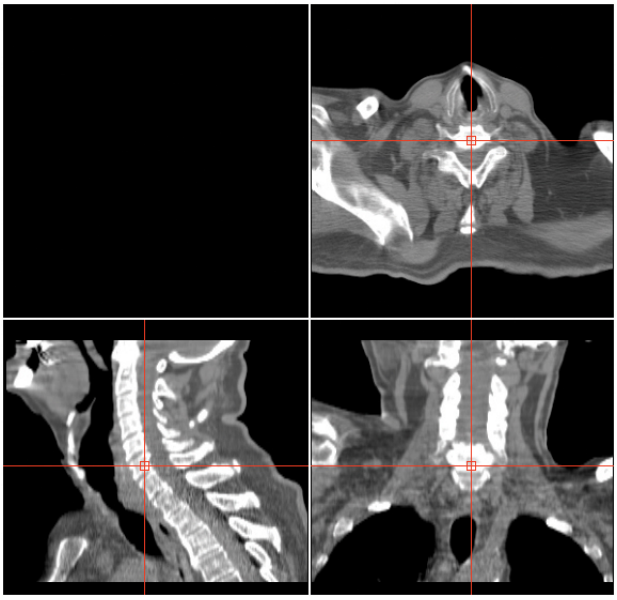


Abbildung 5.1: Darstellung der orthogonalen Schnittebenen eines CT-Datensatzes. Der Cursor markiert den Voxel, der die Schnittmenge der gezeigten Schichtbilder darstellt. Durch die Hilfslinien ist zudem die Lage der jeweils anderen Ebenen zu erkennen. Oben rechts: axiale Ansicht, unten links: sagittale Ansicht, unten rechts: koronare Ansicht.

Hauptschnittebenen[1]. Häufig werden CT-Daten durch die gleichzeitige Darstellung dieser drei Ebenen visualisiert. Dabei kann derart durch den Datensatz navigiert werden, dass der mit der Maus markierte Voxel die Schnittmenge der dargestellten Schichtbilder ergibt. Die in der Regel wesentlich feinere Auslösung in x- und y-Richtung führt zu einer entsprechend höheren Anzahl an möglichen Schnittbildern in den sagittalen und koronalen Darstellungen, als dies für die axiale Ansicht der Fall ist. In Abbildung 5.1 ist die typisch angeordnete Visualisierung der orthogonalen Schnittbilder zu sehen. Die bisherige Realisierung der Nachkorrektur macht von dieser Darstellungsart keinen Gebrauch. Für die dreidimensionale Modellmanipulation dagegen, kann sie vorteilhaft genutzt werden. Vor allem werden somit Deformierungen in z-Richtung bequem durchführbar.

Ein Vorteil der angestrebten dreidimensionalen Korrektur ist es, dass anders als bisher, die Auswirkungen einzelner Manipulationsschritte nicht zwangsläufig auf die Schicht begrenzt sind, in der die Interaktion stattfindet. So führt eine Manipulation der Segmentierung in einem Schichtbild in der Regel auch zu Änderungen der Kontur in benachbarten Schichtbildern. Problematisch ist dabei, dass die gesamte Interaktionsauswirkung nicht direkt zu überblicken ist. Hier erweist sich erneut

[1] Axial, Koronar, Sagittal (siehe auch Abb. 2.7)

die dreidimensionale Visualisierung als hilfreich. Jedoch ist in dieser Darstellung kein Bezug zur Patientenanatomie vorhanden, weshalb in jedem Fall die veränderte Kontur auch im Datensatz überprüft werden muss. Die einfachste Methode dies zu erreichen, besteht darin, durch die Schichten zu navigieren. Werden unerwünschte Auswirkungen einer vorherigen Manipulation entdeckt, können diese wieder behoben werden. Da aber dabei wiederum nur die aktuelle Schicht sichtbar ist, kann es erneut zu problematischen Deformierungen kommen. Im ungünstigsten Fall kann der eigentliche Vorteil der dreidimensionalen Herangehensweise so auch zu unnötigen Verzögerungen bei der Nachkorrektur führen.

Um unerwünschtes Verhalten möglichst direkt zu unterbinden, wird eine Methode benötigt, mit der die Auswirkungen einer Manipulation auf andere Schichten schnell und effizient beurteilt werden können. Eine Möglichkeit dafür, das sogenannte *Ghosting*, ist in Abbildung 5.2 dargestellt. Die Idee dahinter ist, dass zusätzlich zur Segmentierung im aktuellen Schichtbild, die Konturen aus benachbarten Schichten eingeblendet werden. Um die Unterscheidung zu erleichtern, können verschiedene Farben und Transparenz verwendet werden. Dennoch kann diese Art der Darstellung schnell unübersichtlich werden. Außerdem sind zwar die Konturen der Segmentierung, nicht aber die anatomische Situation der Nachbarschichten sichtbar.

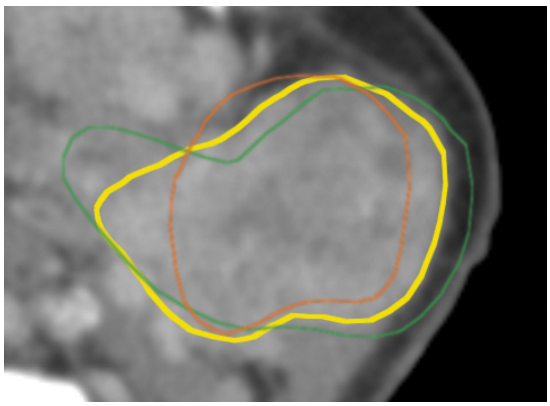


Abbildung 5.2: Ghosting bei der Visualisierung einer Segmentierung.

Um eine übersichtliche Darstellung der Segmentierung mehrerer Schichten mit Bezug zur anatomischen Situation zu ermöglichen, wurde im Rahmen der vorliegenden Arbeit, zur unterstützenden Visualisierung der Interaktionstechniken der *Multi Slice Viewer* entwickelt. Dabei werden ähnlich der orthogonalen Ansicht, drei Schichtbilder gemeinsam dargestellt. Die Bilder zeigen die gleiche Schnittführung, innerhalb dieser jedoch verschiedene Schichten des Datensatzes. In Abbildung 5.3 ist die Anordnung der Bilder zu sehen. Den größten Bereich des Fensters nimmt die Visualisierung der Referenzschicht ein. Darunter befinden sich zwei etwas kleinere

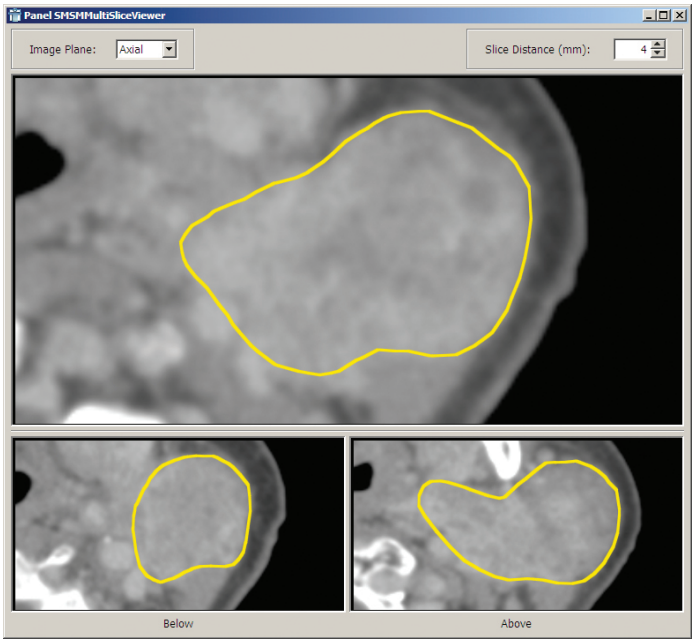


Abbildung 5.3: Der Multi Slice Viewer ermöglicht das simultane Betrachten mehrerer Schichten eines Datensatzes. Auf diese Weise können die Auswirkungen einer Korrektur auf benachbarte Schichten beurteilt und gegebenenfalls sofort berichtigt werden.

Darstellungen benachbarter Schichten. Das linke Bild stammt dabei aus dem Datenvolumen unterhalb, das Rechte aus dem Volumen oberhalb der Referenzebene[2]. Der wichtigste Parameter des Multi Slice Viewers ist der Abstand der dargestellten Nachbarschichten zur Referenzebene. Dadurch können die Auswirkungen einer Korrektur bis auf beliebig entfernte Schichten betrachtet werden. Zudem sind die Visualisierungen derart miteinander gekoppelt, dass die Einstellung von Zoom-, Verschiebungs- und Fensterungsparametern für alle drei Bilder gemeinsam, auf der Referenzschicht vorgenommen wird. Auch navigiert wird ausschließlich durch Änderung der Referenzebene, wobei die Darstellung der Nachbarschichten entsprechend automatisch aktualisiert wird. Als darstellbare Schnittführung kann wie vorher erläutert, zwischen den drei orthogonalen Hauptebenen gewählt werden.

Das Konzept des Multi Slice Viewers ermöglicht dem Nutzer die Auswirkungen seiner Korrekturen auf benachbarte, ansonsten aber nicht sichtbare Schichten des Datensatzes zu beurteilen. Dabei wird nicht nur die Kontur der Segmentierung, sondern auch die anatomische Situation der Nachbarschichten dargestellt. Da diese Ansicht zusätzlich zur dreidimensionalen Visualisierung des Segmentierungsmodells

[2]Unterhalb bzw. oberhalb meint hier, dass die Koordinate bezüglich der Schnittachse, kleiner bzw. größer als die der Referenzschicht ist.

zur Verfügung steht, kann der Nutzer direkt entscheiden, ob ein Korrekturschritt unerwünschte Folgen hatte. Ist dies der Fall, kann sogleich mit Blick auf die benachbarte Anatomie entsprechend berichtigt werden. Die Interaktion zur Nachkorrektur kann dabei in allen drei gezeigten Schichtbildern durchgeführt werden.

5.2.2 Begrenzung des Einflussbereiches

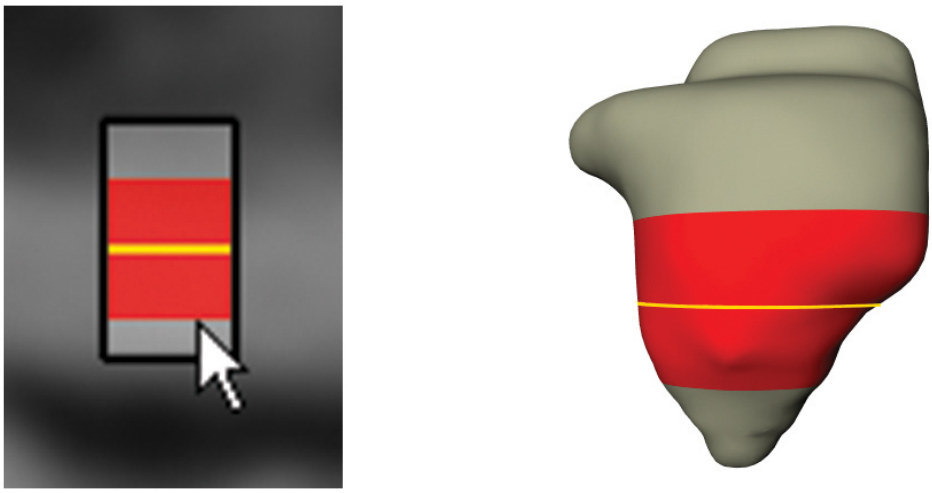
Da bei den Segmentierungen mittels SMSM stets problemspezifische Modelle zum Einsatz kommen, kann grundsätzlich davon ausgegangen werden, dass ein, aus dem dynamischen Segmentierungsprozess hervorgegangenes Ergebnis zum größten Teil mit der tatsächlich gesuchten Segmentierung übereinstimmt. Daher werden eventuelle Nachkorrekturen immer nur an einzelnen Teilen des Segmentierungsmodells erforderlich sein. Der Einfluss der verwendeten Korrekturwerkzeuge sollte deshalb auch nur auf eben solche Bereiche beschränkt sein, damit bereits korrekte Teile des Modells nicht beeinträchtigt werden.

Die Definition der interessierenden Bereiche (ROI[3]) ist von der jeweiligen Interaktionstechnik abhängig und ergibt sich entweder implizit oder muss explizit vorgenommen werden. Da die Interaktion in den zweidimensionalen Schichtbildern stattfindet, erfolgt dort auch die ROI-Definition. Ein großer Vorteil der in dieser Arbeit verfolgten Herangehensweise ist, dass die Dreidimensionalität der zu segmentierenden Objekte berücksichtigt wird - Manipulationen in einer Bildebene sich also auch auf benachbarte Schichten auswirken. Um diesen Effekt sinnvoll einsetzen zu können, ist es erforderlich, zusätzlich zur zweidimensional definierten ROI den Einfluss auf andere Schichten regulieren zu können.

Um dem Nutzer die Möglichkeit zu bieten, den Einflussbereich der Korrekturwerkzeuge explizit zu begrenzen, wurde mit dem *Depth Limiter* ein entsprechendes, einfaches Hilfsmittel entwickelt. Damit können zwei Grenzebenen bestimmt werden, die parallel zur Bildebene verlaufen und sich oberhalb bzw. unterhalb von dieser befinden. So wird der maximale Abstand auf der Schnittachse zur Bildebene definiert, den ein Vertex haben kann, um noch im Einflussbereich der Deformation zu liegen. Ein solcher Maximalabstand wird für jede der drei Bildebenen definiert.

Die nötige Interaktion findet anhand der Manipulation einer kleinen Visualisierung statt, die im zweidimensionalen Schichtbild erscheint. Durch Bewegung der Maus bei gedrückter rechter Maustaste kann die vertikale Ausdehnung eines roten Rechtecks und damit der Abstand der Grenzebenen reguliert werden, wobei eine horizontale, gelbe Linie die aktuelle Bildebene symbolisiert. Zusätzlich wird das ROI-Volumen als farbige Markierung auf der 3D-Visualisierung dargestellt (Abb. 5.4).

[3]von engl. Region of Interest



(a) (b)

Abbildung 5.4: Begrenzung des Einflussbereichs mittels Depth Limiter. (a) Visualisierung des Abstands der Grenzebenen zur Bildebene, dargestellt im Schichtbild. (b) Visualisierung der durch den Depth Limiter bestimmten ROI in der dreidimensionalen Darstellung. Das ROI-Volumen ist rot markiert, der gelbe Linienzug zeigt die im aktuell betrachteten Schichtbild sichtbare Kontur des Modells.

5.2.3 Laplacian Deformation

Die Anwendung des Laplacian Modeling Framework erfordert die Lösung eines linearen Gleichungssystems. Dabei handelt es sich im Allgemeinen um eine besonders aufwändige Rechenoperation. Um das LMF für Interaktionstechniken im Sinne der direkten Manipulation einzusetzen, muss die Interaktionsauswirkung jedoch möglichst unmittelbar auf die Interaktion folgen. Ist dies durch gewöhnliche Matrixoperationen für kleine ROIs noch gewährleistet, kommt es bei steigender Vertexanzahl relativ bald zu nicht hinnehmbaren Verzögerungen bei der Ausführung. Bei genauerer Betrachtung des zu lösenden LGS fällt jedoch auf, dass die Koeffizientenmatrix extrem dünn besetzt ist und somit durch die herkömmlichen Operationen etliche, unnötige Berechnungsschritte durchgeführt werden.

Sei deg_{max} der maximale Grad der insgesamt n betrachteten Vertices eines Polygonmodells. Die um m zusätzliche Deformationsbedingungen erweiterte Laplace-Matrix $\tilde{L}$ besteht aus $n^2+n\cdot m$ Elementen, bei denen es sich zum Großteil jedoch um Nullen handelt. Für die Anzahl der Nicht-Nullelemente der Matrix gilt:

$$n_{\neq 0} \leq n \cdot (1 + deg_{max}) + 3 \cdot m. \tag{5.1}$$

Für das SMSM zur Segmentierung des Schildknorpels etwa, beträgt $deg_{max} = 6$. Bei einer ROI Größe von 10 Vertices führt dies zu einem relativen Anteil der Nicht-Nullelemente in L von etwa 0.7, bei 30 Vertices von etwa 0.25. Schon bei einer ROI Größe von 150 Vertices beträgt der Anteil der von Null verschiedenen Elemente noch nicht einmal mehr fünf Prozent der nun insgesamt 22500 Matrixelemente. Während sich $n_{\neq 0}$ lediglich linear entwickelt, wächst die Gesamtzahl der Elemente quadratisch in n. Für große ROIs ergibt sich daher ein extremer Effizienzverlust bei Speicherplatz und Rechenzeit.

Um trotzdem einen effizienten Umgang, auch mit extrem dünn besetzten Matrizen zu ermöglichen, existieren spezielle Speicherformate, wie das *Compressed Column Storage* (CCS) [14]. Dabei werden nur die Nicht-Nullelemente einer Matrix sequentiell im Speicher $values$ abgelegt. Zu jedem dieser Elemente wird außerdem in einem Array $rowind$ dessen Zeilenindex gespeichert. In einem dritten Array $colptr$ werden zusätzlich die Indizes vermerkt, an denen in $values$ das jeweils erste Nicht-Nullelement der einzelnen Spalten liegt. Nach Konvention wird zudem das $(n+1)$-te Element von $colptr$ auf $n_{\neq 0} + 1$ gesetzt. Auf diese Weise müssen statt n^2 nur noch $2 \cdot n_{\neq 0} + n + 1$ Speicherstellen belegt werden. Somit kann eine Vielzahl unnötiger Iterationen eingespart und auch aufwändigste Rechenoperationen, wie die Faktorisierung von Matrizen in kurzer Zeit ausgeführt werden. Das Prinzip des CCS ist nachfolgend am Beispiel einer 5×5 Matrix veranschaulicht.

$$\begin{pmatrix} 0 & 0 & 6 & 0 & 3 \\ 0 & 1 & 1 & 0 & 0 \\ 4 & 0 & 0 & 2 & 0 \\ 0 & 6 & 9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 8 \end{pmatrix} \quad \Rightarrow$$

values	4	1	6	6	1	9	2	3	8
rowind	3	2	4	1	2	4	3	1	5
colptr	1	2	4	7	8	9	·	·	·

Die erweiterte Koeffizientenmatrix $\tilde{L}$ wird bei der Anwendung für die entwickelten Interaktionstechniken wie folgt aufgebaut. Voraussetzung ist, dass eine als ROI definierte Teilmenge des Polygonmodells bestimmt wurde, auf dem die Manipulation durchgeführt werden soll. Für die darin enthaltenen Vertices sollen während der Deformation die lokalen Formeigenschaften und damit deren Laplace-Vektoren weitestgehend unverändert bleiben. Um die Laplace-Vektoren zu bestimmen, müssen alle Nachbarn der Vertices in die Rechnung miteinbezogen werden. Dies führt dazu, dass auch solche Vertices im LGS berücksichtigt werden müssen, die sich außerhalb der ROI, aber in deren unmittelbarer Nachbarschaft befinden. In Abbildung 5.5 ist diese Situation beispielhaft dargestellt. Alle dort farbig markierten Vertices werden dabei zum Aufbau der Matrix $\tilde{L}$ herangezogen. Zu jedem grün bzw. gelb markierten Vertex wird schließlich der Laplace-Vektor berechnet, für jeden rot markierten Vertex dagegen, eine Fixationsbedingung in die Matrix aufgenommen. Dadurch

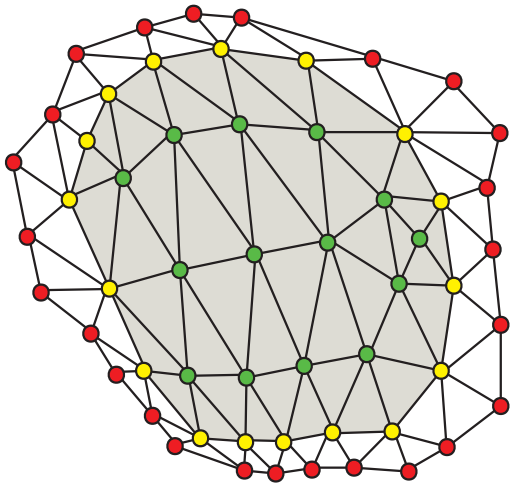


Abbildung 5.5: Definierte ROI eines Polygonnetzes. Die grau gefärbte Fläche markiert den Bereich der ROI. Grüne Vertices liegen im Inneren und gelbe Vertices auf dem Rand der ROI. Rote Vertices befinden sich außerhalb, sind jedoch mit Randvertices benachbart und müssen deshalb in das lineare Gleichungssytem miteinbezogen werden.

kann erreicht werden, dass die deformierte ROI durch die benachbarten Vertices im übrigen Polygonmodell „verankert“ bleibt. Die resultierende Matrix wird dabei wie folgt gebildet.

$$\tilde{L}(i,j) = \begin{cases} -\frac{1}{deg(i)} & (i,\ j) \in \mathcal{N}(i)\ \wedge\ i \in ROI \\ 1 & i = j \\ 0 & sonst \end{cases} \tag{5.2}$$

Als rechte Seiten des LGS werden die Spaltenvektoren der $n \times 3$ Matrix $\tilde{\Delta}$ verwendet[4]. Diese beinhaltet die Laplace-Vektoren der ROI-Vertices, sowie die globalen Koordinaten der fixierten Vertices außerhalb der ROI. Damit sind die Rahmenbedingungen für die formerhaltende Manipulation des Modells definiert.

Die eigentliche Deformierung wird durch zusätzliche Bedingungen beschrieben, die als weitere Zeilen in die Matrix $\tilde{L}$ aufgenommen werden. In der Regel werden dafür sogenannte *On-Face Constraints* [42] verwendet, also bestimmten Punkten inner-

[4]Da die Deformation im dreidimensionalen Raum stattfindet, müssen für die Berechnung der neuen Vertexpositionen jeweils drei Komponenten bestimmt werden. Das lineare Gleichungssytem ist also dreimal mit verschiedenen rechten Seiten zu lösen.

halb der Dreiecksflächen neue Positionen zugeordnet. Dazu wird die ursprüngliche Position eines Punktes $\vec{p}$ auf einer Dreiecksfläche F in baryzentrischen Koordinaten $\vec{b}^F(\vec{p})$ angegeben. Die Komponenten eines baryzentrischen Vektors entsprechen dabei den Koeffizienten, mit denen die Positionen der Dreiecksvertices gewichtet werden müssen, um $\vec{p}$ als deren Linearkombination darzustellen. Um $\vec{p}$ für die Deformation eine neue Position zuzuordnen, wird diese als weitere Zeile in die rechte Seite $\tilde{\Delta}$ aufgenommen.

Um schließlich die Vertexkoordinaten der deformierten ROI zu bestimmen, muss das LGS gelöst werden. Da wegen der geforderten Deformationsbedingungen allerdings mehr Zeilen als Spalten enthalten sind, ist das LGS überbestimmt, so dass keine exakte Lösung existiert. Somit ist das System lediglich näherungsweise lösbar. Dies geschieht im Sinne der Methode der kleinsten Quadrate durch die Anwendung von Normalengleichungen. Dazu ist es erforderlich, die erweiterte Koeffizientenmatrix $\tilde{L}$ von links mit dessen Transponierter zu multiplizieren. Um auch diesen Vorgang weiter zu beschleunigen, wurde Algorithmus 5.1 entwickelt. Hierbei wird ausgenutzt, dass die Elemente (ij) von $\tilde{L}^T\tilde{L}$, gerade dem Skalarprodukt der i-ten und j-ten Spalten aus $\tilde{L}$ entsprechen und die resultierende Matrix symmetrisch ist, so dass lediglich die obere Dreiecksmatrix berechnet und gespeichert werden muss.

Die Anwendung der Normalengleichungen resultiert schließlich in einem exakt lösbaren, linearen Gleichungssytem. Somit kann die formerhaltende Deformation des Polygonmodells durch Faktorisierung der Koeffizientenmatrix und anschließender Rücksubstitution effizient berechnet werden [63]. Trotz Nutzung spezieller, hochoptimierter Algorithmen zur Lösung dünn besetzter LGS[5] kann dieser Vorgang bei großflächigen ROIs oder sehr feinmaschigen Polygonnetzen für den Nutzer spürbare Zeit in Anspruch nehmen und den flüssigen Ablauf der Interaktion behindern. Jedoch lässt sich auch hier noch eine wesentliche Beschleunigung erzielen. Will der Nutzer ein und dieselbe ROI solange mehrfach deformieren, bis das gewünschte Ergebnis erreicht ist, muss die Berechnung von $\tilde{L}^T\tilde{L}$ und die anschließende, aufwändige Faktorisierung nämlich nur einmal, zu Beginn der Manipulation durchgeführt werden. Damit ist gewährleistet, dass auf dem LMF basierende Deformationen mittels direkt manipulativer Interaktionstechniken realisierbar sind.

[5] wie sie z.B. die Programmbibliothek TAUCS [62] beinhaltet

Algorithmus 5.1 Berechnung von $\tilde{L}^T\tilde{L}$

$ind \leftarrow 1$
for $i \leftarrow 1$ to n **do**
  $colptr_{L^TL}[i] \leftarrow ind$
  for $j \leftarrow i$ to n **do**
    $v \leftarrow 0$
    $ind_i \leftarrow colptr_L[i]$
    $ind_j \leftarrow colptr_L[j]$
    while $(\,(\,ind_i < colptr_L[i+1]\,) \wedge (\,ind_j < colptr_L[j+1]\,)\,)$ **do**
      if $(\,rowind_L[ind_i] = rowind_L[ind_j]\,)$ **then**
        $v \leftarrow v + values_L[ind_i] * values_L[ind_j]$
        $ind_i \leftarrow ind_i + 1$
        $ind_j \leftarrow ind_j + 1$
      else
        if $(\,rowind_L[ind_i] < rowind_L[ind_j]\,)$ **then**
          $ind_i \leftarrow ind_i + 1$
        else
          $ind_j \leftarrow ind_j + 1$
        end if
      end if
    end while
    if $(\,v \neq 0\,)$ **then**
      $rowind_{L^TL}[ind] \leftarrow j$
      $values_{L^TL}[ind] \leftarrow v$
      $ind \leftarrow ind + 1$
    end if
    $j \leftarrow j + 1$
  end for
  $i \leftarrow i + 1$
end for
$colptr_{L^TL}[n+1] \leftarrow ind$

5.2.4 Detailgrad der Modelle

Bei der Generierung eines zur SMSM Segmentierung verwendeten Modells müssen generell zwei Gesichtspunkte berücksichtigt werden. Zum einen sollte die Anzahl der Massen möglichst gering sein. Jede äußere Masse entspricht einem Vertex im Polygonmodell, dass zur Korrektur und Visualisierung herangezogen wird. Manche SMSM allerdings bestehen zusätzlich noch aus weiteren Massefronten, so dass die tatsächliche Massenanzahl einem Vielfachen der Anzahl der Vertices entspricht. Das wesentliche Problem bei hoch aufgelösten Modellen ist, dass der notwendige Zeitaufwand für die Segmentierung direkt von der Anzahl der verwendeten Massen abhängt.

Allerdings muss auch eine gute Annäherung an die Gestalt der zu segmentierenden Objekte erreicht werden, denn ein gewisser Detailgrad ist notwendig, um die Modelle überhaupt erfolgreich zur Segmentierung einsetzen zu können. Je geringer die Auflösung, desto größer die Flächeninhalte der Dreiecke und umso schlechter die resultierende Approximation der gekrümmten Oberflächen. Dies macht sich letztlich auch im optischen Eindruck bemerkbar, den der Nutzer durch die Visualisierung der Modelle gewinnt. So wurde etwa von Seiten unseres klinischen Partners das kantige Aussehen der SMSM bemängelt. Aber auch für die Korrektur ist es mitunter förderlich, höher aufgelöste Modelle zu verwenden, da sich lokale Deformierungen dadurch wesentlich zielgerichteter durchführen lassen bzw. teilweise sogar erst möglich werden.

Organ	Massen	Vertices
Lymphknoten	100	50
Schildknorpel	300	300
Kehldeckel	602	301
Linker Herzventrikel	153	64

Tabelle 5.1: Auflösung verschiedener SMSM.

Das Problem des ungenügend ansprechenden Aussehens der SMSM ist zusätzlich zur geringen Auflösung im *Flat Shading*[6] begründet, dass bisher für die Visualisierung verwendet wurde. Durch die Nutzung des *Gouraud Shading* [25] konnte der optische Eindruck im Rahmen der Korrektur bereits wesentlich verbessert werden. Da es hierbei jedoch nur zu einer scheinbaren Steigerung des Detailgrades kommt, lässt sich auf diese Weise kein Vorteil für die Nachkorrektur erzielen.

[6] d.h. sämtliche Bildpunkte eines Dreiecks bekommen beim Rendering ein und derselben Farbwert zugewiesen

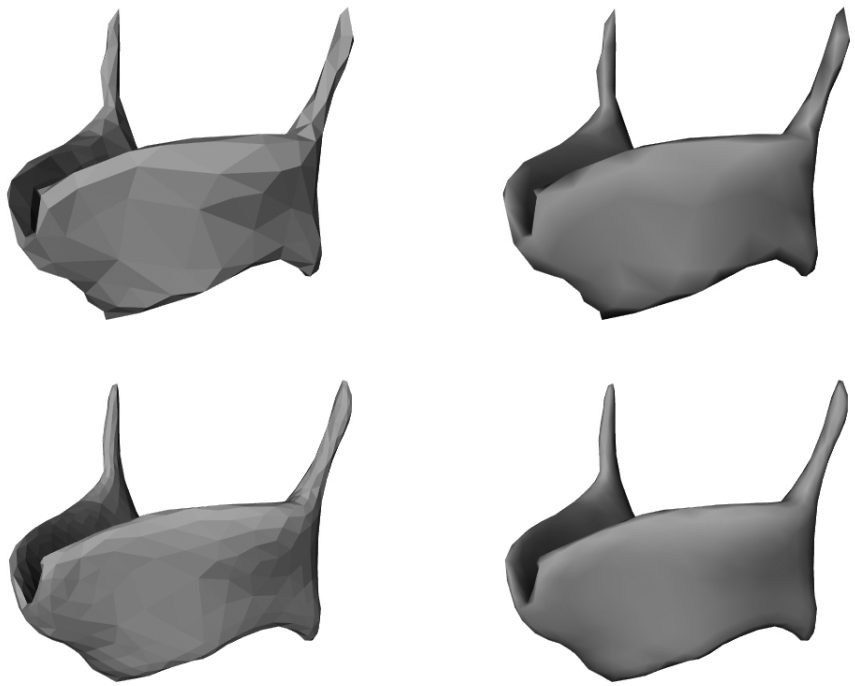


Abbildung 5.6: Detailgrad bei SMSM des Schildknorpels. Obere Reihe: Ausgangsmodell, untere Reihe: $\sqrt{3}$-Subdivision, eine Iteration. (a) Flat Shading, (b) Gouraud Shading.

Um tatsächlich die Auflösung der Modelle zu erhöhen, bieten sich Subdivision Verfahren an. Darunter versteht man Algorithmen, die jede Fläche eines Polygonnetzes in eine bestimmte Anzahl kleinerer Flächen unterteilen, so dass eine bessere Approximation glatter Oberflächen erzielt werden kann. Die verschiedenen, dazu existierenden Algorithmen unterscheiden sich im verwendeten Verfeinerungsschema, das der Unterteilung der Flächen zugrunde liegt. Die Subdivision Verfahren werden in der Regel sukzessive auf ein Polygonnetz angewendet. Durch die Anzahl der Iterationen lässt sich der Verfeinerungsgrad des Modells steuern. Problematisch ist allerdings, dass die Anzahl der Flächen dabei schnell anwächst. Klassische Algorithmen, wie etwa der von Catmull und Clark [11], führen pro Iteration zu einem Zuwachs an Dreiecken um den Faktor vier. So erlauben höher aufgelöste Modelle zwar detailliertere Korrekturen, bedeuten aber gleichzeitig auch höheren Berechnungsaufwand für die Interaktionstechniken. Daher muss auch hier ein sinnvoller Kompromiss gefunden werden. Darüber hinaus ist an den Subdivision Algorithmen problematisch, dass mit dem gewünschten, glättenden Effekt, je nach Form des Objektes, auch eine mitunter starke Reduktion des Volumens einher geht. Damit ist vor allem dann zu

rechnen, wenn das Objekt besonders gering aufgelöst ist, wodurch sich spitze Winkel auf der Oberfläche ergeben. Durch die Verfeinerung kann jedoch eine wesentlich natürlichere Approximation erreicht werden und der Volumenverlust durch einen einfachen Skalierungsschritt wieder ausgeglichen werden.

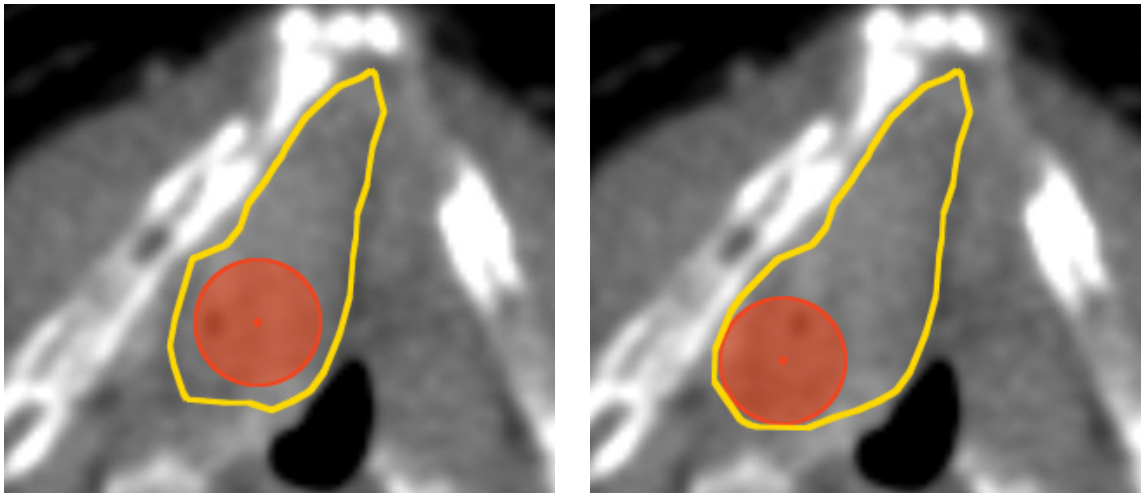
Zur Unterstützung der Nachkorrektur im Rahmen der entwickelten Interaktionstechniken wird dem Nutzer das $\sqrt{3}$-Subdivison Verfahren [34] zur Verfügung gestellt. Dieser Algorithmus unterteilt die Flächen des Modells lediglich um den Faktor drei, wodurch oftmals schon nach nur einer Iteration ein ausreichendes Ergebnis erzielt wird. Die Entscheidung darüber, zu welchem Zeitpunkt der Nachkorrektur eine Verfeinerung des Modells erforderlich ist und ob überhaupt, wird momentan dem Nutzer überlassen. So können etwa größere Deformierungen am Modell lokal eine wesentliche Vergrößerung der Dreiecke zur Folge haben oder für bestimmte Korrekturen ein höherer Detailgrad benötigt werden. In solchen Fällen ist es sinnvoll, eine Subdivision während des Korrekturvorgangs vorzunehmen. Die Verfeinerung könnte aber auch den letzten Schritt darstellen, in dem der Nutzer die Korrektur abschließt um schließlich ein glattes, optisch ansprechendes Segmentierungsergebnis zu erreichen.

5.3 Entwickelte Interaktionstechniken

5.3.1 Bulge Tool

Das erste der insgesamt vier entwickelten Korrekturwerkzeuge zur Anwendung bei medizinischen Segmentierungen ist das *Bulge Tool*. Die Grundidee dieser Interaktionstechnik besteht darin, dem Nutzer ein kugelförmiges Werkzeug zur Verfügung zu stellen, mit dem lokale Deformationen am Modell vorgenommen werden können.

Die Anwendung ist denkbar einfach. Klickt der Nutzer ins Schichtbild, erscheint zentriert an der Position des Cursors das zweidimensionale, kreisförmige Abbild der Kugel. Zusätzlich wird das Bulge Tool auch in der dreidimensionalen Visualisierung des Modells dargestellt, wodurch die Orientierung des Nutzers in beiden Ansichten weiter gefördert wird. Durch Bewegung der Maus bei gedrückter, rechter Maustaste lässt sich nun der Kugelradius bestimmen. Mit gedrückter linker Maustaste kann die Kugel durch den Datensatz bewegt werden. Stößt sie dabei gegen eine Konturfläche des Modells, führt dies zu einer lokalen Verschiebung der Kontur in Bewegungsrichtung (Abb. 5.7). Mit dem Bulge Tool kann das Modell also gewissermaßen „ausgebeult“ werden. Diese Art der Interaktion kann sich der Nutzer leicht auch als Vorgang in der „realen“ Welt veranschaulichen, in dem sich das Modell aus flexiblem Material bestehend gedacht wird, dass bei leichtem Druck nachgibt und sich lokal verformt. Daher ist zu erwarten, dass diese Interaktionstechnik eine intuitive Bedienung ohne nennenswerte Einarbeitungszeit ermöglicht.



(a) (b)

Abbildung 5.7: Nachkorrektur einer Segmentierung mit dem Bulge Tool. Die Objektkontur kann gewissermaßen „ausgebeult“ werden. (a) Vor und (b) nach der Deformation.

Für die algorithmische Umsetzung ist zunächst denkbar, dass alle Flächen F des Modells auf Schnitt mit der Kugel zu testen sind. Dazu wäre es lediglich notwendig, den Abstand zwischen den Dreiecken und dem Mittelpunkt $\vec{m}$ der Kugel zu betrachten und zu prüfen, ob dieser kleiner als der Radius r des Bulge Tools ist. Allerdings müsste dieser Vorgang bei jedem einzelnen Aufruf der Maus-Ereignisbehandlung ausgeführt werden. Für sich allein genommen, ist ein solcher Schnittest nicht sonderlich aufwändig. Ist die Berechnung jedoch während der gesamten Mausbewegung für sämtliche Dreiecke des Modells erforderlich, ist davon auszugehen, dass der flüssige Ablauf der Interaktion stark beeinträchtigt wird.

Weniger aufwändig ist es dagegen, nicht den Abstand zu den Flächen, sondern nur zu deren Vertexpositionen $\vec{v}$ zu testen. Zum einen ist diese Operation einfacher, da lediglich der euklidische Abstand zwischen $\vec{v}$ und $\vec{m}$ zu berechnen ist, zum anderen bestehen die verwendeten Modelle aus wesentlich weniger Vertices als Flächen, so dass sich auch insgesamt die Anzahl der benötigten Berechnungsschritte verringert. Die Deformierung des Modells kann dann als Verschiebung betroffener Vertices nach Gleichung 5.3 berechnet werden.

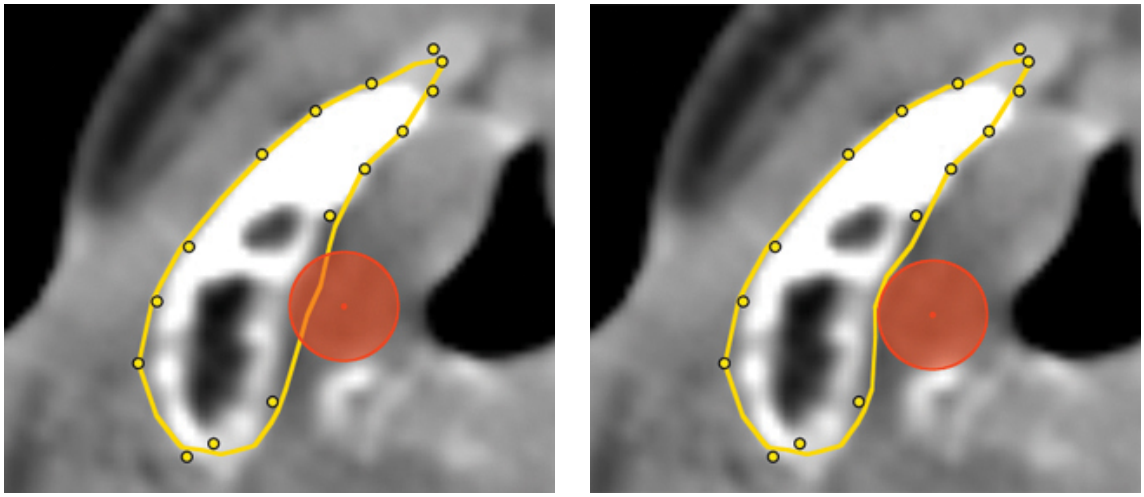
$$\vec{v}\,' = \vec{m} + r\,\frac{\vec{v} - \vec{m}}{\|\vec{v} - \vec{m}\|} \tag{5.3}$$

Die auf diese Weise erzielte Modellmanipulation ist keine exakte Simulation der zur Veranschaulichung gedachten, „realen“ Situation. Dafür müssten tatsächlich die Berührungspunkte der Kugel mit den einzelnen Dreiecken errechnet und die entsprechenden Veränderungen der Vertexpositionen abgeleitet werden. Allerdings kann durch den vorgeschlagenen, einfacheren Algorithmus eine ausreichende Annäherung erreicht und allem voran eine wesentlich flüssigere Interaktion gewährleistet werden.

Ein Problem ergibt sich bei dieser Herangehensweise jedoch, wenn das zu deformierende Polygonmodell zu gering aufgelöst ist, so dass die Abstände benachbarter Vertices untereinander sehr groß im Vergleich zum Kugelradius sind. In solch einem Fall ist es möglich, dass sich Kugel und Kontur im Schichtbild großflächig schneiden, aber nur ein Vertex des Dreiecks innerhalb des Radius liegt. Zwar kommt es trotzdem zu einer Deformation, die jedoch nicht mit dem gewünschten Interaktionsziel übereinstimmt. Eigentlich erwartet der Nutzer vor allem die korrekte Verschiebung der im Schichtbild berührten Liniensegmente, die sich als Durchschnitt der Konturflächen mit der Bildebene ergeben. Im ungünstigsten Fall befindet sich überhaupt keiner der Vertices einer betroffenen Fläche im Radiusbereich, wodurch auch keine Deformation berechnet wird. Eine solche Situation ist in Abbildung 5.8a dargestellt. Zusätzlich zur Kontur der Segmentierung sind dort die Vertices visualisiert, deren Abstand in z-Richtung kleiner bzw. gleich dem Radius ist. Wie zu erkennen ist, befinden sich sämtliche Vertices außerhalb der Reichweite des Bulge Tools. Das in diesem Fall eigentlich vom Nutzer verfolgte Interaktionsziel zeigt Abbildung 5.8b.

Zur Umgehung solch unerwünschten Verhaltens bieten sich prinzipiell zwei Möglichkeiten an. Eine Option besteht darin, ganz einfach die Auflösung des Modells zu erhöhen. Dies kann zum Beispiel durch die bereits erwähnten Subdivision Algorithmen erreicht werden. Im Ergebnis verringern sich die Flächeninhalte der Dreiecke, so dass die Wahrscheinlichkeit für Situationen wie in Abbildung 5.8 abnimmt. Ein Vorteil ist zudem, dass sich somit auch wesentlich detailiertere Deformierungen vornehmen lassen. Die einerseits vorteilhafte Tatsache der signifikant erhöhten Vertexzahl wird jedoch mit dem Nachteil erkauft, einen höheren Berechnungsaufwand für das Bulge Tool zu verursachen.

Die andere Möglichkeit besteht darin, den Algorithmus so anzupassen, dass auch in „problematischen“ Situationen das gewünschte Verhalten erzielt wird. Um dies zu gewährleisten, ist es erforderlich neben den Vertices auch die Konturflächen einzubeziehen. Wie weiter oben bereits diskutiert, kommen Schnittberechnungen mit sämtlichen Dreiecken des Modells dabei nicht in Frage. Da sich das Bulge Tool gerade deshalb problematisch verhält, weil der Kugelradius klein, die Flächeninhalte im Vergleich dazu aber groß sind, sind in solchen Situationen jedoch überhaupt nur wenige Dreiecke von der Deformation betroffen. Wird außerdem berücksichtigt, dass sich der Eindruck der unerwünschten Interaktionsauswirkung vor allem auf die Darstellung des Schichtbildes bezieht, in der die Interaktion stattfindet, lässt sich



(a) (b)

Abbildung 5.8: Problematisches Verhalten des Bulge Tools. Zusätzlich zur Objektkontur sind die Vertices dargestellt. (a) Kein Vertex liegt innerhalb der Reichweite des Bulge Tools, so dass der entsprechende Konturbereich nicht deformiert wird. (b) Die vom Nutzer erwartete Interaktionsauswirkung.

der Suchraum der infrage kommenden Dreiecke wesentlich eingrenzen. Somit ist es lediglich notwendig jene Konturflächen zu betrachten, die von der betreffenden Bildebene $\mathbb{P}$ geschnitten werden.

Die Bestimmung dieser Teilmenge braucht nur zu dem Zeitpunkt erfolgen, an dem der Nutzer die Maustaste drückt und somit die Interaktion einleitet. Eine weitere Effizienzsteigerung ergibt sich daraus, dass der Durchschnitt aus Dreieck und Bildebene einem Liniensegment[7] entspricht, weshalb die Abstandsbestimmung auf die Berechnung der Entfernung zwischen eben diesem und dem Mittelpunkt der Kugel reduziert werden kann. Wie bereits erwähnt, ist die Anpassung des Algorithmus dann erforderlich, wenn das Modell aus (im Verhältnis zum Radius) großen Dreiecken besteht. Daher ist es zudem sinnvoll, den Berührungspunkt des Bulge Tools mit dem Durchschnitt von Dreieck und Bildebene zu bestimmen. Dies wird durch Projektion des Mittelpunktes der Kugel auf das Liniensegment erreicht. Wird der resultierende Punkt nun in baryzentrischen Koordinaten bezüglich des Dreiecks dargestellt, lässt sich daraus eine sinnvolle Gewichtung für die Verschiebung der Vertices ableiten. Somit wird die Deformierung vor allem auf Vertices angewendet, die sich nahe des Berührungspunktes befinden. Werden benachbarte Konturflächen vom Bulge Tool

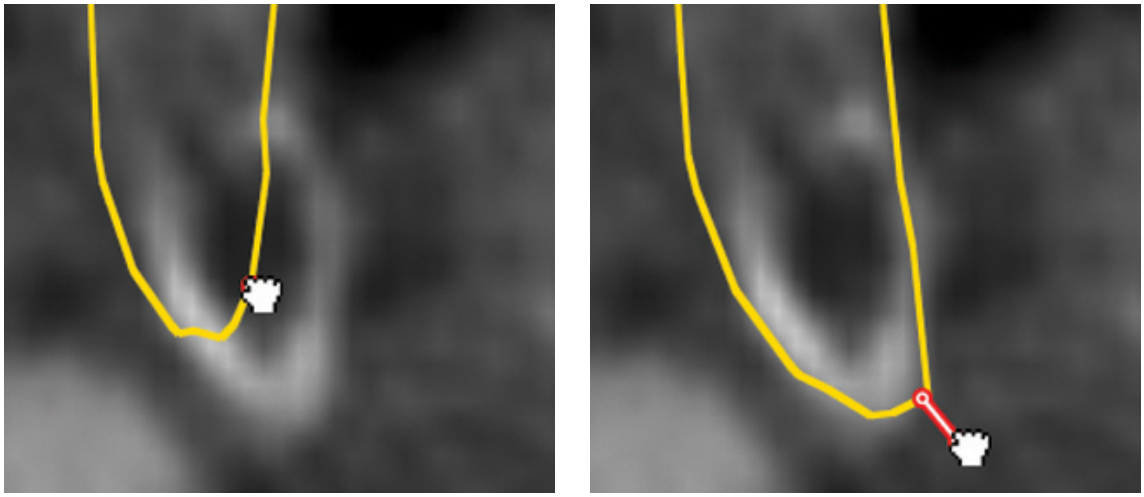
[7] Fälle, in denen die Dreiecke vollständig in der Ebene liegen bzw. deren Durchschnitt genau einem Vertex entspricht sind hier nicht von Bedeutung.

berührt, ergeben sich die Deformationsparameter gemeinsamer Vertices als arithmetisches Mittel der pro Dreieck berechneten Verschiebevektoren. Um aber auch solche Vertices einzubeziehen, die keiner der betrachteten Konturflächen zugehören, für die aber dennoch $\|\vec{v}-\vec{m}\| < r$ gilt, wird schließlich die Verschiebung nach Gleichung 5.3 berechnet. Das gesamte Verfahren als Pseudocode ist in Algorithmus 5.2 nachzulesen.

Algorithmus 5.2 Deformation beim Bulge Tool.

for all $F \in \mathsf{intersection}(\mathcal{F}, \mathbb{P})$ **do**
  $\overline{i_1 i_2} \leftarrow \mathsf{intersection}(F, \mathbb{P})$
  if ($\mathsf{distance}(\vec{m}, \overline{i_1 i_2}) < r$) **then**
    $\vec{p} \leftarrow \mathsf{projection}(\vec{m}, \overline{i_1 i_2})$
    $\vec{b} \leftarrow \mathsf{barycentric}(\vec{p}, F)$
    $j \leftarrow 0$
    for all $\vec{v} \in \mathsf{vertices}(F)$ **do**
      $move[\vec{v}] \leftarrow move[\vec{v}] + (\vec{b}[j] \,/\, \mathsf{max}(\vec{b})) * (r * (\vec{p}-\vec{m}) \,/\, \|\vec{p}-\vec{m}\| - (\vec{p}-\vec{m}))$
      $count[\vec{v}] \leftarrow count[\vec{v}] + 1$
      $j \leftarrow j + 1$
    end for
  end if
end for

for all $\vec{v} \in \mathcal{V}$ **do**
  if ($count[\vec{v}] > 0$) **then**
    $\vec{v} \leftarrow \vec{v} + move[\vec{v}] \,/\, count[\vec{v}]$
  else
    if ($\|\vec{v}-\vec{m}\| < r$) **then**
      $\vec{v} \leftarrow \vec{m} + r * (\vec{v}-\vec{m}) \,/\, \|\vec{v}-\vec{m}\|$
    end if
  end if
end for



(a) (b)

Abbildung 5.9: Nachkorrektur einer Segmentierung mit dem Traction Tool. (a) Der Nutzer „greift“ den zu korrigierenden Teil der Kontur und (b) zieht daran, bis die gewünschte Form erreicht ist.

5.3.2 Traction Tool

Als nächste der entwickelten Interaktionstechniken wird im Folgenden das *Traction Tool* vorgestellt. Dieses Korrekturwerkzeug basiert auf dem Laplacian Modeling Framework, weshalb es den Nutzer in die Lage versetzt Deformationen unter Aufrechterhaltung von Formeigenschaften durchzuführen. Die Interaktion hierbei verläuft ähnlich einfach wie beim Bulge Tool. Statt jedoch gegen die Kontur zu schieben, wird mit dem Traction Tool daran gezogen. Dafür kommt das in der Polygonmodellierung verbreitete Handle-Konzept zum Einsatz. Da allerdings anders als bei den üblichen FFD-Methoden nicht ein Hüllgitter, sondern das Modell selbst „angefasst“ wird, ist auch für das Traction Tool eine intuitive Bedienbarkeit zu erwarten. Um die Metapher des Anfassens und der Modellierung mit eigener Hand noch stärker zu verdeutlichen, wird als Mauszeiger bei diesem Korrekturwerkzeug das Icon einer greifenden Hand verwendet.

Für die Anwendung klickt der Nutzer einfach auf den Teil der Kontur, an dem er „ziehen“ möchte. Mit gedrückt gehaltener, linker Maustaste kann nun durch Bewegung der Maus die Deformation herbeigeführt werden. Da die Modellmanipulation unter Berücksichtigung lokaler Formeigenschaften stattfindet, wird die deformierte Kontur in Richtung des Mauszeigers gezogen, in vielen Fällen jedoch einen gewissen Abstand zu diesem gewinnen. Dies mag den Nutzer zunächst verwundern, resultiert aber aus der Tatsache, dass die aktuelle Cursorposition nur eine unter vielen Bedingungen im LMF Gleichungssystem darstellt. Um das Verhalten für den Anwen-

der plausibler erscheinen zu lassen, wird zusätzlich ein Handle visualisiert, dass den Mauszeiger mit der Kontur verbindet. So entsteht der Eindruck, an einer Art elastischem, an der Kontur befestigten Griff zu ziehen (Abb. 5.9).

Darüber hinaus bietet das Traction Tool eine zweite Art der Interaktion. Durch Doppelklick auf einen Bildpunkt wird der nächstgelegene Teil der Kontur in die Nähe dieser Position bewegt. Dies führt zum gleichen Resultat, als wäre der nächstgelegene Konturpunkt wie vorher beschrieben an die Zielposition gezogen worden. Um auch bei dieser Variante die Art der Interaktion zu symbolisieren, wird der Mauszeiger als zeigende Hand dargestellt. Somit wird zudem unmittelbar deutlich, in welchem Modus sich das Traction Tool befindet. Während sich die Ausführung beider Modi geringfügig unterscheidet, ist der zugrunde liegende Mechanismus identisch. Die Doppelklick-Variante bietet sich besonders dann an, wenn ein sehr hoch aufgelöstes Polygonmodell vorliegt. Zwar erlauben die optimierten LMF Algorithmen eine zügige Lösung des LGS, dennoch kann es bei sehr feinen Modellen (vor allem auch durch die ständige Aktualisierung der 3D-Darstellung) dazu kommen, das der Deformationsvorgang weniger flüssig als vom Nutzer gewünscht abläuft. Letztlich bleibt es auch eine Frage des Geschmacks, welche Variante den Anwender mehr anspricht.

Wie sich die Deformation beim Traction Tool genau verhält, ist maßgeblich von der ROI abhängig. Deren Bestimmung erfolgt zwar implizit, die maximale Ausdehnung allerdings, wird durch Definition eines Radius r vom Nutzer vorgegeben. Dabei wird ähnlich wie beim Bulge Tool verfahren, indem der Nutzer durch Bewegung der Maus bei gedrückter rechter Maustaste die Größe eines, an Cursorposition zentrierten Kreises bestimmt. Alternativ dazu stehen drei vordefinierte, mit *S*, *M* und *L* bezeichnete[8] Radiusgrößen zu Verfügung, die der Nutzer durch Radio-Buttons oder Tastenkombination auswählen kann. Diese Default-Werte ergeben sich als prozentuale Anteile der maximalen Ausdehnung des Polygonmodells bezüglich dessen Mittelpunktes. Insgesamt lässt sich sagen, dass kleine ROIs lokale Deformationen ermöglichen, während mit wachsender ROI-Größe lokale Formeigenschaften aufrecht erhalten und entsprechend größere Deformationen durchgeführt werden können.

Die eigentliche Bestimmung der ROI erfolgt, nachdem per Mausklick auf die Kontur der Zugpunkt $\vec{g}$ definiert wurde. Dieser entspricht der Projektion der Cursorposition auf das, innerhalb eines Toleranzbereichs am nächsten gelegene Dreieck[9] $F_{\vec{g}}$. Anschließend wird der Vertex $\vec{v_0}$ des Dreiecks $F_{\vec{g}}$ bestimmt, dessen Abstand zu $\vec{g}$ am geringsten ist. Dieser Vertex wird nun als Saatpunkt für einen Flutungsalgorithmus verwendet, wobei all jene besuchten Vertices in die ROI aufgenommen werden, deren Abstand zu $\vec{g}$ kleiner bzw. gleich r ist (Algorithmus 5.3).

[8] *Small*, *Medium*, *Large*

[9] bzw. auf das Liniensegment $\overline{i_1 i_2}$, das sich als Durchschnitt von $F_{\vec{g}}$ und der Bildebene $\mathbb{P}$ ergibt

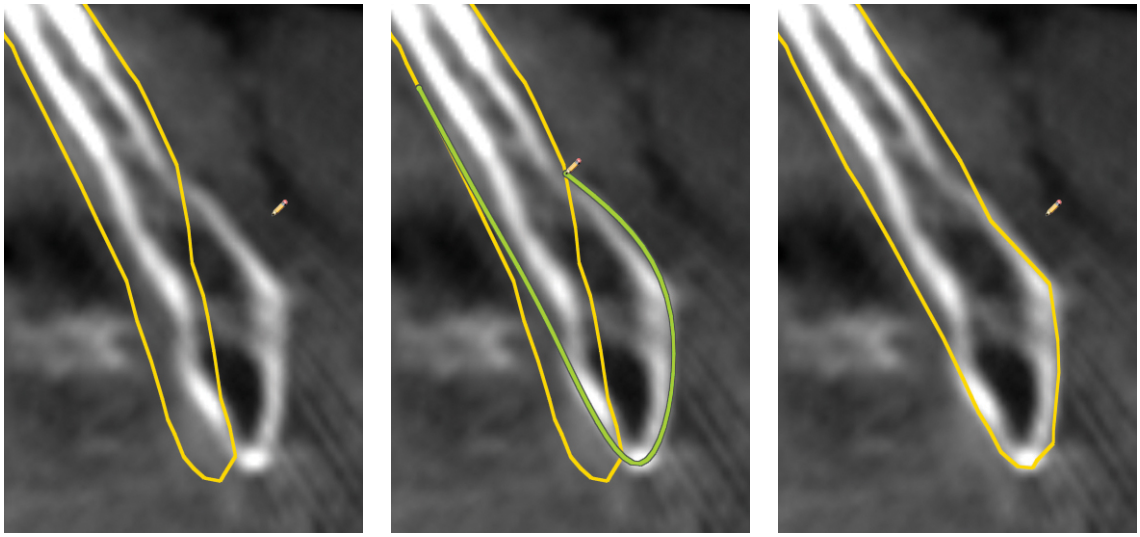
Algorithmus 5.3 Bestimmung der ROI beim Traction Tool.

$\vec{v}_0 \leftarrow \mathsf{argmin}_{\vec{v} \in \mathsf{vertices}(F_{\vec{g}})}(\|\vec{v} - \vec{g}\,\|)$
$stack.\mathsf{push}(\vec{v}_0)$
while ($stack.\mathsf{is_not_empty}()$) **do**
  $\vec{v}_0 \leftarrow stack.\mathsf{pop}()$
  $ROI \leftarrow ROI \cap \vec{v}_0$
  for all $\vec{v} \in \mathsf{neighbors}(\vec{v}_0)$ **do**
    if ($(\|\vec{v} - \vec{g}\,\| \leq r) \wedge (\vec{v} \ni ROI)$) **then**
      $stack.\mathsf{push}(\vec{v}\,)$
    end if
  end for
end while

Aus der so gewonnenen Vertex-Teilmenge des Modells wird anschließend, wie in Kapitel 5.2.3 beschrieben, das LMF Gleichungssystem aufgebaut. Als Deformationsbedingung werden die baryzentrischen Koordinaten von $\vec{g}$ bezüglich $F_{\vec{g}}$ in die Koeffizientenmatrix aufgenommen. Dann wird die Faktorisierung der Matrix berechnet. Die Deformation ergibt sich schließlich aus der Lösung des LGS, die nach jeder Mausbewegung durch Rücksubstitution neu berechnet wird, wobei jeweils die letzte Zeile der rechten Seite durch die aktuelle Cursorposition ersetzt wird.

5.3.3 Sketch Tool

Ebenfalls auf dem LMF basierend ist das *Sketch Tool*. Die hierfür verwendete Interaktion erinnert stark an die manuelle Segmentierung, denn zur Anwendung muss der Nutzer zu korrigierende Teile der Modellkontur nachzeichnen. Um möglichst einfach und schnell vor allem das Zeichnen glatter Kurven zu ermöglichen, brauchen anders als bei der bisherigen Nachkorrektur, nur einige wenige Punkte der Kontur angegeben zu werden. Die gesamte Kurve wird dann als Interpolation zwischen den Stützpunkten generiert. So dürfte auch dieses Korrekturwerkzeug vom Nutzer grundsätzlich als leicht verständlich angesehen werden. Zum einen weil die Art der Interaktion der Bisherigen ähnelt, zum anderen weil das Überzeichnen der fehlerhaften Kontur eine besonders naheliegende Herangehensweise für die Nachkorrektur darstellt. Um dem Nutzer die Anwendung direkt zu verdeutlichen, wird als Mauszeiger beim Sketch Tool das Piktogramm eines Stiftes verwendet. Abbildung 5.10



(a) (b) (c)

Abbildung 5.10: Nachkorrektur einer Segmentierung mit dem Sketch Tool. (a) Ausgangssituation mit Fehlsegmentierung, (b) mit dem Sketch Tool wurde die korrekte Kontur nachgezeichnet, (c) resultierende Deformierung des Modells.

zeigt den Einsatz des Werkzeugs zur Nachkorrektur. Die grün dargestellte Kontur ist die vom Nutzer gezeichnete Zielkurve, für die lediglich fünf Mausklicks benötigt wurden. Bei vergleichsweise geringem Interaktionsaufwand können mit dem Sketch Tool also nicht-triviale Deformationen erreicht werden, die wie in der Abbildung zur korrekten Segmentierung führen.

Auch wenn die Interaktionstechnik an die manuelle Segmentierung erinnert, zwischen den Interaktionsauswirkungen besteht ein großer Unterschied. Denn - wie bei allen im Rahmen dieser Arbeit entwickelten Werkzeugen - resultiert die Interaktion beim Sketch Tool in einer dreidimensionalen Manipulation des Segmentierungsmodells. Obwohl die Kontur lediglich im aktuell betrachteten Schichtbild überzeichnet wird, hat die Manipulation Einfluss auf benachbarte Schichten. Wie weit dieser Einfluss genau reicht, kann durch den bereits vorgestellten Depth Limiter bestimmt werden.

Ein weiterer Unterschied zur bisherigen Nachkorrektur besteht darin, dass die gezeichnete Zielkurve lediglich als Skizze der Segmentierungskontur zu verstehen ist. Dies resultiert direkt aus der Anwendung des LMF. Zwar werden beim Sketch Tool wesentlich mehr Positionsbedingungen in das LGS aufgenommen als beim Traction Tool, jedoch weiterhin auch die Aufrechterhaltung der Laplace-Vektoren angestrebt. Zudem hängt die genaue Gestalt der resultierenden Modellkontur auch wesentlich vom Detailgrad des Modells ab. Während die gezeichnete Kurve durch kubische

Spline-Interpolation entsteht, ergibt sich die Modellkontur im Schichtbild als mehr oder weniger grobe, lineare Approximation einer gekrümmten Oberfläche. So steigt mit der Auflösung des Modells auch die Güte der Annäherung an die skizzierte Kontur, gleichzeitig allerdings auch der zur Deformation benötigte Rechenaufwand.

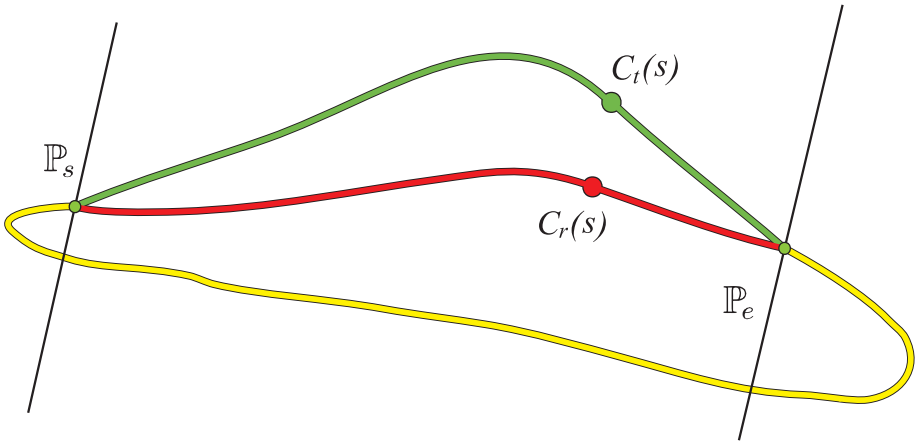


Abbildung 5.11: Geometrische Situation beim Sketch Tool. Die grüne Kurve ist die vom Nutzer gezeichnete Zielkurve $C_t(s)$. Die gelbe und rote Kurve stellen zusammen die segmentierte Objektkontur dar, während der rote Abschnitt, der als Referenzkurve $C_r(s)$ gefundene Pfad ist. Die beiden schwarzen Linien symbolisieren die senkrecht auf dem Schichtbild stehenden Grenzebenen der ROI. Zusätzlich sind zwei korrespondierende Kurvenpunkte dargestellt.

Ausschlaggebend für das Sketch Tool ist vor allem die Information darüber, welcher Teil der segmentierten Kontur fehlerhaft und deshalb vom Nutzer überzeichnet wurde. Dies wird automatisch nach Eingabe der Zielkurve $C_t(s)$ bestimmt. Dazu werden zunächst deren Start- und Endpunkt auf das, im Schichtbild dargestellte Liniensegment der jeweils nächstgelegenen Modellfläche F_s bzw. F_e projiziert, um sicherzustellen, dass die Kurve exakt an der Modellkontur ansetzt. Befinden sich innerhalb eines Toleranzbereiches keine Flächen aus dem Durchschnitt von Modell und Bildebene, ist die Interaktion ungültig und wird abgebrochen. Andernfalls wird die Kurve anschließend in eine parametrische Repräsentation überführt, so dass jeder Kurvenpunkt über einen Parameter $0 \leq s \leq 1$ darstellbar ist.

Um nun den fehlerhaften Teil der Kontur und damit die Referenz- zur Zielkurve zu finden, wird auf den, von der aktuellen Bildebene geschnittenen Flächen ein Pfad von F_s nach F_e gesucht. Sind diese beiden Liniensegmente miteinander verbunden und stellt sich das Modell im Schichtbild als Menge geschlossener Konturen dar, dann existieren jedoch genau zwei solcher Pfade. Aus dieser Ambivalenz heraus ergibt sich die Frage, welcher der Pfade nun die Referenzkurve repräsentiert. Für das Sketch Tool wird dabei davon ausgegangen, dass genau jene Kurve als Referenz

herangezogen wird, welche die größere Ähnlichkeit zur Zielkurve aufweist. Dafür werden aus den Pfaden stückweise lineare Kurven generiert und eine Reihe von korrespondierenden Kurvenpunkten bestimmt. Die Ähnlichkeit zur Zielkurve ergibt sich dann als Summe der quadratischen Differenzen dieser Kurvenpunkte.

Anhand des so gefundenen, fehlerhaften Bereichs der Modellkontur kann nun die ROI für die Deformation bestimmt werden. Wie auch beim Traction Tool geschieht dies durch einen Flutungsalgorithmus. Als Saatpunkt wird hier der erste Vertex der Fläche F_s herangezogen. Begrenzt wird die ROI beim Sketch Tool durch vier Ebenen. Zwei von diesen befinden sich parallel zur Bildebene und werden durch Anwendung des Depth Limiter definiert. Die anderen beiden $\mathbb{P}_s$ und $\mathbb{P}_e$ liegen jeweils senkrecht zur Referenzkurve und verlaufen durch deren Start- bzw. Endpunkt. Abbildung 5.11 veranschaulicht noch einmal die geometrische Situation.

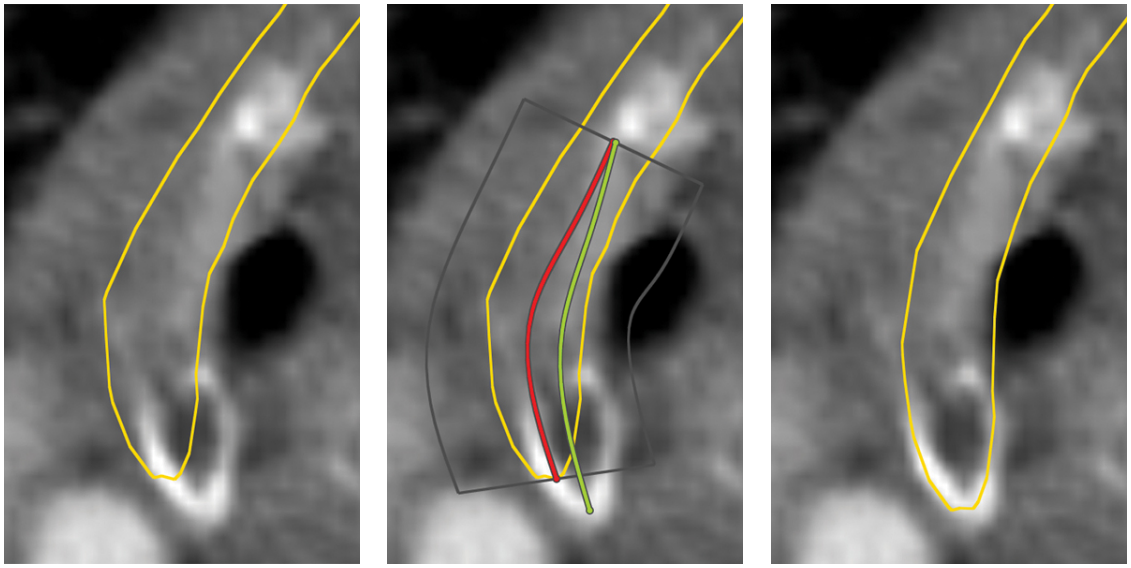
Da auch beim Sketch Tool die Deformation mit Hilfe des LMF berechnet wird, muss aus den Vertices der ROI zunächst die erweiterte Laplace-Matrix aufgebaut werden. Die zusätzlichen Positionsbedingungen werden im Anschluss daran, wie auch beim Traction Tool als On-Face Constraints in das LGS aufgenommen. Dazu werden Referenz- und Zielkurve an einer bestimmten Anzahl von Punkten abgetastet, die sich jeweils im Abstand d_C zueinander befinden. Dieser Parameter muss gegenwärtig noch vom Nutzer definiert werden und stellt daher einen Nachteil für das Sketch Tool dar. Der Deformationsvorgang wird schließlich wie bei der Anwendung des LMF üblich, durch die Lösung des Gleichungssystems abgeschlossen.

5.3.4 Bending Tool

Das als nächstes vorgestellte Korrekturwerkzeug ist das *Bending Tool*. Diese Interaktionstechnik zur Modellmanipulation ist an [33] angelehnt, wo sie jedoch hauptsächlich für die Animation von Charaktermodellen genutzt wird. Um sie für den Einsatz zur Nachkorrektur medizinischer Segmentierungen anwenden zu können, sind daher einige Modifikationen erforderlich.

Wie auch beim Sketch Tool wird beim Bending Tool die Modellmanipulation durch das Zeichnen von Kurven vorgenommen. Der Unterschied besteht allerdings darin, dass die Referenzkurve nicht implizit aus der Objektkontur hervorgeht, sondern vielmehr explizit vom Nutzer als annähernde Skelettlinie des Objektes eingezeichnet wird[10]. Dies stellt den ersten Schritt der Interaktion dar. Im zweiten Schritt wird die eigentliche Deformation eingeleitet, indem eine weitere Kurve gezeichnet und somit eine Abbildung von der Referenz- auf diese Zielkurve definiert wird. Anders

[10]Die Skelettlinie als Referenzkurve zu verwenden ist besonders sinnvoll, jedoch nicht zwangsläufig erforderlich.



(a) (b) (c)

Abbildung 5.12: Nachkorrektur einer Segmentierung mit dem Bending Tool. (a) Ausgangssituation mit Fehlsegmentierung. (b) Mit dem Bending Tool wurden Referenz- (rot) und Zielkurve (grün) eingezeichnet. Die dunkelgraue Kontur begrenzt die ROI. (c) Resultierende Deformierung des Modells.

als in [33] ist es hierbei möglich, die beiden Kurven in verschiedenen Ebenen derselben Schnittführung zu zeichnen; mit dem Resultat, dass betroffene Vertices um die Differenz der Schnittachsenkoordinate der Kurven in senkrecht zur Bildebene verschoben werden. Die Deformation beim Bending Tool wird lediglich auf Grundlage dieser Abbildung berechnet und nicht etwa, wie bei den vorherigen beiden Korrekturwerkzeugen mit Hilfe des LMF. In Abbildung 5.12 ist die Anwendung der Interaktionstechnik dargestellt.

Die Eingabe der Kurven erfolgt in [33] als Freihandzeichnung, die anschließend einer Glättung unterzogen wird. Dieses Vorgehen ist für die dort beschriebene Anwendung angemessen, da in erster Linie Deformationen für die Animation generiert werden und die Eingabe hauptsächlich mit Tablett und Stift erfolgt. Beim Einsatz des Bending Tools dagegen, soll eine (je nach Korrekturfall) mehr oder minder exakte Anpassung des Polygonmodells an einen anatomischen Datensatz erzielt werden. Um dies für den Nutzer zu vereinfachen, werden wie auch schon beim Sketch Tool kubische Spline-Kurven verwendet, die lediglich die Eingabe weniger Stützpunkte erfordern.

Ein weiterer Unterschied besteht zudem darin, dass die in [33] deformierten Modellteile in der Regel Gliedmaßen sind und damit eine annähernd zylindrische Form aufweisen. Zum einen ist es für solch geformte Objekte besonders einfach, eine eindeutige Skelettlinie zu definieren. Zum anderen kann die Bestimmung einer sinn-

vollen ROI implizit vorgenommen werden, da sich die Objekte gut vom übrigen Modell abgrenzen lassen. Durch Begrenzung der ROI mit den beiden Ebenen, die im Start- und Endpunkt senkrecht zur Kurve verlaufen, können die betreffenden Vertices durch einen Flutungsalgorithmus gefunden werden. Für die Korrektur von Segmentierungsmodellen hat man es jedoch oftmals mit einer anderen Situation zu tun. Die Form der zu korrigierenden Modellbereiche ist selten zylindrisch und eine eindeutige Trennung vom Gesamtmodell in den meisten Fällen nicht ohne weiteres möglich.

Daher wird beim Bending Tool die Definition der ROI explizit durch den Nutzer vorgenommen. Zur Begrenzung des Einflussbereichs innerhalb der Ebene wird dabei folgendermaßen vorgegangen. Durch Bewegung der Maus bei gedrückter Alt- und rechter Maustaste kann die Ausdehnung d_{ROI} einer geschlossenen Hüllkurve bestimmt werden, die die Referenzkurve umgibt. Ein Vertex wird genau dann zur ROI gezählt, wenn seine Projektion auf die Bildebene innerhalb dieser Hülle liegt. Dazu werden die Normalen der Referenzkurve in den Stützpunkten bestimmt. Aus den Punkten, die beiderseits der Kurve im Abstand d_{ROI} auf den Normalen liegen, werden zwei weitere spline-interpolierte Kurven generiert, die im Start- und Endpunkt der Referenzkurve durch Liniensegmente zur geschlossenen Hüllkurve verbunden werden. Abbildung 5.13 zeigt eine schematische Darstellung dieses Sachverhaltes. Der Einflussbereich senkrecht zur Bildebene wird wie gewohnt mit Hilfe des Depth Limiters beschränkt. Durch einen Flutungsalgorithmus können nun alle benachbarten Vertices gefunden werden, die der ROI Bedingung genügen.

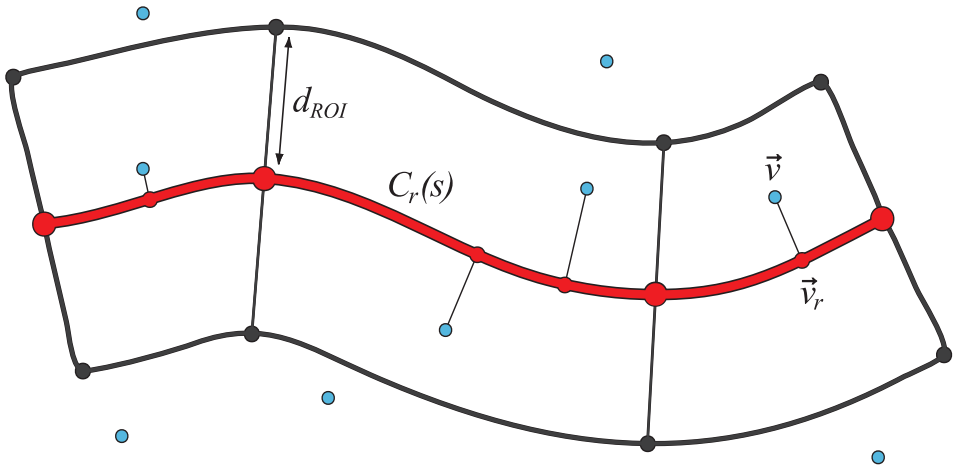


Abbildung 5.13: Bestimmung der ROI beim Bending Tool. Die geschlossene Hüllkurve wird aus Punkten gebildet, die im Abstand d_{ROI} senkrecht zu den Stützpunkten der Referenzkurve liegen. Nur die Vertices, deren Projektion auf die Bildebene im Inneren dieser Hülle liegt, gehören der ROI an und werden für die Deformation mit dem jeweils nächsten Punkt der Referenzkurve assoziiert.

Wesentlich für den Deformationsvorgang ist, dass jeder Vertex $\vec{v}$ innerhalb der ROI mit dem nächstgelegenen Punkt $\vec{v}_r$ auf der Referenzkurve assoziiert wird. Dazu wird dieser Punkt zunächst für die lineare Interpolation der Referenz-Stützpunkte bestimmt. Dieses Zwischenergebnis stellt bereits eine gute Approximation des tatsächlich gesuchten Punktes dar, so dass nur wenige Iterationen des Algorithmus nach [50] genügen, um den tatsächlich gesuchten $\vec{v}_r$ auf der spline-interpolierten Referenzkurve zu finden. Da, wie auch beim Sketch Tool, die Kurven als parametrische Repräsentationen $C(s)$ vorliegen, kann jedem Vertex $\vec{v}$ ein Parameter $s(\vec{v})$ zugeordnet werden, mit dem die Koordinaten des Punktes $C_r(s(\vec{v})) = \vec{v}_r$ berechnet werden können. Mit $s(\vec{v})$ lässt sich nun auch der korrespondierende Punkt $\vec{v}_t$ auf der Zielkurve bestimmen.

Für den Ablauf der Modellmanipulation kann sich der Nutzer vorstellen, dass die ROI-Vertices über starre Verbindungen an die Referenzkurve gekoppelt sind. Die Referenzkurve wird nun derart „gebogen“ und „verzogen“, dass sie die Gestalt der Zielkurve annimmt. Wegen der starren Verbindungen werden die Vertices entsprechend mit bewegt, woraus die Deformation des Polygonmodells resultiert. Mathematisch ausgedrückt, wird $(\vec{v} - \vec{v}_r)$ im Winkel θ zwischen der Tangente von C_r in $\vec{v}_r$ und der Tangente von C_t in $\vec{v}_t$ um die Schnittachse rotiert und schließlich mit $\vec{v}_t$ addiert (Algorithmus 5.4).

Anders als bei den LMF-basierten Verfahren, muss beim Bending Tool gesondert darauf geachtet werden, dass an den Grenzbereichen der ROI keine ungewollten Verzerrungen des Polygonnetzes auftreten. Deshalb werden folgende Maßnahmen ergriffen. Bevor es überhaupt zur Deformation kommt, wird die Zielkurve so verschoben, dass deren Startpunkt mit dem der Referenzkurve übereinstimmt. Um weiter einen möglichst glatten Übergang zu den benachbarten Bereichen der ROI in Richtung der Schnittachse zu erreichen, wird der Einfluss der Deformation gewichtet: Vertices in Nähe der Bildebene werden stark deformiert, während solche in Nähe der Grenzebenen fast unverändert bleiben. Im Folgenden wird sich der Einfachheit halber nur auf axiale Schichtbilder bezogen. Diese liegen in der xy-Ebene und deren Schnittachse entspricht der z-Achse. Für sagittale und koronare Schichtbilder müssen die Achsenbezeichnungen entsprechend vertauscht werden.

Sei $z_{\vec{v}}$ die z-Koordinate eines Vertex $\vec{v}$, $z_{\mathbb{P}}$ die z-Koordinate der Bildebene $\mathbb{P}$ und z_{max} der Abstand der Grenzebenen zu $\mathbb{P}$ auf der z-Achse. Die Gewichtung $w(\vec{v})$ für die Deformation berechnet sich dann nach der Vorschrift:

$$w(\vec{v}) = \frac{1 + \cos(\pi d)}{2}, \qquad \text{mit } d = \frac{|z_{\vec{v}} - z_{\mathbb{P}}|}{z_{max}}. \tag{5.4}$$

Algorithmus 5.4 Deformation beim Bending Tool.

for all $\vec{v} \in ROI$ **do**

$$
\begin{aligned}
s &\leftarrow \mathsf{argmin}_{0 \le p \le 1}(\, \| \vec{v} - C_r(p) \| \,) \\
\theta &\leftarrow \angle(\vec{\nabla} C_r(s), \vec{\nabla} C_t(s)) \\
\vec{v}' &\leftarrow \mathsf{translation}(\vec{v}, -C_r(s)) \\
\vec{v}' &\leftarrow \mathsf{rotation}_z(\vec{v}', \theta) \\
\vec{v}' &\leftarrow \mathsf{translation}(\vec{v}', C_t(s)) \\
\vec{v} &\leftarrow (1 - w(\vec{v})) * \vec{v} + w(\vec{v}) * \vec{v}'
\end{aligned}
$$

end for

5.4 Zusammenfassung

In diesem Kapitel wurde der Entwurf der entwickelten Interaktionstechniken behandelt. Dabei wurden zunächst allgemeine Konzepte betrachtet, mit denen die Nachkorrektur unterstützt werden soll. Um die dreidimensionalen Modellmanipulationen vor allem mit Bezug auf die zweidimensionalen Schichten des Datensatzes effizient zu gestalten, wurde der Multi Slice Viewer entwickelt. Dieser ermöglicht es dem Nutzer drei verschiedene, benachbarte Schichtbilder einer Schnittführung simultan zu betrachten um die Interaktionsauswirkungen besser einschätzen und gegebenenfalls umgehend revidieren zu können. Damit der Einfluss einer Korrektur in Richtung der Schnittachse bequem reguliert werden kann, wurde außerdem mit dem Depth Limiter ein entsprechendes Hilfsmittel entwickelt, mit dem unabhängig vom verwendeten Korrekturwerkzeug zwei Grenzebenen definiert werden können. Darüber hinaus wurde vorgestellt, wie mit Hilfe spezieller Matrixrepräsentationen bei der Berechnung für die LMF-Deformationen möglichst effizient vorgegangen werden kann. Anschließend wurde der Detailgrad der Segmentierungmodelle besprochen sowie die Vor- und Nachteile unterschiedlich hoher Auflösungen diskutiert. Um dem Nutzer die Kontrolle über den Detailgrad zu ermöglichen, wurde vorgeschlagen zusätzlich zu den Korrekturwerkzeugen ein Subdivision Verfahren zur Verfügung zu stellen.

Schließlich wurden die eigentlichen Konzepte der Korrekturwerkzeuge vorgestellt. Das Bulge Tool ist ein kugelförmiges Werkzeug, mit dem die Objektkontur gewissermaßen ausgebeult werden kann. Die Deformation wird dabei als Verschiebung berührter Flächen bzw. Vertices, entsprechend der Größe der verwendeten Kugel realisiert. Beim Traction Tool kann der Nutzer das Modell verformen, indem er daran zieht. Da hierbei das LMF zum Einsatz kommt, werden während der Deformation Formeigenschaften beibehalten und somit auf Modellwissen zurückgegriffen.

Zur Anwendung des Sketch Tools werden fehlerhafte Teile der Modellkontur einfach überzeichnet. Die resultierende Deformationen wird auch hier mittels LMF berechnet. Beim Bending Tool zeichnet der Nutzer zwei Spline-Kurven in das Schichtbild: Eine, um die momentane Situation (etwa als Hauptskelettlinie) zu repräsentieren sowie eine andere, die den gewünschte Zustand nach der Korrektur darstellt. Die Deformation des Modells ergibt sich dann anhand der Abbildung von der Referenz- auf die Zielkurve. Im nächsten Kapitel geht es nun um die prototypische Implementierung der Interaktionstechniken.

6 Implementierung

Die im vorigen Kapitel entworfenen Interaktionstechniken zur Nachkorrektur medizinischer Segmentierungen wurden im Rahmen deren Entwicklung prototypisch implementiert. Dabei konnte auf verschiedene Softwarebibliotheken zurückgegriffen werden. Die Korrekturwerkzeuge wurden als Module für die Nutzung innerhalb der Entwicklungsumgebung MeVisLab[1] realisiert, wobei die Implementierung vorrangig in Microsoft Visual C++ 2005 erfolgte. Im Folgenden werden erst die verwendete Software, anschließend die entstandenen Klassen und Module näher betrachtet.

6.1 Verwendete Software

MeVisLab. MeVisLab ist eine Entwicklungsumgebung zur Visualisierung und Verarbeitung insbesondere medizinischer Bilddaten. In MeVisLab lassen sich komplexe Algorithmen durch das einfache Vernetzen von Modulen auf einer graphischen Oberfläche entwickeln. Unter Nutzung verschiedener Bibliotheken (wie etwa OpenInventor oder VTK) steht eine große Zahl solcher Module für unterschiedlichste Aufgaben zur Verfügung. Ganze Netzwerke können dabei zu sogenannten Makromodulen zusammengefasst werden, die wiederum als Bestandteil komplexerer Netze verwendet werden können. MeVisLab erlaubt damit die schnelle Entwicklung von Software-Prototypen und den Entwurf von Algorithmen, ohne dass Programmierkenntnisse erforderlich wären. Darüber hinaus existiert jedoch auch eine Schnittstelle zu C++, mit der Programmierern noch flexiblere Möglichkeiten zur Entwicklung eigener Module zur Verfügung stehen.

MSML / MLMSML. Die *Magdeburg Shape Model Library*[2] ist eine C++ Bibliothek für die Segmentierung von Bilddaten mit Hilfe stabiler Feder-Masse-Modelle. Die als Ausgangsmaterial dieser Arbeit verwendeten, zu korrigierenden Segmentierungen liegen im MSML-Datenformat vor. Diese Repräsentation wird auch für die Nachkor-

[1] MeVis Research GmbH, Bremen. http://www.mevislab.de/
[2] Lars Dornheim

rektur beibehalten. Die MLMSML ist eine Sammlung von Modulen zur Anbindung der MSML an MeVisLab. Im Rahmen dieser Arbeit werden vor allem die Visualisierungsmodule verwendet, mit denen die Darstellung der SMSM sowohl in den 2D-Schichtbildern, als auch in der 3D-Ansicht einfach zu realisieren ist.

TAUCS. TAUCS[3] ist eine, auf die Lösung dünn besetzter, linearer Gleichungssysteme spezialisierte C Bibliothek. Die verschiedenen, darin enthaltenen Lösungsverfahen bieten jeweils unterschiedliche Vor- und Nachteile. Für das Traction Tool sowie das Sketch Tool, die durch derart effiziente Implementierungen überhaupt erst realisierbar sind, wird auf die *Multifrontal Supernodal* Cholesky Faktorisierung zurückgegriffen. Dieser Algorithmus hat zwar gegenüber den anderen, in TAUCS enthaltenen Implementierungen einen erhöhten temporären Speicherplatzbedarf, benötigt aber die mit Abstand geringste Rechenzeit [62] und ist daher für die Verwendung bei direkt manipulativen Verfahren besonders prädestiniert.

GSL. Die *GNU Scientific Library*[4] ist eine der umfangreichsten numerischen C/C++ Bibliotheken, deren Funktionsumfang die verschiedensten mathematischen Bereiche abdeckt. Für die Entwicklung der Korrekturwerkzeuge im Rahmen dieser Arbeit wird von der GSL Implementierung der Spline-Interpolation Gebrauch gemacht. Diese zeichnet sich durch hohe Effizienz und besonders einfache Verwendung aus. Die bequeme Möglichkeit der Berechnung von erster und zweiter Ableitung der Spline-Kurven kommt insbesondere der Implementierung des Bending Tools zu gute.

CGAL. Die *Computational Geometry Algorithms Library*[5] ist eine C++ Bibliothek aus effizienten und verlässlichen geometrischen Algorithmen. Im Rahmen dieser Arbeit wird CGAL für verschiedene geometrische Operationen verwendet. Vor allem die effiziente Implementierung der $\sqrt{3}$-Subdivision kommt zum Einsatz.

6.2 Entwickelte Komponenten

Die Interaktionstechniken und verwandten Konzepte wurden als MeVisLab-Module im Projekt *SMSMCorrectionToolbox* implementiert. Abbildung 6.1 zeigt eine schematische Darstellung der dabei entstandenen Klassenstruktur. Jedes der vier Korrekturwerkzeuge wird in einem eigenen Modul gekapselt. Diese sind von der Basisklasse *SMSMCorrectionTool* abgeleitet, von der sie gemeinsame Attribute und Methoden erben. Dazu gehören etwa die Ein- und Ausgabe der SMSM-Datenstruktur oder die

[3] http://www.tau.ac.il/ stoledo/taucs/
[4] http://www.gnu.org/software/gsl/
[5] http://www.cgal.org/

Realisierung der Undo-Operation. *SMSMCorrectionTool* selbst ist wiederum Unterklasse von *SoView2DExtension*. Diese abstrakte MeVisLab-OpenInventor-Klasse ermöglicht die graphische Ausgabe sowie die Ereignisbehandlung von Nutzereingaben auf den Schichtbildern dreidimensionaler Bilddatensätze.

Beim Sketch Tool sowie beim Bending Tool ist die Eingabe von Spline-Kurven erforderlich. Diese werden durch die Klasse *SketchCurve* repräsentiert. Darin enthalten sind alle benötigten Funktionen wie etwa für das Zeichnen der Kurven, zur Bestimmung beliebiger Kurvenpunkte, zur Bestimmung des Parameters $s(\vec{v})$ für beliebige Vertices[6] oder zur Berechnung der Kurvengradienten. *SMSMSketchTool* und *SMSMBendingTool* besitzen Attribute des Typs *SketchCurve*.

Sketch Tool und Traction Tool basieren auf dem LMF. Die entsprechende Deformationsberechnung ist in der Klasse *LaplacianDeformation* gekapselt. Diese benutzt wiederum ein Objekt der Klasse *SparseLinearSystem*, die eine einfach verwendbare Repräsentation dünn besetzter LGS darstellt. Intern wird die Koeffizientenmatrix im CCS-Format repräsentiert. Der Zugriff auf die enthaltenen Elemente von Außen ist jedoch bequem, durch herkömmliche Zeilen-Spalten-Indizierung möglich. Außerdem enthält *SparseLinearSystem* den entwickelten Algorithmus zur Berechnung von $L^T L$ und die Schnittstelle zur Faktorisierung und Lösung des Systems unter Verwendung von TAUCS.

In *LaplacianDeformation* erfolgt die Anwendung von *SparseLinearSystem* für eine gegebene, als ROI definierte Menge von Veritces[6]. Nach Außen stehen dazu Methoden bereit, mit denen eine ROI übergeben, verschiedene Deformationsbedingungen sowie rechte Seiten des LGS definiert und schließlich die Lösungen des Systems auf die ROI-Vertices[6] angewandt werden können.

Das Modul *SoDepthLimiter* realisiert das entworfene Konzept zur Begrenzung des Einflussbereichs der Korrekturwerkzeuge. Da die notwendige Interaktion und Visualisierung im zweidimensionalen Schichtbild stattfindet, erbt auch *SoDepthLimiter* von *SoView2DExtension*. Darüber hinaus hat die Benutzung des DepthLimiters aber auch Einfluss auf die dreidimensionale Modelldarstellung[7], weshalb das Modul einen zusätzlicher OpenInventor-Ausgang besitzt. Die Kommunikation zwischen *SoDepthLimiter* und den Korrekturwerkzeugen erfolgt über OpenInventor-Fields.

Eine Sonderstellung nimmt *SMSMSubdivison* ein. In diesem Modul ist die Anwendung der Subdivision-Algorithmen zur Erhöhung der Modellauflösung der SMSM implementiert. Da hierfür weder Visualisierung noch Interaktion mit den Bilddaten oder der 3D-Darstellung erforderlich sind, ist *SMSMSubdivison* als einfache Erweiterung von *SoSeparator* ausgelegt.

[6] bzw. Massen

[7] durch die farbliche Markierung des Einflussbereiches auf der Modelloberfläche (siehe Abb. 5.4)

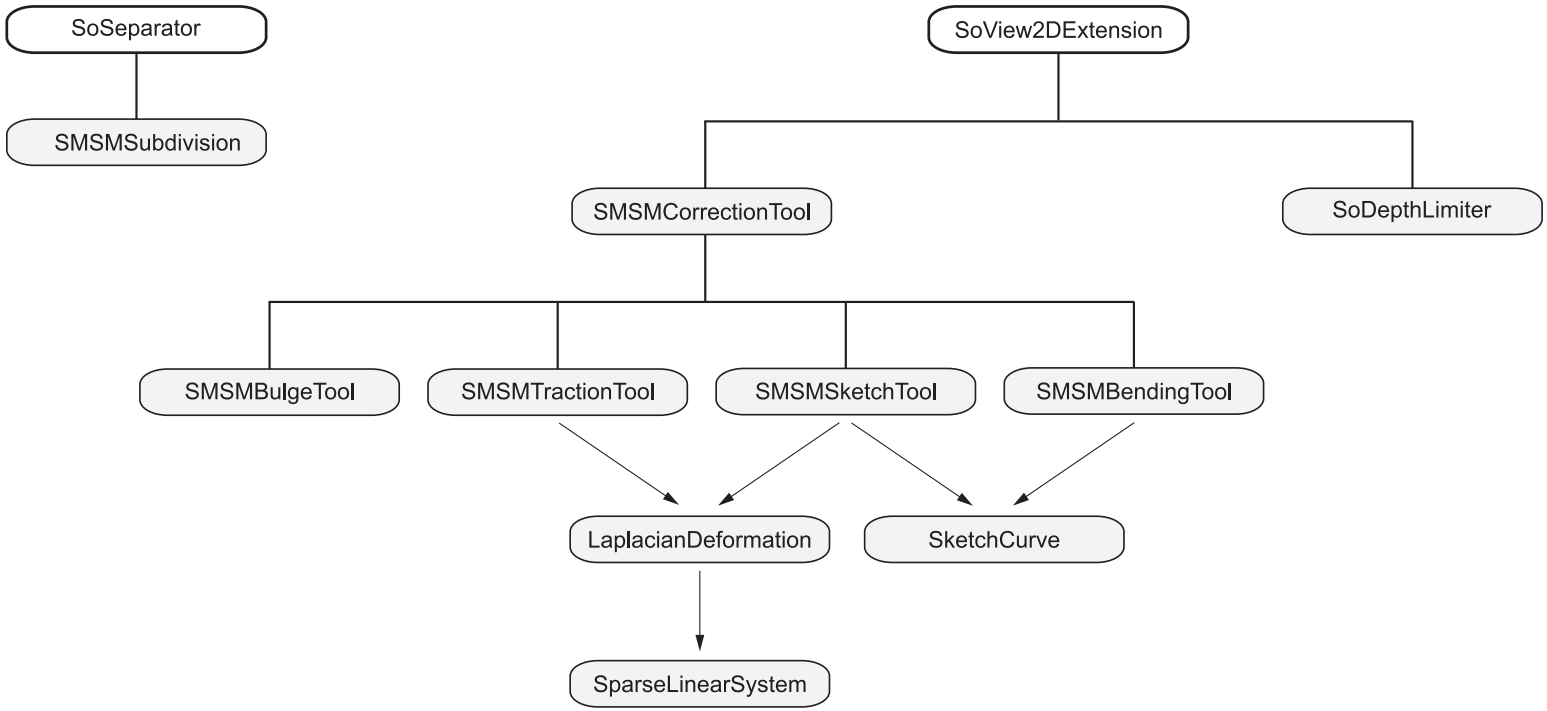


Abbildung 6.1: Klassenstruktur der *SMSMCorrectionToolbox*. Die hellgrau eingefärbten Komponenten repräsentieren die im Rahmen dieser Arbeit entstandenen Klassen. Ein Pfeil von A nach B symbolisiert hier, dass Klasse A Attribute vom Typ B besitzt.

Die Entscheidung zum Aufbau der vorliegenden Klassenstruktur wurde aufgrund folgender Überlegungen getroffen. Durch die Auslagerung der Korrekturwerkzeuge in eigene Module ist es möglich, diese auch völlig eigenständig in beliebige MeVisLab-Netzwerke einzubinden. So können bevorzugte Werkzeuge etwa für weitergehende Untersuchungen ganz einfach in bestehende SMSM-Segmentierungsnetze integriert werden.

Die Kapselung der Hilfskonzepte für die skizzenbasierte Interaktion und die Deformation mittels LMF bringt vor allem zwei Vorteile mit sich. Zum einen kann die Verwendung von stark redundantem Programmcode in den Korrekturwerkzeug-Klassen vermieden werden. Dies erleichtert eventuell erforderliches Debugging und führt zudem zur Verbesserung der Lesbarkeit des Quellcodes, aufgrund dessen schlanker äußerer Form. Außerdem wird die zukünftige Wiederverwendbarkeit der einzelnen Komponenten über die hier bearbeitete Aufgabenstellung hinaus gewährleistet. So ist es ohne größeren Aufwand möglich, durch Definition eigener Deformationsbedingungen weitere LMF-basierte Modellmanipulationen zu implementieren. Auch die Entwicklung andersartiger Interaktionstechniken auf Grundlage der Eingabe von Spline-Kurven lässt sich mit Hilfe der Klasse *SketchCurve* leicht realisieren.

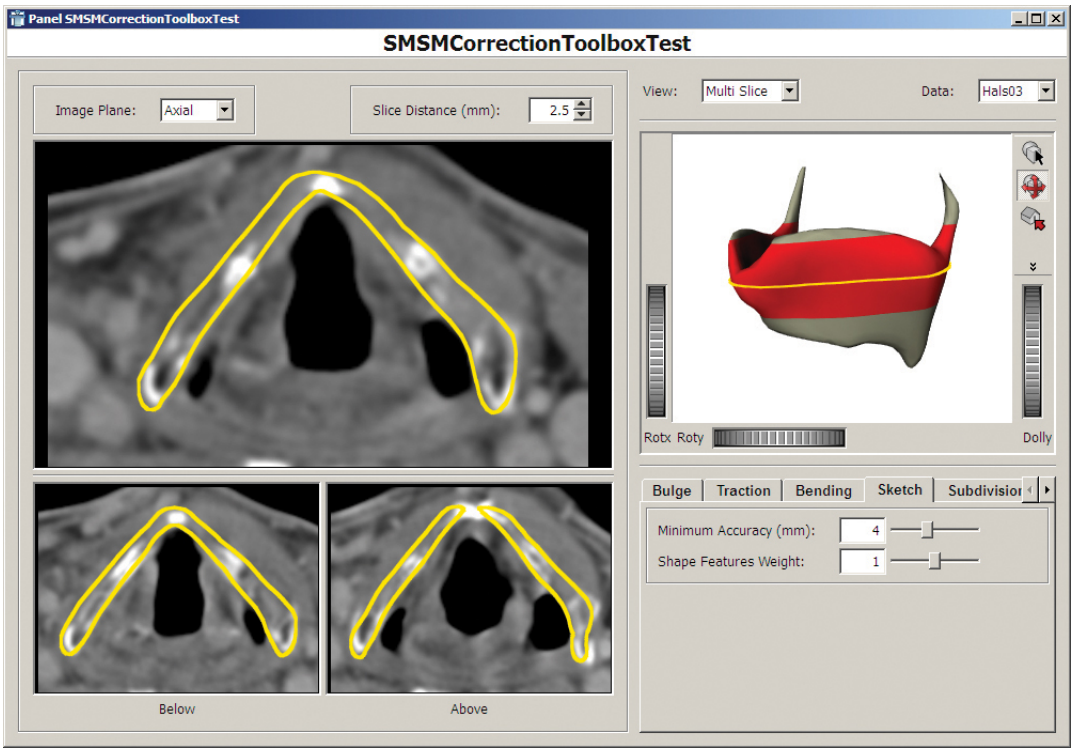


Abbildung 6.2: Graphische Oberfläche der Testumgebung *SMSMCorrectionToolboxTest*.

In MeVisLab wurden die auf C++ Ebene entwickelten Komponenten zusammen mit bereits vorhandenen Modulen zu Netzwerken zusammengeschaltet. Auf diese Weise entstand das Makromodul *SMSMCorrectionToolbox*, das die Korrekturwerkzeuge vereint und über ein Auswahlmenü zugänglich macht. Außerdem wurde mit dem Makromodul *SMSMMutliSliceViewer* die im vorigen Kapitel entworfene Mehrschicht-Visualisierung zur Unterstützung der Nachkorrektur realisiert. Schließlich entstand mit dem Modul *SMSMCorrectionToolboxTest* eine Testumgebung zur Evaluierung der Korrekturwerkzeuge. Darin enthalten ist die 2D-Schichtdarstellung der orthogonalen Schnittführungen, der Multi Slice Viewer, eine 3D-Visualisierung des Segmentierungsmodells und das Auswahlmenü der Korrekturwerkzeuge mit der Zugriffsoption auf deren Parameter. Abbildung 6.2 zeigt die graphische Oberfläche des Moduls *SMSMCorrectionToolboxTest*.

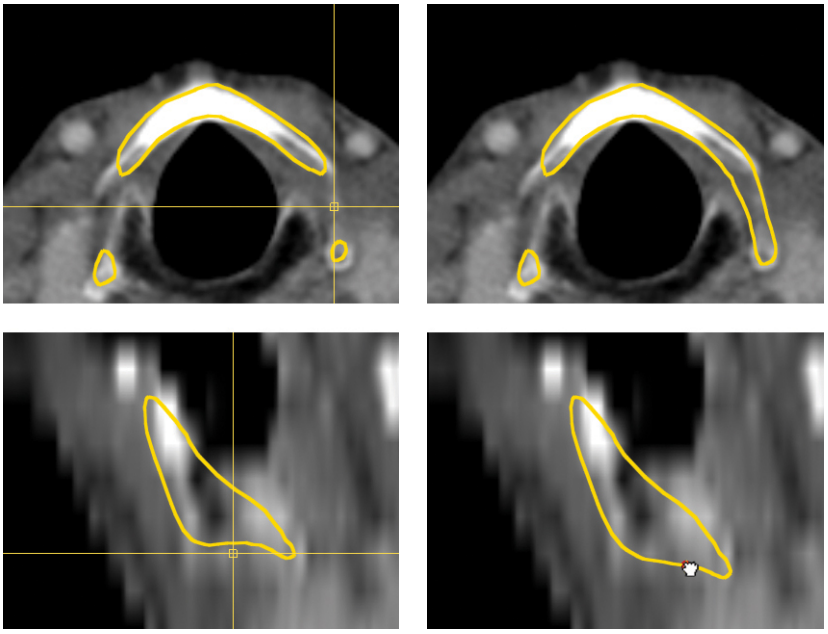
7 Evaluierung

Dieses Kapitel behandelt die Evaluierung der entwickelten Interaktionstechniken. Nach der Implementierung wurden die Werkzeuge zur Nachkorrektur ausgewählter Beispielsegmentierungen testweise angewendet. Die Evaluierung wurde in zwei Phasen unterteilt. In einem ersten Schritt wurde die Eignung der Werkzeuge zur Nachkorrektur medizinischer Segmentierungen getestet. Das Ziel hierbei war herauszufinden, wie flexibel die Werkzeuge eingesetzt werden können und welche Probleme sich bei deren Anwendung ergeben. In der zweiten Phase wurde im kleinen Rahmen die Nutzbarkeit der Korrekturwerkzeuge durch Endanwender untersucht. Dazu wurden fünf Testpersonen gebeten, die Korrekturen an einigen Testbeispielen durchzuführen. Das Hauptinteresse in dieser Phase der Evaluierung lag beim benötigten Zeitaufwand und der subjektiven Einschätzung der Korrekturwerkzeuge durch die Nutzer. Um auch Erkenntnisse darüber zu gewinnen, wie und ob sich ein Unterschied zur trainierten Nutzung ergibt, nahm der Autor dieser Arbeit ebenfalls an den Untersuchungen teil. Sämtliche Tests wurden auf einem modernen Standard PC mit Intel Pentium IV 3 GHz CPU und 2,5 GB RAM durchgeführt.

7.1 Eignung der Interaktionstechniken

Im Mittelpunkt der ersten Phase der Evaluierung stand die Frage, ob sich die entwickelten Interaktionstechniken ausreichend zur Nachkorrektur der Segmentierungen eignen. Dazu wurden die verschiedenen Werkzeuge an einigen Fallbeispielen getestet. Insgesamt lässt sich festhalten, dass die Nachkorrekturen durch Anwendung der *SMSMCorrectionToolbox* möglich waren. Es zeigten sich allerdings größere Unterschiede in der Flexibilität der einzelnen Korrekturwerkzeuge.

Als am vielseitigsten einsetzbar erwies sich das Traction Tool. Besonders hilfreich ist dieses Werkzeug vor allem deshalb, weil es sich auch sinnvoll für Manipulationen in der sagittalen bzw. koronaren Ansicht eignet. Dies ist besonders dann vorteilhaft, wenn Fehlsegmentierungen in z-Richtung vorliegen, wie es etwa häufig beim Schildknorpel aber auch den Lymphknoten der Fall ist. Wie in Abbildung 7.1 dargestellt, lassen sich solche Unter- bzw. Übersegmentierungen durch die Nutzung einer zur



(a) (b)

Abbildung 7.1: Korrektur einer Untersegmentierung in z-Richtung beim Schildknorpel mit dem Traction Tool. Oben: axiale Ansicht, unten: sagittale Ansicht. (a) Zunächst wird mit dem Fadenkreuz im Datensatz zu einer Stelle innerhalb der Untersegmentierung navigiert. (b) Dann wird der untere Teil der Kontur im sagittalen Bild mit dem Traction Tool weiter nach unten gezogen. Dies führt dazu, dass die im axialen Schichtbild dargestellte Kontur an der gewünschten Stelle nun geschlossen erscheint.

axialen Ansicht orthogonalen Ebene leicht beheben. Die Kontur muss dann an den entsprechenden Stellen ganz einfach in z-Richtung verzogen zu werden. Als problematisch beim Traction Tool erwies sich allerdings, dass die richtige Wahl des Einflussradius mitunter schwer nachvollziehbar ist. Insgesamt betrachtet war das Traction Tool aber die einzige der Interaktionstechniken, mit der eine vollständige Nachkorrektur ohne Zuhilfenahme der anderen Werkzeuge möglich war.

Auch das Bulge Tool konnte sich als hilfreiches Werkzeug behaupten. Da bei der Korrektur mit dem Bulge Tool keine Formerhaltung berücksichtigt wird, lassen sich damit besonders solche Fehlsegmentierungen beheben, die aufgrund fehlerhafter Modellannahmen bzw. größerer Abweichung des zu segmentierenden Objektes vom Idealmodell auftreten. Unerwünschtes Verhalten beim Bulge Tool resultierte oftmals aus unvorsichtiger Verwendung. Größere Korrekturen sollten auch mit größeren Kugelraiden durchgeführt werden. Andernfalls ergibt sich das Problem, dass einzelne Dreiecke weit verschoben werden, während sämtliche Nachbarflächen aber unverändert bleiben. Außerdem kann es so zu Überschneidungen der Dreiecksflächen kommen, die ohne Nutzung der Undo-Funktion nur noch schwer wieder rückgängig

zu machen sind. Mit Bedacht angewandt, konnte das Bulge Tool allerdings gut für die Nachkorrekturen eingesetzt werden. Besonders für die Modifikation von Tumorsegmentierungen, denen kein Modellwissen zugrunde gelegt werden kann, erwies sich das Bulge Tool als geeignet.

Das Sketch Tool zeigte sich als bedingt verwendbar für die Segmentierungskorrektur. Die besten Ergebnisse wurden auf höher aufgelösten Modellen erzielt, wie sie zum Beispiel aus Anwendung der Subdivision hervorgehen. Problematisch war hierbei vor allem, dass je nach Modellauflösung und Größe der sich im Schichtbild dargestellten Kontur, die Berechnung spürbare Zeit in Anspruch nahm. Dies resultiert zum einen aus der Verwendung größerer LGS wegen der erhöhten Anzahl an Deformationsbedingungen, zum anderen aber auch aus der Suche nach dem zu deformierenden Teil der Objektkontur (Referenzkurve). Da das Sketch Tool jenen Konturteil als Referenzkurve identifiziert, der die größere Ähnlichkeit mit der Zielkurve aufweist, müssen zur Berechnung des Ähnlichkeitswertes sämtliche, im zweidimensionalen Schichtbild benachbarten Konturflächen besucht werden. Dieser Vorgang benötigt in der aktuellen Implementierung spürbare Zeit, kann vermutlich jedoch noch effizienter gelöst werden. Darüber hinaus führt der formerhaltende Charakter des Sketch Tools mitunter zu unerwarteten Auswirkungen, besonders dann, wenn größere Konturteile in einem einzigen Interaktionsschritt deformiert werden sollen.

Das Bending Tool erwies sich als das am wenigsten flexible Werkzeug der *SMSMCorrectionToolbox*. Allerdings ließ es sich in manchen Fällen gut für Korrekturen im Bereich der Seitenplatten des Schildknorpels anwenden. Auch für die Korrektur von Lymphknotensegmentierungen konnte es eingesetzt werden. Interessant ist zudem die Beobachtung, dass die Verwendung des Bending Tools auch ohne Annäherung der Hauptskelettlinie sinnvoll sein kann. Nachteilig ist wohl am meisten, dass die Interaktionsauswirkungen nicht immer vollständig nachvollziehbar sind. Um eine zielgerichtete Korrektur vorzunehmen waren oftmals mehrere Durchgänge von Kurvenzeichnung und Undo-Funktion notwendig. Insgesamt kann das Bending Tool als ein interessantes Korrekturwerkzeug betrachtet werden. Zwar ließ sich mit dieser Interaktionstechnik keine komplette Segmentierungskorrektur vornehmen, in manchen Fällen erwies es sich jedoch als hilfreich.

Insgesamt hat sich für die Nachkorrekturen folgende Vorgehensweise bewährt. Zu Beginn sollten eventuelle Unter- bzw. Übersegmentierungen in z-Richtung behoben werden. Dies geschieht am effizientesten durch Wahl eines Ansicht-Layouts in dem die drei orthogonalen Schnittebenen zu sehen sind. Hier können die betreffenden Korrekturen mit Hilfe des Traction Tools einfach vorgenommen werden. Sind diese Fehler grob behoben, kann zur Multi Slice Ansicht gewechselt und in den axialen Schichtbildern weitergearbeitet werden. Dort kann die Nachkorrektur dann unter Berücksichtigung des Einflusses auf Nachbarschichten mit dem bevorzugten Werkzeug fortgeführt werden. Empfehlenswert ist es zudem, dabei eine leichte

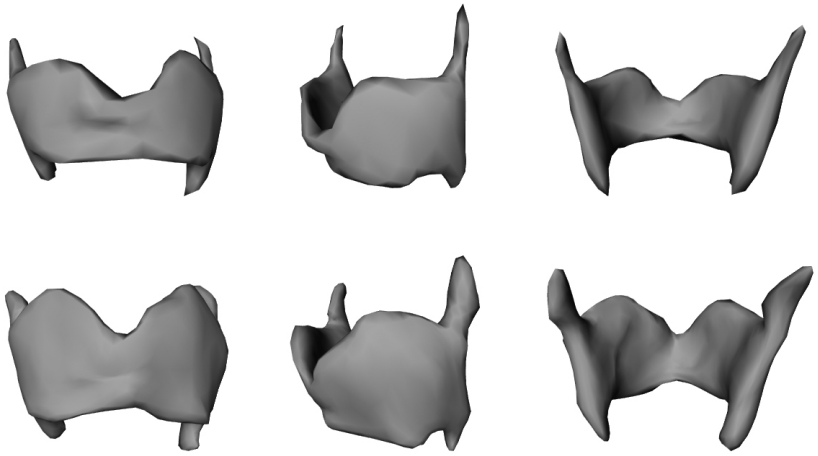


Abbildung 7.2: Korrektur der Segmentierung eines Schildknorpels. Verschiedene Ansichten. Obere Reihe: Fehlerhaftes Segmentierungsmodell, untere Reihe: mit Hilfe der entwickelten Interaktionstechniken korrigiertes Modell. Deutlich zu erkennen sind vor allem die Unterschiede in der Höhe der Seitenplatten sowie der Gestalt der Hörner. Außerdem ist das korrigierte Modell insgesamt breiter als das automatisch erzeugte Segmentierungsergebnis. Die Korrektur wurde mit den Traction Tool, dem Bulge Tool sowie einer Iteration $\sqrt{3}$-Subdivision vorgenommen.

Übersegmentierung zuzulassen. Hat das bearbeitete Modell eine vergleichsweise geringe Auflösung, lassen sich gekrümmte Konturen ohnehin nur begrenzt modellieren. Ist die Approximation nicht ausreichend, bietet sich die Anwendung einer Subdivision Iteration auf das Modell an. Danach lassen sich wesentlich feinere Korrekturen vornehmen. Außerdem wird durch den glättenden Effekt an vielen Stellen der Segmentierung vermutlich bereits das gewünschte Ergebnis erzielt werden. Zusätzlich verliert das Modell bei der Subdivision aber auch an Volumen. Dies ist aufgrund deren rundlicher Form, ganz besonders bei den Lymphknoten zu beobachten. Da die hier verwendeten Modelle relativ gering aufgelöst sind, ist eine Subdivision besonders zu empfehlen, um ein natürlich wirkendes Resultat zu erreichen. Der Volumenverlust lässt sich hierbei, wie bereits erwähnt, durch eine leichte Übersegmentierung im Vorfeld kompensieren.

Generell kann zudem angemerkt werden, dass etwaige unerwünschte Interaktionsauswirkungen während des Korrekturvorgangs am besten unmittelbar per Undo-Funktion wieder rückgängig gemacht werden sollten. Andernfalls, so zeigt die Erfahrung, werden diese Situationen in den meisten Fällen eher verschlechtert und können nur unter unverhältnismäßig großem Zeitaufwand wieder behoben werden. Außerdem hat ein solches *trial and error* Vorgehen den Vorteil, bessere Einsicht in das Verhalten der Korrekturwerkzeuge zu gewinnen, wodurch unerwünschte Situationen künftig vermieden werden können. Abbildung 7.2 zeigt schließlich das Ergebnis einer erfolgreich durchgeführten Nachkorrektur.

7.2 Nutzertests

Neben den generellen Untersuchungen zur Eignung der Interaktionstechniken für die Nachkorrektur, wurde auch im kleinen Rahmen eine Evaluierung mit Testnutzern durchgeführt. Hierfür wurden fünf Teilnehmer gebeten, ausgewählte Segmentierungsbeispiele zu bearbeiten. Bei den Testnutzern handelte es sich allerdings nicht um medizinisches Personal. Vielmehr gaben alle Teilnehmer an, wenig bis keine Erfahrung mit medizinischen Segmentierungen zu haben. Um dieses Defizit möglichst zu kompensieren, wurde den Teilnehmern das jeweils durch die Korrekturen zu erreichende Ziel, in Form einer vom Experten erzeugten Goldstandard-Segmentierung vorgegeben. Zwar sind die hier präsentierten Testergebnisse aufgrund der geringen Teilnehmerzahl und der fehlenden medizinischen Kenntnisse nur begrenzt aussagekräftig, vermitteln aber dennoch einen ersten Eindruck von der Nutzbarkeit der Korrekturwerkzeuge im derzeitigen Entwicklungsstadium.

Für die Nutzertests hatten die Teilnehmer die Aufgabe zwei übersegmentierte Lymphknoten und eine durchschnittlich fehlerhafte Schildknorpelsegmentierung zu korrigieren. Zu Beginn der Tests stand eine Einführung in die entwickelten Korrekturwerkzeuge sowie die zu korrigierenden Testfälle. Anschließend wurde den Teilnehmern die Möglichkeit gegeben, die Werkzeuge selbständig auszuprobieren. Nachdem die Nutzer angaben, sie hätten das Gefühl, die Interaktionstechniken ausreichend verstanden zu haben, sollten die Beispielsegmentierungen korrigiert werden. Dabei stand es den Testpersonen frei, zwischen den angebotenen Werkzeugen beliebig zu wählen. Während der Korrekturvorgänge wurde die insgesamt benötigte Zeit gemessen. Außerdem wurde registriert, wie lange, welche der Korrekturwerkzeuge zur Lösung der Aufgaben verwendet wurden. Um den Vergleich zur bisherigen, manuellen Methode anstellen zu können, sollten die gleichen Testfälle auch auf die herkömmliche Weise korrigiert werden. Im Anschluss wurden die Testpersonen noch um ihre subjektive Einschätzung zur Intuitivität und Eignung der Werkzeuge gebeten. Dies geschah durch Verwendung einer Bewertungsskala von 1.0 (sehr gut) bis 5.0 (schlecht). Tabelle 7.1 zeigt das durchschnittliche Ergebnis dieser Befragung.

	Bulge Tool	Traction Tool	Sketch Tool	Bending Tool
Intuitivität	1,5	1,8	1,5	3,6
Eignung	1,6	1,4	3,1	3,7

Tabelle 7.1: Durchschnittliche Einschätzung der Korrekturwerkzeuge durch die fünf Testpersonen nach den Kriterien Intuitivität und Eignung.

Abgesehen vom Bending Tool wurden den Korrekturwerkzeugen gute Bewertungen für die Intuitivität gegeben. Bei der Einschätzung der Eignung zur Segmentierungskorrektur wurden das Traction Tool und das Bulge Tool präferiert. Entsprechend dieser Bewertungen wurde auch während der Testaufgaben zwischen den Werkzeugen gewählt. In erster Linie wurden hierbei Bulge Tool und Traction Tool verwendet, nur ein Nutzer machte mehrfach vom Sketch Tool Gebrauch. Das Bending Tool wurde nur sehr selten in Betracht gezogen, oftmals auch nur für kurze Zeit.

Segmentierung	Nutzer	*SMSMCorrectionToolbox*	Manuelle Methode
Lymphknoten	trainiert	00:39 min	00:43 min
	untrainiert	03:12 min	01:25 min
Schildknorpel	trainiert	04:47 min	04:21 min
	untrainiert	14:49 min	06:14 min

Tabelle 7.2: Durchschnittlich benötigte Korrekturzeiten bei den Testaufgaben.

Den zur Lösung der Testaufgaben insgesamt benötigten Zeitaufwand zeigt Tabelle 7.2. Hier wird bei den Angaben zwischen trainierten und untrainierten Nutzern unterschieden. Letztere sind die Testteilnehmer, während der Autor dieser Arbeit als trainierter Nutzer bezeichnet wird. Größere Probleme als erwartet, bereitete den Testpersonen die Koordination der verschiedenen Schnittführungsansichten bzw. ganz allgemein die Idee, ein eigentlich dreidimensionales Objekt zu manipulieren. Da dies allerdings ein zentraler Gedanke der entwickelten Interaktionstechniken ist, ergaben sich Schwierigkeiten bei der Durchführung der Korrekturaufgaben. Während die Testpersonen in erster Linie darauf bedacht waren, die Kontur im axialen Schichtbild anzupassen, wurde häufig der Bezug zum dreidimensionalen Charakter des Segmentierungsmodells außer Acht gelassen. Dies schlägt sich vor allem in den benötigten Zeiten nieder. Insbesondere bei der Korrektur des komplex geformten Schildknorpels, fiel es den Teilnehmern schwer sich zu orientieren. Wie aus der Tabelle aber auch ersichtlich wird, sind die Nachkorrekturen bei trainierter Nutzung und Erfahrung im Umgang mit CT-Daten zumindest innerhalb vergleichbarer Zeit wie mit der manuellen Methode zu erreichen. Interessant ist die Frage, ob sich medizinisch erfahrene Anwender wesentlich besser in den Datensätzen orientieren können oder ob sie anfangs ähnlich Probleme mit der Nachkorrektur wie die Testpersonen haben werden.

8 Zusammenfassung und Ausblick

8.1 Zusammenfassung

Die durch automatische Verfahren gewonnenen Ergebnisse medizinischer Segmentierungen sind oftmals gut, in vielen Fällen jedoch zumindest stellenweise korrekturbedürftig. Die nachträglichen Korrekturen werden bisher durch das schichtweise, manuelle Überzeichnen der fehlerhaften Segmentierungen durchgeführt. Dies ist zeitaufwändig und führt außerdem zu stark subjektiven und ungenauen Resultaten. An alternativen Methoden für die Nachkorrektur mangelt es allerdings.

Das Ziel dieser Diplomarbeit war es daher Ansätze für möglichst effiziente und intuitive Alternativen zu entwickeln und zu implementieren. Speziell wurde die Nachkorrektur von auf stabilen Feder-Masse-Modellen basierenden Segmentierungen des Schildknorpels und der Halslymphknoten in CT-Aufnahmen betrachtet. Dabei sollte versucht werden bereits in Form der Modelle vorhandenes Formwissen auszunutzen. Außerdem sollten die neu entwickelten Verfahren möglichst auch für die Modifizierung manuell erstellter Segmentierungen von Halstumoren anwendbar sein.

Als Lösungsansatz wurde im Rahmen dieser Arbeit vorgeschlagen, anstatt wie bisher aus den Segmentierungsmodellen schichtweise Konturen zu berechnen, die 3D-Modelle beizubehalten und die Korrektur als deren direkte Manipulation zu realisieren. Daraus ergeben sich eine Reihe von Vorteilen. Bei der Segmentierung (sowie der Nachkorrektur) kann sich nicht nur starr auf die im CT-Datensatz enthaltene Information gestützt werden, weil diese immer gestört ist und nur in einem bestimmten Schichtabstand gemessen wurde. Vielmehr greift der ausführende Arzt während des Korrekturprozesses auch stets auf sein medizinisches Fachwissen zurück. Da die zu segmentierenden Objekte in den Datensätzen tatsächlich dreidimensional[1] sind, wird vor allem auch das Ziel einer anatomisch plausiblen Segmentierung verfolgt. Durch die hier vorgeschlagene Art der Nachkorrektur ist zu jedem Zeitpunkt eine 3D-Visualisierung des Segmentierungsmodells verfügbar, wodurch der Korrektur-

[1] und nicht Aneinanderreihung zweidimensionaler Schichten mit mehreren mm Abstand

vorgang sinnvoll unterstützt werden kann. Da in vielen Fällen aus den Segmentierungsergebnissen ohnehin dreidimensionale Visualisierungen generiert werden sollen, kann bei dieser Art der Nachkorrektur auf die dazu notwendigen Schritte verzichtet werden. Vor allen Dingen lässt sich durch die Beibehaltung der Modelle bei richtiger Anwendung auch die Korrektur selbst erleichtern. Die Manipulationen am Modell haben in der Regel Auswirkungen auf die dargestellten Konturen in mehreren benachbarten Schichtbildern des Datensatzes. So kann vermieden werden, dass ähnliche Korrekturen in verschiedenen Schichten mehrfach erforderlich werden.

Die Interaktion für die Nachkorrektur findet in den zweidimensionalen Schichtbildern der Datensätze statt. Um den Einfluss der Deformationen auch auf benachbarte Schichten direkt beurteilen zu können, wurde die Multi Slice Ansicht entwickelt, mit der drei parallele Schichten gleichzeitig betrachtet werden können. Zusätzlich wird eine 3D-Visualisierung angezeigt, die ständig die momentane Situation des dreidimensionalen Modells erkenntlich macht.

Konkret wurden für diese Diplomarbeit erstmals Interaktionstechniken aus dem Bereich der intuitiven Polygonmodellierung für die Korrektur medizinischer Segmentierungen angewandt. Im Einzelnen wurden dabei folgende Werkzeuge entwickelt. Mit dem Bulge Tool kann ein kugelförmiges Objekt durch den Datensatz bewegt und die berührte Modellkontur gewissermaßen ausgebeult werden. Das Traction Tool erlaubt dem Nutzer an der Modellkontur zu ziehen, während das Sketch Tool das einfache Überzeichnen fehlerhafter Konturbereiche mit einer Spline-Kurve ermöglicht. Die beiden letztgenannten Korrekturwerkzeuge basieren auf der Anwendung des Laplacian Modeling Framework, mit dem auf einfache Weise Manipulationen von Polygonmodellen beschrieben werden können. Die Deformierung resultiert dabei aus der näherungsweisen Lösung eines linearen Gleichungssystems unter Aufrechterhaltung lokaler Formeigenschaften. Das LMF findet deshalb hier Anwendung, weil auf diese Weise das Formwissen der SMSM ausgenutzt wird. Dies ist sinnvoll, weil in vielen Fällen davon ausgegangen werden kann, dass die der Segmentierung zugrunde liegenden Modellannahmen zutreffend sind und lediglich eine falsche Anpassung an den Datensatz erfolgte. Als viertes Werkzeug wurde das Bending Tool entwickelt. Hier wird durch Angabe zweier Kurven eine Abbildung definiert, aufgrund derer das Modell deformiert wird.

Schließlich wurden die entwickelten Interaktionstechniken auf ihre Eignung zur Nachkorrektur sowie im Rahmen einer klein angelegten Nutzerstudie, auf ihre intuitive Nutzbarkeit untersucht. Dabei zeigte sich die grundsätzliche Eignung der Korrekturwerkzeuge und insgesamt eine Überlegenheit des Traction Tools. Auch von den Testnutzern wurde das Traction Tool bevorzugt - ebenso das Bulge Tool. Im Gegensatz zum trainierten Nutzer waren die Testteilnehmer jedoch nicht in der Lage, mit den neu entwickelten Werkzeugen vergleichbar schnelle Korrekturen wie mit der bisherigen manuellen Methode durchzuführen. Die Ergebnisse der trainierten

Anwendung stimmen jedoch zuversichtlich, dass durch Verfeinerung der hier entwickelten Ansätze die gewünschte intuitive Nutzbarkeit erreicht werden kann.

8.2 Ausblick

Um die entwickelten Interaktionstechniken in Zukunft für den Einsatz zur Nachkorrektur dreidimensionaler Segmentierungen bei den Medizinern etablieren zu können, besteht ein wichtiger, nächster Schritt in der Durchführung einer ausgedehnten Nutzerstudie. Dabei sollte mehr Datenmaterial zur Verfügung stehen, um die Korrekturwerkzeuge auch an selteneren Fehlsegmentierungen zu testen und um Informationen zu sammeln, mit deren Hilfe die Werkzeuge für besonders häufig auftretende Fälle optimiert werden können. Interessant wäre auch den Einsatz an Segmentierungen von anderen, als den hier beispielhaft betrachteten anatomischen Strukturen zu untersuchen. Gerade für den HNO-Bereich entstehen in letzter Zeit zunehmend SMSM Segmentierungsmethoden für unterschiedliche Organe.

Besonders wichtig ist sicherlich, dass diese Studie mit der tatsächlich angesprochenen Zielgruppe, also medizinischem Fachpersonal und wesentlich mehr Teilnehmern durchgeführt wird. Die entsprechenden Ärzte in den chirurgischen Abteilungen sind jedoch naturgemäß viel und oftmals auch unplanmäßig mit besonders wichtigen Aufgaben fernab der Arbeit am Computer beschäftigt. Somit gestalten sich überwachte Evaluierungen, in denen die Versuchsperson nach einem festgelegten Ablauf verschiedene Testfälle bearbeitet, in der Regel schwierig. Besser wäre es daher vermutlich, die Korrekturwerkzeuge über einen längeren Zeitraum in den regulären Arbeitsablauf der Mediziner zu integrieren. Nach einer einmaligen Einführung in die Bedienweise könnten die Interaktionstechniken dann von den Teilnehmern selbstständig evaluiert werden. Es ist zu erwarten, dass sich nach einiger Zeit herausstellen wird, ob und welche der Korrekturwerkzeuge von den Medizinern akzeptiert werden und eine wesentliche Verbesserung der bisherigen Situation ermöglichen.

Bevor dieser nächste Evaluierungsschritt jedoch angegangen wird, sollten zuvor Verbesserungen an den Korrekturwerkzeugen vorgenommen werden. Außerdem sollte der Fokus vermutlich auf die Weiterentwicklung von Bulge Tool und Traction Tool gelegt werden. Die anderen beiden Werkzeuge stellten sich, bei den im Rahmen dieser Arbeit durchgeführten Untersuchungen, als die am wenigsten flexiblen heraus. Zudem scheint es wegen der notwendigen Gewöhnung an die neuen Interaktionstechniken wahrscheinlicher, dass sich die zukünftigen Nutzer auf die Verwendung von ein bis zwei Werkzeugen konzentrieren werden. Eine größere Auswahl an differenzierteren Korrekturwerkzeugen anzubieten erscheint zwar sinnvoll, jedoch ist

zu vermuten, dass die Ärzte nicht in der Lage sind, den damit einhergehenden Zeitaufwand für die Einarbeitung aufzubringen.

Wichtig für die Verbesserung des Bulge Tools ist sicherlich die Möglichkeit einer adaptiven Verfeinerung des Polygonmodells. Werden momentan größere Deformationen mit dem Bulge Tool vorgenommen, resultiert dies zwangsläufig in einer starken Vergrößerung der betroffenen Dreiecksflächen. Stattdessen wäre es jedoch besser, wenn dies durch einen automatischen Verfeinerungsmechanismus unterbunden werden würde. Steigt der Flächeninhalt eines Dreiecks über eine bestimmte Größe, sollte das Dreieck unterteilt werden. Genauso sollte es aber auch zum Zusammenfassen mehrerer Dreiecke kommen, sobald der Flächeninhalt eine minimale Größe unterschreitet.

Über die Anwendung der entwickelten Interaktionstechniken zur Nachkorrektur hinaus, wären aber auch andere Einsatzbereiche denkbar. So könnte zum Beispiel untersucht werden, ob bei den bisher manuell durchgeführten Tumorsegmentierungen, die Werkzeuge nicht nur für Modifikationen, sondern auch für die eigentliche Segmentierung anwendbar sind. Denkbar wäre hier, dass der Tumor ausgehend von einem geometrischen Primitiv komplett modelliert wird. Interessant ist sicher auch, die Interaktionstechniken bei der Erstellung stabiler Feder-Masse-Modelle einzusetzen. Für deren Generierung ist oftmals ein möglichst repräsentativer Datensatz erforderlich. Falls dies nicht gegeben ist, könnten die Korrekturwerkzeuge auch hier zur unterstützenden Modellierung verwendet werden.

Literaturverzeichnis

[1] Adams, R. und L. Bischof: *Seeded Region Growing*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(6):641–647, 1994.

[2] Al-Zubi, S.: *Survey of Deformable Models*. Technischer Bericht, Otto-von-Guericke-Universität Magdeburg, 2002.

[3] Alexa, M.: *Differential coordinates for local mesh morphing and deformation*. The Visual Computer, V19(2):105–114, 2003.

[4] Alexa, M.: *Mesh editing based on discrete Laplace and Poisson models*. In: *SIGGRAPH '06: ACM SIGGRAPH 2006 Courses*, Seiten 51–59. ACM, 2006.

[5] Aliroteh, M. und T. McInerney: *SketchSurfaces: Sketch-Line Initialized Deformable Surfaces for Efficient and Controllable Interactive 3D Medical Image Segmentation.* In: Bebis, G., R. D. Boyle, B. Parvin, D. Koracin, N. Paragios, T. F. Syeda-Mahmood, T. Ju, Z. Liu, S. Coquillart, C. Cruz-Neira, T. Müller und T. Malzbender (Herausgeber): *ISVC (1)*, Band 4841 der Reihe *Lecture Notes in Computer Science*, Seiten 542–553. Springer, 2007.

[6] Barr, A.H.: *Global and local deformations of solid primitives*. SIGGRAPH Comput. Graph., 18(3):21–30, 1984.

[7] Barrett, W.A. und E.N. Mortensen: *Fast, Accurate, and Reproducible Live-Wire Boundary Extraction*. In: *VBC '96: Proceedings of the 4th International Conference on Visualization in Biomedical Computing*, Seiten 183–192, London, UK, 1996. Springer-Verlag.

[8] Boles, D.: *Multimediasysteme*. Carl von Ossietzky Universität Oldenburg, Begleitbuch zur Vorlesung, 1998.

[9] Buzug, T.M.: *Computertomographie*. In: Kramme, R. (Herausgeber): *Medizintechnik, 3., vollständig überarbeitete und erweiterte Auflage*. Spinger, Heidelberg, Berlin, 2007.

[10] Carpendale, M.S.T. und C. Montagnese: *A framework for unifying presentation space*. In: *UIST '01: Proceedings of the 14th annual ACM symposium on*

User interface software and technology, Seiten 61–70, New York, NY, USA, 2001. ACM.

[11] Catmull, E. und J. Clark: *Recursively generated B-spline surfaces on arbitrary topological meshes*. Computer-Aided Design, 10(6):350 – 355, 1978.

[12] Cohen, L.D.: *On active contour models and balloons*. CVGIP: Image Underst., 53(2):211–218, March 1991.

[13] Dijkstra, E.W.: *A note on two problems in connexion with graphs*. Numerische Mathematik, 1:269–271, 1959.

[14] Dongarra, J.: *Sparse Matrix Storage Formats (Section 10.1)*. In: Bai, Z., A. Ruhe J. Demmel, J. Dongarra und H. van der Vorst (Herausgeber): *Templates for the solution of algebraic eigenvalue problems: a practical guide*. SIAM, Philadelphia, USA, 2000.

[15] Dornheim, J., L. Dornheim, B. Preim, I. Hertel und G. Strauß: *Generation and Initialization of Stable 3D Mass-Spring Models for the Segmentation of the Thyroid Cartilage*. In: *DAGM-Symposium*, Seiten 162–171, 2006.

[16] Dornheim, L.: *Generierung und Dynamik physikalisch basierter 3D-Modelle zur Segmentierung des linken Ventrikels in SPECT-Daten*. Diplomarbeit, Otto-von-Guericke Universität Magdeburg, 2005.

[17] Dugas, S.: *Connected Threshold Segmentation of Ventricles in the Brain*. The Insight Journal [http://insight-journal.org:8180/dspace-oai/request] (United States), 2007.

[18] Eicke, D.: *Detektion von Lymphknoten in Hals CT-Datensätzen*. Diplomarbeit, Otto-von-Guericke Universität Magdeburg, 2005.

[19] Falcao, A.X., J.K. Udupa, S. Samarasekera, S. Sharma, B. E. Hirsch und R. de A. Lotufo: *User-Steered Image Segmentation Paradigms: Live Wire and Live Lane*. Graphical Models and Image Processing, 60(4):233–260, July 1998.

[20] Foley, J.D., A. van Dam, S.K. Feiner und J.F. Hughes (Herausgeber): *Computer Graphics: Principles and Practice, Second Edition*. Addison-Wesley, 1992.

[21] Gademann, G.: *Humanpathologie II - Onkologie*. Otto-von-Guericke Universität Magdeburg, Skript zur Vorlesung, 2006.

[22] Galasnki, M. und M. Prokop: *Ganzkörper-Computertomographie*. Georg Thieme Verlag, 1998.

[23] Gerhardt, H.-J.: *Vademekum der HNO-Heilkunde*. VEB Georg Thieme, Leipzig, 1978.

[24] Gilles, B., L. Moccozet und N. Magnenat-Thalmann: *Anatomical Modelling of*

the Musculoskeletal System from MRI. In: *MICCAI*, Seiten 289–296, 2006.

[25] Gouraud, H.: *Continuous Shading of Curved Surfaces*. IEEE Transactions on Computers, 20(6):623–628, 1971.

[26] Gramatzki, M.: *Der Fokus im HNO-Bereich aus internistisch-onkologischer Sicht*. In: Biesinger, E. und H. Iro (Herausgeber): *Der Fokus im HNO-Bereich: Fakt oder Fiktion?* Spinger, Heidelberg, Berlin, 2006.

[27] Gray, H.: *Anatomy of the Human Body*. Lea & Febiger, Philadelphia, 1918. Bartleby.com, 2000. www.bartleby.com/107/. [10.02.2009].

[28] Igarashi, T. und J.F. Hughes: *Smooth meshes for sketch-based freeform modeling*. In: *SIGGRAPH '07: ACM SIGGRAPH 2007 courses*, Seite 22, New York, NY, USA, 2007. ACM.

[29] Igarashi, T., S. Matsuoka und H. Tanaka: *Teddy: a sketching interface for 3D freeform design*. In: *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, Seiten 409–416, 1999.

[30] Kalender, W.A.: *Computertomographie: Grundlagen, Gerätetechnologie, Bildqualität, Anwendungen*. Publicis Corporate Publishing, Erlangen, 2006.

[31] Kass, M., A. Witkin und D. Terzopoulos: *Snakes: Active contour models*. International Journal of Computer Vision, V1(4):321–331, 1988.

[32] Kaus, M., S.K. Warfield, F.A. Jolesz und R. Kikinis: *Adaptive template moderated brain tumor segmentation in MRI*. In: *Bildverarbeitung in der Medizin*, Seiten 102–105. Springer, 1999.

[33] Kho, Y. und M. Garl: *Sketching mesh deformations*. In: *I3D '05: Proceedings of the 2005 symposium on Interactive 3D graphics and games*, Seiten 147–154. ACM, 2005.

[34] Kobbelt, L.: *sqrt(3)-Subdivision*. In: *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, Seiten 103–112, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[35] Konrad-Verse, O.: *Interactive Refinement of a Balloon Segmentation on Medical Data*. Diplomarbeit, Universität Bremen, 2004.

[36] Kunert, T., M. Heiland und H.P. Meinzer: *Interaktive Segmentierung von zweidimensionalen Datensätzen mit Hilfe von Aktiven Konturen*. In: *Bildverarbeitung für die Medizin*, Seiten 257–261, 2001.

[37] Lawrence, J. und T. Funkhouser: *A Painting Interface for Interactive Surface Deformations*. Graphical Models (Special Issue), 66(6):418–438, jan 2004.

[38] Lehmann, D.: *Segmentierung von Blutgefäßen des Halses in CT-Daten mittels Stabiler Feder-Masse-Modelle*. Diplomarbeit, Otto-von-Guericke Universität Magdeburg, 2008.

[39] Lipman, Y., O. Sorkine, M. Alexa, D. Cohen-Or, D. Levin, C. Rössl und H.P. Seidel: *Laplacian Framework for Interactive Mesh Editing*. International Journal of Shape Modeling, 11(1):43–61, 2005.

[40] Majeed, T.: *Segmentation of the Sternocleidomastoid Muscle in CT Datasets of the Neck using Stable 3D Mass-Spring Models*. Diplomarbeit, Otto-von-Guericke Universität Magdeburg, 2009.

[41] Masuda, H. und K. Ogawa: *Interactive Deformation of 3D Mesh Models*. Computer Aided Design and Applications, 5(1-4):47–57, 2008.

[42] Masuda, H., Y. Yoshioka und Y. Furukawa: *Preserving form features in interactive mesh deformation*. Computer Aided Design, 39(5):361–368, 2007.

[43] Nealen, A., T. Igarashi, O. Sorkine und M. Alexa: *FiberMesh: designing freeform surfaces with 3D curves*. In: *SIGGRAPH '07: ACM SIGGRAPH 2007 papers*, Seite 41, New York, NY, USA, 2007. ACM.

[44] Nealen, A., O. Sorkine, M. Alexa und D. Cohen-Or: *A sketch-based interface for detail-preserving mesh editing*. ACM Transactions on Graphics, 24(3):1142–1147, 2005.

[45] Nealen, A.V.: *Interfaces and Algorithms for the Creation, Modification, and Optimization of Surface Meshes*. Doktorarbeit, Fakultät IV, Computer Graphics, TU Berlin, 2007.

[46] Neumann, P., T. Isenberg, S. Carpendale und T. Strothotte: *Expressive Distortion of Strokes and 3D Meshes*. Technischer Bericht, Department of Computer Science, University of Calgary, Canada, 2005.

[47] O'Donnell, L., C.-F. Westin, W.E.L. Grimson, J. Ruiz-Alzola, M.E. Shenton und R. Kikinis: *Phase-Based User-Steered Image Segmentation*. In: Niessen, W. (Herausgeber): *Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'01)*, Seiten 1022–1030, Utrecht, Netherlands, October 14–17 2001. Springer Verlag.

[48] Olabarriaga, S.D.: *Human-Computer Interaction for the Segmentation of Medical Images*. Doktorarbeit, Universiteit van Amsterdam, 1999.

[49] Olsen, L., F.F. Samavati, M.C. Sousa und J. Jorge: *Sketch-Based Modeling: A Survey*. Computers & Graphics, 33:85 – 103, 2009.

[50] Plass, M. und M. Stone: *Curve-fitting with Piecewise Parametric Cubics*. SIGGRAPH Comput. Graph., 17(3):229–239, 1983.

[51] Pohle, R.: *Computergestützte Bildanalyse zur Auswertung medizinischer Bilddaten*. Habilitationsschrift, Otto-von-Guericke Universität Magdeburg, 2004.

[52] Preim, B.: *Entwicklung interaktiver Systeme*. Springer, Berlin, 1999.

[53] S., Pirner, K. Tingelhoff, K.W.G. Eichhorn, I. Wagner, M.E. Kunkel, A.I. Moral, M.E. Rilk, F.M. Wahl und F. Bootz: *Analysis of manual segmentation in paranasal CT images*. European Archives of Oto-Rhino-Laryngology, 265(9):1061–1070, 2008.

[54] Salah, Z.: *Segmentation and Illustrative Visualization of Medical Data*. Doktorarbeit, Wilhelm Schickard Institute for Computer Science, Graphical-Interactive Systems (WSI/GRIS), University of Tübingen, 2006.

[55] Schünke, M., E. Schulte und U. Schumacher: *Prometheus - LernAtlas der Anatomie*. Georg Thieme Verlag, 2007.

[56] Sederberg, T.W. und S.R. Parry: *Free-form deformation of solid geometric models*. SIGGRAPH Comput. Graph., 20(4):151–160, 1986.

[57] Seim, H.: *Modellbasierte Segmentierung von Lymphknoten in CT-Daten des Halses*. Diplomarbeit, Otto-von-Guericke Universität Magdeburg, 2006.

[58] Shneideman, B.: *Designing the user interface: strategies for effective human-computer interaction. 2. Auflage*. Addison-Wesley, 1992.

[59] Sorkine, O., Y. Lipman, D. Cohen-Or, M. Alexa, C. Rössl und H.P. Seidel: *Laplacian Surface Editing*. In: *Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*, Seiten 179–188. Eurographics Association, 2004.

[60] Tillmann, B.: *Atlas der Anatomie des Menschen*. Springer, Berlin, 2005.

[61] Tönnies, K.: *Grundlagen der Bildverarbeitung*. Pearson Studium, München, 2005.

[62] Toledo, S.: *Taucs: A library of sparse linear solvers, version 2.2*. http://www.tau.ac.il/ stoledo/taucs, 2003.

[63] Watkins, D.S.: *Fundamentals of matrix computations (Second edition)*. John Wiley & Sons, Inc., New York, NY, USA, 2002.

[64] Yan, J., T. Zhuang, B. Zhao und L.H. Schwartz: *Lymph node segmentation from CT images using fast marching method*. Computerized Medical Imaging and Graphics, 28(1-2):33 – 38, 2004.

[65] Yoshioka, Y., H. Masuda und Y. Furukawa: *A Constrained Least Squares Approach to Interactive Mesh Deformation*. In: *SMI '06: Proceedings of the IEEE International Conference on Shape Modeling and Applications 2006*, Seite 23, Washington, DC, USA, 2006. IEEE Computer Society.

[66] Zeleznik, R.C., K.P. Herndon und J.F. Hughes: *SKETCH: An Interface for Sketching 3D Scenes*. In: *Proceedings of SIGGRAPH 96*, Computer Graphics Proceedings, Annual Conference Series, Seiten 163–170, 1996.

Abbildungsverzeichnis

2.1 Kehlkopfskelett 6
2.2 Verschiedene Ausprägungen der Schildknorpelform 7
2.3 Lymphknoten im Kopf- und Halsbereich 8
2.4 Beispiele unterschiedlich geformter Lymphknoten 8
2.5 Beispiele verschiedenartiger Halstumoren 10
2.6 Modernes CT-Aufnahmesystem 11
2.7 Radiologische Schnittebenen 13
2.8 Schildknorpel im axialen CT-Bild 15
2.9 Verschiedene Lymphknoten im axialen CT-Bild 16
2.10 Kehlkopfkarzinom im axialen CT-Bild 17

3.1 Terminologische Zusammenhänge der Interaktionsbegriffe 21
3.2 Stabiles Feder-Masse-Modell des Kehldeckels 25
3.3 Initialisierung der SMSM Segmentierung des Schildknorpels 27
3.4 Interaktive Steuerung der Segmentierung mit *Snakes* 29
3.5 Framework for Interactive Segmentation 30
3.6 Lupenwerkzeug zur Manipulation von Polygonmodellen 34
3.7 Farbpinsel-Metapher zur Manipulation von Polygonmodellen 35
3.8 Beispiel einer möglichen Modellmanipulation bei Teddy 36
3.9 Skizzenbasierte Modellmanipulation nach Kho *et al.* 37

4.1 Manuelle Nachkorrektur 42
4.2 Dreidimensionale Visualisierung eines segmentierten Halstumors . . . 44
4.3 Fehlsegmentierungen des Schildknorpels 48

5.1 Darstellung der orthogonalen Schnittebenen eines CT-Datensatzes . . 53
5.2 Ghosting bei der Visualisierung einer Segmentierung 54
5.3 Multi Slice Viewer 55
5.4 Begrenzung des Einflussbereichs mittels Depth Limiter 57
5.5 Definierte ROI eines Polygonnetzes 59
5.6 Detailgrad bei SMSM des Schildknorpels 63
5.7 Nachkorrektur einer Segmentierung mit dem Bulge Tool 65

5.8 Problematisches Verhalten des Bulge Tool 67
5.9 Nachkorrektur einer Segmentierung mit dem Traction Tool 69
5.10 Nachkorrektur einer Segmentierung mit dem Sketch Tool 72
5.11 Geometrische Situation beim Sketch Tool 73
5.12 Nachkorrektur einer Segmentierung mit dem Bending Tool 75
5.13 Bestimmung der ROI beim Bending Tool 76

6.1 Klassenstruktur der *SMSMCorrectionToolbox* 84
6.2 GUI der Testumgebung *SMSMCorrectionToolboxTest* 85

7.1 Korrektur einer Untersegmentierung in z-Richtung 88
7.2 Korrektur der Segmentierung eines Schildknorpels 90