Aus dem Institut für Medizinische Informatik
der Universität zu Lübeck
Direktor: Prof. Dr. rer. nat. habil. H. Handels

“Die Entwicklung einer Java-basierten Bildverarbeitungssoftware für Telekonferenz und Lehre”

Inauguraldissertation
zur
Erlangung der Doktorwürde
der Universität zu Lübeck
- Aus der Technisch-Naturwissenschaftlichen Fakultät -

vorgelegt von
Heike Schmidt
aus Hannover
Lübeck 2010

1. Berichterstatter: Prof. Dr.-Ing. Dr. med. habil. S. J. Pöppl
2. Berichterstatter: Prof. Dr. rer. nat. M. Herczeg

Tag der mündlichen Prüfung: 17.12.2010

Zum Druck genehmigt. Lübeck, den 14.01.2011

Kurzfassung

Verfahren und Einsatzgebiete der Medizinischen Bildverarbeitung wie die computergestützte Generierung medizinischer Bilddaten sowie deren digitale Weiterverarbeitung und Archivierung zählen heutzutage ebenso zum Standard in Kliniken und Arztpraxen wie der Einsatz telemedizinischer Anwendungen zum computerbasierten Bilddatenaustausch oder zur telekooperativen Datenbesprechung.
Gerade telemedizinische Verfahren gewinnen im Zuge der Einrichtung einer Telematikinfrastruktur sowie neuer Versorgungsformen im Gesundheitswesen immer mehr an Bedeutung. Heute schon werden vielerorts Webportallösungen genutzt, um klinikübergreifend Bild- und Befundmaterial schnell kommunizieren zu können. In der Teleradiologie werden für die gemeinsame Besprechung dieser Daten Telekonferenztools eingesetzt, die jedoch selten losgelöst von kostenpflichtigen, komplexen Workstations genutzt werden können.
Daneben wird auch die Lehre in der Medizinischen Bildverarbeitung immer wichtiger. Bildverarbeitung ist allgemein ein schnell wachsendes Feld, daher sollte auch das Ausbildungsniveau hier angepasst werden. Obwohl der Einsatz von Konzepten und Techniken der Informatik den Bildungssektor insgesamt nicht unbeeinflusst gelassen hat, sind heute immer noch die meisten der verfügbaren Lernmaterialien für den Bereich der Medizinischen Bildverarbeitung papierbasiert. “Learning-by-doing” findet im Zusammenhang mit dem Einsatz von Telekonferenz- oder Bildverarbeitungstools in Medizin und Informatik kaum statt. Außerdem sind Angebote, die den Ablauf von Bildverarbeitungsalgorithmen verdeutlichen sowohl für die Fern- als auch für die Präsenzlehre schwer zu finden.

Unter dem Namen **JAMIP** (**Ja**va based **M**edical **I**mage **P**rocessing) ist daher nun am Institut für Medizinische Informatik der Universität zu Lübeck ein frei verfügbares Werkzeug für die Betrachtung und Bearbeitung medizinischer Bilddaten sowie für deren synchrone kooperative Besprechung zum Einsatz in Forschung, Lehre und Praxis entstanden, das insbesondere dem Einfluss der neuen Technologien im Gesundheitswesen und in der Lehre Rechnung trägt. Das Programm ist vollständig Java-basiert und damit weitestgehend plattformunabhängig, was die Ausführung auf jedem handelsüblichen PC ermöglicht.

Zusätzlich steht mit der Applet-Variante des Systems eine Version zur Verfügung, die eine Installation des Programms gar nicht erst erforderlich macht. Das Tool wurde unabhängig von anderer Software als flexibel einzusetzendes Bildverarbeitungsprogramm realisiert, um so eine Eingliederung in bereits bestehende Lösungen wie z. B. Webportale für die integrierte Versorgung oder in zukünftige Einrichtungen der Telematikinfrastruktur wie die elektronische Patientenakte leicht zu ermöglichen.
Einsatz- und Anwendungsszenarien finden sich sowohl in der Lehre als auch im klinischen Alltag. JAMIP konnte bereits in verschiedenen Veranstaltungen der Fern- und Präsenzlehre der Medizinische Bildverarbeitung für die Universität zu Lübeck und andere Institutionen erfolgreich eingesetzt werden. Daneben konnte JAMIPTele, die Telekonferenzanwendung des Systems, als Applet in die Gesundheitsplattform des Instituts für Medizinische Informatik integriert werden, weitere Einbindungen im Rahmen klinik- oder abteilungsübergreifender Serverlösungen sind hier denkbar.

Für die Anzeige und Bearbeitung medizinischer Bilddaten enthält JAMIP einen DICOM-Viewer sowie Standardfunktionen medizinischer Bildverarbeitungssoftware wie das Zoomen, die Invertierung oder die Level-Window-Operation. Um Bilddaten in einer telekooperativen Sitzung quasi-synchron anzeigen und bearbeiten zu können, wurde mit JAMIPTele ein eigenständiges Modul entwickelt, was aus JAMIP heraus gestartet werden kann. Des Weiteren beinhaltet das Programm die Möglichkeit, Bilddaten in verschlüsselter oder unverschlüsselter Form über ein Netzwerk zu verschicken.
Für den Einsatz des Systems als virtuelles Labor für die Lehre wurden ausgewählte Bildverarbeitungsmethoden aus dem Grundlagenbereich der Medizinischen Bildverarbeitung integriert, die nicht nur in ihrer “normalen” Ausführung zur Verfügung stehen sondern im sogenannten Animationsmodus schrittweise in ihrem Ablauf visualisiert werden. zum besseren Verständnis der Algorithmen wurden außerdem unterschiedliche Sichten auf die zugrunde liegenden Daten realisiert wie beispielsweise eine numerische Ansicht der Bilddaten.

Inhaltsverzeichnis

Kurzfassung 3

1 Einleitung 7

1.1 Problemstellung 8

1.2 Zielsetzung 9

1.3 Gliederung 10

2 Einordnung 13

2.1 Gesundheitstelematik 14

2.1.1 Telemedizin 15

2.1.2 Telematikinfrastruktur 18

2.1.3 Rechtliche Grundlagen, Datenschutz und Datensicherheit 20

2.2 E-Learning / Einflüsse der Technologieentwicklung auf das Bildungswesen 23

2.2.1 Begriffsdefinitionen E-Learning 23

2.2.2 Das Bmbf-Projekt medin 25

3 Grundlagen 27

3.1 Medizinische Bildverarbeitung 27

3.1.1 Bilderzeugung 29

3.1.2 Bildvorverarbeitung 34

3.1.3 Bildanalyse 35

3.1.4 Bildvisualisierung 36

3.1.5 Bildverwaltung 37

3.2 DICOM 38

3.2.1 Das DICOM-Dateiformat 39

3.3 Verschlüsselung 43

3.3.1 Symmetrische Verschlüsselung 44

3.3.2 Asymmetrische Verschlüsselung 44

3.3.3 Hybride Systeme 46

4 Existierende Ansätze **49**
4.1 Teleradiologiesysteme - Konzepte und Stand der Forschung 49
4.2 Telekonferenztools 51
4.3 DICOM-Viewer 53
4.4 Bildverarbeitungsprogramme für die Medizin 55
4.5 Software und Lernplattformen für die (Medizinische) Bildverarbeitung . 56
4.5.1 BIO - Bildverarbeitung Interaktiv Online der FH Regensburg . . 57
4.5.2 ODITEB - Open Distributed Textbook der TU München 57
4.5.3 Lehrgang Digitale Bildverarbeitung der FH Jena 59

5 Material und Methoden **61**
5.1 Java 61
5.1.1 Java Swing für grafische Benutzeroberflächen 62
5.1.2 Bilder in Java 63
5.1.3 Java RMI für entfernte Methodenaufrufe 64
5.1.4 Java Cryptography Architecture für die Verschlüsselung 65
5.1.5 Die Zusatzbibliothek JavaHelp für die Erstellung der Benutzeranleitung 66
5.2 Application und Applet 67
5.2.1 Applets und Sicherheit 67
5.2.2 Das Archivformat Jar 68
5.3 Das Telekooperationswerkzeug Jermes 68

6 JAMIP - Systementwurf und Implementierung **71**
6.1 Die Entwicklung des Basisprogramms 72
6.1.1 DICOM-Daten in JAMIP 74
6.2 Application und Applet 76
6.3 Die Realisierung der Telekonferenz 77
6.4 Datenversand 79
6.4.1 Der Datenversand in JAMIP 80
6.4.2 Die Verschlüsselung der Daten in JAMIP 81
6.5 Die Bildverarbeitung 82
6.5.1 Bedienungsanleitung 83
6.6 Übersicht der implementierten Klassen 84
6.7 Software-Übersicht 84

7 JAMIP - die Realisierung **87**

7.1 Die Benutzeroberfläche 89
7.2 Die Realisierung der Standardfunktionen 93
7.3 Die Medizinische Bildverarbeitung in JAMIP 97
7.3.1 Histogrammerstellung 98
7.3.2 Profilerstellung 100
7.3.3 Histogrammbasierte Grauwerttransformation 100
7.3.4 Filter 102
7.3.5 Schwellwertverfahren 107
7.3.6 Bereichswachstumsverfahren (Regiongrowing) 108
7.3.7 Histogrammbasierte Segmentierung 111
7.3.8 Aktive Konturmodelle - Snakes 113
7.3.9 K-Means-Clusteranalyse 115
7.3.10 Statistische Texturanalyse mit Cooccurrence-Matrizen 116
7.4 Datenversand und Verschlüsselung 119
7.4.1 Realisierung des Datenversandes in JAMIP 120
7.4.2 Realisierung der Verschlüsselung in JAMIP 121
7.5 Das Telekonferenz-Modul JAMIPTele 122
7.5.1 Start einer Telekonferenz mit JAMIPTele 123
7.5.2 Die Benutzeroberfläche von JAMIPTele 124
7.5.3 Die Bildverarbeitung in JAMIPTele 126

8 JAMIP - Einsatz und Vergleich 129
8.1 Einsatz 129
8.1.1 Einsatz JAMIP 129
8.1.2 Einsatz JAMIPTele 132
8.2 Vergleich 134
8.2.1 Vergleich mit DICOM-Viewern 134
8.2.2 Vergleich mit dem Telekonferenztool TeleConsult 135
8.2.3 Vergleich mit Bildverarbeitungstools für die Lehre 136

9 Fazit und Ausblick 139
9.1 Fazit 139
9.2 Ausblick 141

Literaturverzeichnis 143

Abbildungsverzeichnis 151

Tabellenverzeichnis 153

Abkürzungsverzeichnis 155

Links 157

1 Einleitung

Der technologische Fortschritt der letzten Jahrzehnte und die zunehmende Nutzung moderner Informations- und Kommunikationstechnologien haben auch im Bereich des Gesundheitswesens und im Bildungssektor Veränderungen herbeigeführt.
Im Bereich der Medizinischen Bildverarbeitung haben sich vielfältige neue Möglichkeiten ergeben. Die computergestützte Generierung medizinischer Bilddaten sowie deren digitale Weiterverarbeitung und Archivierung zählen heute ebenso zu den Standardverfahren in Kliniken und Arztpraxen wie der Einsatz telemedizinischer Anwendungen zum computerbasierten Bilddatenaustausch oder zur telekooperativen Datenbesprechung.
Für den klinischen Alltag sind in erster Linie kommerzielle Systeme entstanden, die hohen Anforderungen an Funktionalität, Bildauflösung und Bedienfreundlichkeit genügen und mit Bilddaten im in der Medizin üblichen DICOM-Format arbeiten. Diese Systeme setzten sich zumeist aus mehreren Teilbausteinen für die verschiedenen Aufgabengebiete im Bereich der Bildbetrachtung, -kommunikation und -archivierung zusammen. So gibt es beispielsweise für die Bildbetrachtung neben einfachen DICOM-Viewern Module für anspruchsvolle 2D- oder 3D-Befundungstools, für die Bildarchivierung Module für eigenständige PACS-Systeme oder für die Teleradiologie Module für den Datenversand und die kooperative Befundung.
Die Zunahme der sektorübergreifenden und integrierten Versorgungsformen im Gesundheitswesen, der Aufbau einer Telematikinfrastruktur mit klinikübergreifenden Patientenakten sowie die Verfügbarkeit flächendeckender Netzwerkinfrastrukturen fördern die Verbreitung telemedizinischer Anwendungen. Heute stehen Bild- und Befunddaten für eine schnelle Kommunikation zwischen verschiedenen Akteuren des Gesundheitswesens wie z. B. Klinikärzten und zuweisenden Hausärzten oft auf einem gemeinsam zugänglichen Webserver zur Verfügung. Während den Nutzern die Auswahl eines Programms für die Bildbetrachtung und -kommunikation der Serverdaten zumeist frei steht und den jeweils vorherrschenden Bedingungen in Klinik oder Praxis angepasst werden kann, so erfordert die Durchführung einer Telekonferenz in der Regel die Nutzung einer gemeinsam verfügbaren kommerziellen Software. Vor dem Hintergrund der steigenden Kosten im Gesundheitswesen und der hier oft vorherrschenden heterogenen Computersysteme

erscheint der Einsatz eines kostenlosen, plattformunabhängigen Tools für die kooperative Befundung medizinischer Bilddaten sinnvoll.

Der Einsatz von Konzepten und Techniken der Informatik und Informationstechnologie hat auch den Bildungssektor nicht unbeeinflusst gelassen. Aspekte wie E-Learning und der Gebrauch des Internet als didaktisches Medium spielen heute eine immer größere Rolle. Mit dem Einsatz von Computern und Lernsoftware wird im schulischen und universitären Bereich versucht, das Lernen mit Hilfe von Animationen, Grafiken, Videos und interaktiven Selbstlerntests zu optimieren. Fernlernangebote wie das Fernstudium werden durch Online-Plattformen und integrierte Tools erweitert.
Gerade in Schnittstellenbereichen wie der Medizinischen Bildverarbeitung, wo es auch um das Kennenlernen realistischer Softwaresysteme geht, bieten Programme zum Ausprobieren einen deutlichen Mehrwert gegenüber der reinen Textversion. Erstaunlicherweise findet sich jedoch speziell für diesen Bereich immer noch relativ wenig Lernsoftware. Reine Bildverarbeitungsprogramme sind in den seltensten Fällen für den Einsatz in der Lehre der Medizinischen Bildverarbeitung geeignet, sie bieten entweder nicht den medizinischen Fokus oder sind für so komplexe Fragestellungen konzipiert, dass ein Einstieg in die Thematik mit ihnen schwerfällt. Gerade im Bereich der Grundlagenalgorithmen bietet sich zur Verständnisunterstützung eine animierte Darstellung interner Algorithmenabläufe für viele Verfahren an, dennoch erfolgt die Lehre hier nach wie vor in erster Linie papierbasiert. Der Einsatz eines Tools in Studium und Fernstudium der Medizinischen Bildverarbeitung zur Vermittlung eines grundlegenden Verständnisses für den Umgang mit digitalen medizinischen Bilddaten, Bildverarbeitungsalgorithmen - inklusive deren detaillierter Abläufe - und Telekooperationssoftware wäre daher wünschenswert.

1.1 Problemstellung

Telemedizinische Verfahren gewinnen im Zuge der Einrichtung einer Telematikinfrastruktur sowie neuer Versorgungsformen immer mehr an Bedeutung. Heute schon werden vielerorts Webportallösungen genutzt, um klinikübergreifend Bild- und Befundmaterial schnell kommunizieren zu können. Gerade in der Teleradiologie werden für die gemeinsame Besprechung dieser Daten Telekonferenztools eingesetzt, die jedoch selten losgelöst von kostenpflichtigen, komplexen Workstations genutzt werden können.
Daneben wird auch die Lehre in diesem Bereich immer wichtiger. Bildverarbeitung ist allgemein ein schnell wachsendes Feld, auch in der Medizin. Daher muss auch das Ausbildungsniveau hier angepasst werden. Immer noch sind die meisten der heutzutage

verfügbaren Lernmaterialien für diesen Bereich papierbasiert, “learning-by-doing” findet im Zusammenhang mit dem Einsatz von Telekonferenz- oder Bildverarbeitungstools in Medizin und Informatik kaum statt. Außerdem sind Angebote, die den Ablauf von Bildverarbeitungsalgorithmen verdeutlichen sowohl für die Fern- als auch für die Präsenzlehre schwer zu finden.
Die Problemstellung für diese Arbeit gliedert sich entsprechend der beiden beschriebenen Kernthemen - Telekonferenz und Lehre - in zwei Teile:

Problemstellung 1: Es gibt kein kostenloses, plattformunabhängiges Telekonferenztool für die Betrachtung und Bearbeitung medizinischer Bilddaten, was sich unabhängig von PACS-Workstations oder großen Komplettlösungen einfach in bestehende Strukturen integrieren lässt.

Problemstellung 2: Es gibt keine Lehr- und Lernsoftware im Bereich der Medizinischen Bildverarbeitung zum interaktiven Arbeiten mit medizinischen Bilddaten, um Studenten ein Grundverständnis für Bildverarbeitungs- und Bildkommunikationssoftware zu vermitteln und um den Lehrenden bei der Präsentation der Lehrinhalte - insbesondere bei der Vermittlung der Grundlagenalgorithmen - zu unterstützen.

1.2 Zielsetzung

Das Ziel dieser Arbeit ist die Entwicklung eines kostenlosen, plattformunabhängigen Telekooperationstools, was mehreren Nutzern gleichzeitig die Möglichkeit bietet, gemeinsam genutztes medizinisches Bildmaterial in einer Telekonferenz zu betrachten. Gleichzeitig soll dieses System auch in der Lehre einsetzbar sein und dafür Grundlagenalgorithmen aus dem Bereich der Medizinischen Bildverarbeitung integrieren, die für die Lehre angemessen aufbereitet werden sollen.
Die Plattformunabhängigkeit soll bei der Entwicklung des in dieser Arbeit umzusetzenden Programms dazu beitragen, die Voraussetzungen für den Einsatz dieses Systems auf die Einrichtung eines multimedialen Arbeitsplatzes mit einem marktüblichen, vernetzten PC zu reduzieren und so hohe Investitionskosten zum Kauf neuer Rechensysteme oder plattformgebundener Softwarelizenzen vermeiden.

Ziel 1: Es soll eine plattformunabhängige Software entwickelt werden, die die telekooperative Besprechung medizinischer Bilddaten möglichst unabhängig von bestehenden Strukturen und bereits vorhandener Software ermöglicht. Basis des Programms soll dabei ein einfaches Bildverarbeitungstool sein, was medizinische

Bilddaten im DICOM-Format einlesen, anzeigen und mit Standardfunktionen bearbeiten kann. Um den Anforderungen an medizinische Bedürfnisse gerecht zu werden, soll der Aufbau der Benutzeroberfläche des Programms an der üblicher Bilddarstellungssoftware für die Medizin wie z. B. DICOM-Viewern orientiert sein.

Ziel 2: Das in Ziel 1 entwickelte Programm soll auch als Lehr- und Lernsoftware in der Informatik und der Medizin im Bereich der Medizinischen Bildverarbeitung eingesetzt werden können, um den Umgang mit Bildverarbeitungs- und Bildkommunikationssoftware sowie Kenntnisse über Methoden der digitalen Bildverarbeitung und deren Ablauf zu vermitteln. Dafür sollen in der Vorlesung vorgestellte Grundlagenalgorithmen aus dem Bereich der Medizinischen Bildverarbeitung in das Programm integriert und lehr-/lernunterstützend präsentiert werden. Zu diesem Zweck soll ein spezieller "erklärender Modus" realisiert werden, der den Ablauf eines Verfahrens im Detail visualisiert.

Die Software soll sich einerseits in den Fernstudiengang Medizinische Informatik, den das Institut für Medizinische Informatik in Kooperation mit der FernUniversität Hagen anbietet, einbinden lassen und andererseits auch in den Präsenzveranstaltungen an der Universität zu Lübeck genutzt werden können.

1.3 Gliederung

Die vorliegende schriftliche Fassung gibt in den folgenden Kapiteln einen Überblick über die für die Entstehung und Entwicklung dieser Arbeit relevanten Themen. Kapitel 2 gibt dabei zunächst mit der Darstellung der durch den technologischen Wandel begründeten Veränderungen im Gesundheitswesen und Bildungsbereich einen Einblick in die Rahmenbedingungen des hier zu entwickelnden Projekts. In Kapitel 3 werden dann theoretische Grundlagen aus den Bereichen Medizinische Bildverarbeitung, DICOM und Verschlüsselung beschrieben, bevor Kapitel 4 einen Überblick über derzeit verfügbare Softwaresysteme in den Bereichen Telekonferenz und Lehre für die Medizinische Bildverarbeitung liefert. Anschließend werden in Kapitel 5 die für die Erstellung dieser Arbeit verwendeten Softwaretechniken vorgestellt, mit denen die Implementierung des in Kapitel 6 beschriebenen Systemkonzepts erfolgte. Das realisierte Programm wird in dem darauf folgenden Kapitel 7 vorgestellt und in Kapitel 8 werden dann die Veranstaltungen und Projekte, in deren Rahmen die Software bereits eingesetzt wird, beschrieben. Des Weiteren erfolgt hier ein Vergleich mit zuvor beschriebenen aktuellen Softwaresystemen

für die Telekonferenz und Lehre. Kapitel 9 liefert abschließend eine kritische Betrachtung der entstandenen Arbeit sowie einen Ausblick auf künftige Erweiterungsmöglichkeiten.

2 Einordnung

Die Erfindung des Mikrochips Ende der 1950er Jahre löste analoge Schaltungen durch digitale Algorithmen ab und damit einen Umbruch - die sogenannte *Digitale Revolution* - aus, der seit Ausgang des 20. Jahrhunderts einen technologischen und kulturellen Wandel herbeiführt.

Eine entscheidende Rolle spielte hierbei die Entwicklung des Computers, die bereits in den 1940er Jahren mit dem Bau des ersten funktionsfähigen Digitalrechners durch Konrad Zuse begann. Seit der Einführung des *Personal Computers* Mitte der 1970er Jahre finden Computer nicht mehr nur in Forschung und Beruf Anwendung, sondern verstärkt auch im privaten Bereich.

Parallel dazu entstand 1969 mit dem *Arpanet (Advanced Research Projects Agency Network)* ein Vorläufer des heutigen *Internet*, das Universitäten und Forschungseinrichtungen miteinander vernetzte. Erst 1990 wurde das Internet für die kommerzielle Nutzung freigegeben und mit dem zeitgleich von dem Briten Tim Berners-Lee herausgegebenen *World Wide Web-Standard* einer breiten Öffentlichkeit zugänglich gemacht.

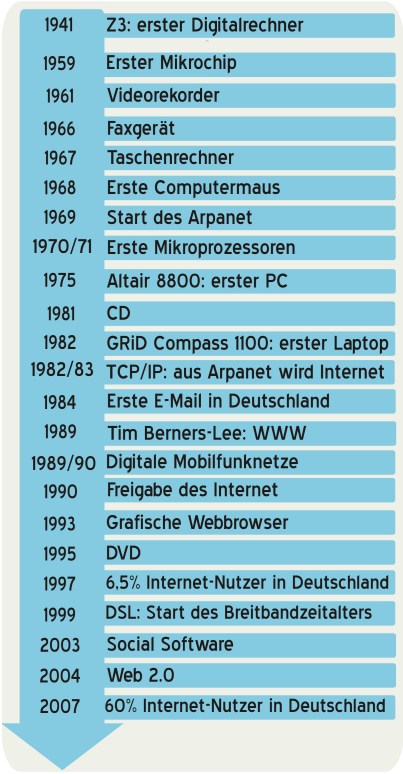


Abbildung 2.1: Die digitale Revolution im zeitlichen Überblick

Heute hat sich das Internet durch die beständig voranschreitende Digitalisierung von Informationen, Technologien und Medien zu einem “all-in-one”-Medium entwickelt [10]. Nach und nach werden traditionelle Medien ersetzt und zusätzlich eigene neue Formen wie z. B. Versteigerungsbörsen, “Mitmach”-Enzyklopädien, Diskussionsforen, Video-Communities oder Social Networks angeboten. Seit dem Jahr 2004 spricht man vom *Web 2.0* und meint damit den Wandel des Nutzers von einem passiv surfenden Medienkonsumenten hin zu einem aktiven gestaltenden Anwender.

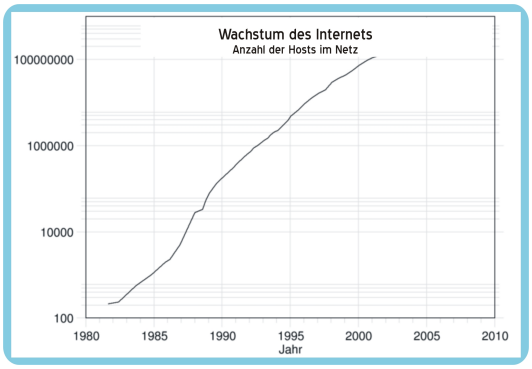


Abbildung 2.2: Das Wachstum des Internets interpretiert anhand der Anzahl zur Verfügung stehender Hosts - die Ordinate des Diagramms wurde logarithmisch skaliert

In Deutschland stieg die Zahl der Internetnutzer von 6,5% im Jahr 1997 auf knapp 60% (bzw. 80% der unter 50-Jährigen) im Jahr 2007 [11]. Kein anderes Medium konnte bisher innerhalb so kurzer Zeit einen so hohen Verbreitungsgrad erreichen. Das Internet ist zu dem Standardmedium für die Verbreitung von Informationen jeder Art geworden und hat das Industriezeitalter durch das *Informationszeitalter* abgelöst, dessen Kern die digitalisierte *Information* als Rohstoff und Ware darstellt.

In den folgenden Abschnitten dieses Kapitels werden die auf die technologische Entwicklung der letzten Jahre zurückzuführenden Einflüsse auf das Gesundheitswesen (Abschnitt 2.1) und den Bildungsbereich (Abschnitt 2.2) näher erläutert. Es wird sowohl grundlegendes Wissen dieser Anwendungsgebiete erörtert als auch Begriffsdefintionen gegeben.

2.1 Gesundheitstelematik

Die Verfügbarkeit flächendeckender IT-Infrastrukturen hat auch in vielen Bereichen der Medizin eine Digitalisierung herbeigeführt. Medizinische Daten werden nicht mehr nur elektronisch gespeichert, sondern auch elektronisch über Netzwerke übertragen. Die Akteure im Gesundheitswesen werden zunehmend und umfassend elektronisch miteinander vernetzt, die Prozesse werden digital abgebildet. So erfolgt der Umgang mit konventionellen Patientendaten einschließlich administrativer Anwendungen wie z. B. der Abrechnung längst computerbasiert. Insbesondere für stark technisierte Bereiche der Medizin - hierzu zählen vor allen Dingen die geräteabhängigen Bereiche wie z. B. die bildgebenden Verfahren (siehe dazu Kapitel 3.1.1) oder die Intensivmedizin - hat der Einsatz von IuK-Technologien neue Möglichkeiten und Anwendungsszenarien eröffnet.

Mit dem Begriff *Gesundheitstelematik* wird diese Nutzung moderner Informations- und Kommunikationstechnologien im Gesundheitswesen zur Verbesserung und Effektivierung der Patientenversorgung zusammengefasst. International gebräuchlich ist auch der Begriff *E-Health*. [12, 13]
Anwendungen der Gesundheitstelematik lassen sich in verschiedene Stufen einteilen:

- **Information:** Das Bereitstellen von Informationen für Patienten oder Ärzte über Informationsportale
- **Kommunikation:** Der Austausch von Informationen zwischen zwei Beteiligten (Patient - Arzt, Arzt - Arzt usw.) ohne direkte und zeitnahe Reaktion des Kommunikationspartners
- **Interaktion:** Der Austausch von Informationen oder Daten zwischen Beteiligten mit unmittelbarer Reaktion des Kommunikationspartners (z. B. Telekonferenz)
- **Transaktion:** Der gezielte Datenaustausch zwischen verschiedenen Partnern, mit dem Ziel, die Erbringung medizinischer Leistungen vollständig elektronisch abbilden und abwickeln zu können (z. B. elektronische Gesundheitskarte)
- **Integration:** Die Zusammenführung aller Daten eines Patienten aus medizinischen und paramedizinischen Bereichen sowie die Ergänzung durch Angaben und Einträge des Patienten selbst (z. B. elektronische Patientenakte)

Im folgenden Abschnitt wird die *Telemedizin* als Teilbereich der Gesundheitstelematik näher beschrieben, bevor in Abschnitt 2.1.2 kurz auf die Einrichtung einer *Telematikinfrastruktur* für das deutsche Gesundheitswesen eingegangen wird. Anschließend beschreibt Abschnitt 2.1.3 Sicherheitsaspekte und rechtliche Grundlagen beim Einsatz von Informations- und Kommunikationstechnologien im Gesundheitswesen.

2.1.1 Telemedizin

Allgemein versteht man unter dem Begriff *Telemedizin* die Erbringung medizinischer Dienstleistungen, die über eine Entfernung hinweg unter Nutzung von Informations- und Kommunikationstechnologie stattfinden. Dabei können beispielsweise Konsultationen zwischen sich an verschiedenen Orten befindenden Experten realisiert oder räumliche Entfernungen zwischen Arzt und Patient überbrückt werden. Die Telemedizin ist ein vergleichsweise neues Tätigkeitsfeld im Gesundheitswesen und konnte in ihrer heutigen Form erst durch eine flächendeckende IT-Infrastruktur entstehen. [14]

In größerem Umfang werden telemedizinische Verfahren seit den 1980er Jahren erprobt, erste Projekte gab es in den USA mit der Fernüberwachung des Gesundheitszustandes von Astronauten im Weltraum schon in den 1960er Jahren. Auch heutzutage gehört die Versorgung entlegener Gebiete, also die telekommunikationstechnische Anbindung von Gebieten mit geringerer Ausstattung hinsichtlich medizinischer Einrichtung oder Personal, zu den typischen Anwendungsgebieten der Telemedizin (siehe z. B. die Realisierung eines landesweiten Teleradiologie-Netzwerkes in Grönland [15]). So werden unter anderem Telekonsultationen mit Spezialisten zur Diagnosestellung und -sicherung ermöglicht.
Daneben gibt es immer mehr Projekte die zur Bildung eines offenen und verteilten Gesundheitsnetzes regional und überregional Unikliniken, kommunale Krankenhäuser, niedergelassene Ärzte und Labors miteinander vernetzen. So soll sowohl Expertenwissen als auch ein zentrales Bild- und Informationsarchiv sowie die zugehörigen Verarbeitungssysteme bereitgestellt werden (siehe dazu z. B. das *Aktionsprogramm Telemedizin in NRW* unter `http://www.egesundheit.nrw.de`).
Unter anderem zählen folgende Bereiche in den Anwendungskontext der Telemedizin:

- der Bilddatentransfer,
- die Telearchivierung medizinischer Bilddaten und Informationen,
- Telekonferenzen und Expertenkonsultationen,
- die Teleoperation,
- das Telecomputing (wie z. B. 3D-Visualisierungen),
- das Telemonitoring sowie
- die Telelernangebote für Aus- und Weiterbildung.

Zielrichtung und Aufwand für diese unterschiedlichen Anwendungsgebiete können dabei stark variieren. So ist z. B. für die Telekonsultation in ihrer einfachsten Ausprägung - per Telefon - sicherlich der geringste technische Aufwand vonnöten, wohingegen die Durchführung minimalinvasiver Teleoperationen nur unter Einsatz größtmöglicher technischer Ausstattung machbar ist.
Insgesamt soll durch die Umgestaltung medizinischer Dienstleistungsprozesse mit Telematik die Versorgung im Gesundheitswesen effizienter gestaltet und auch qualitativ verbessert werden.

2.1.1.1 Teleradiologie

Die *Teleradiologie* bildet einen Teilbereich der Telemedizin, unter dem eine Fernübertragung von radiologischen Bildern für die Interpretation und wissenschaftliche Kooperation oder Konsultation verstanden wird. Sie ist vermutlich das am weitesten entwickelte Teilgebiet der Telemedizin und konnte sich parallel zur Verbreitung des DICOM-Standards (siehe dazu Abschnitt 3.2) entwickeln. Die dabei entstandenen Systeme beinhalten meist Möglichkeiten für den Datenversand oder den Datenimport sowie für die Anzeige und Bearbeitung medizinischer Bilddaten, des Weiteren stellen sie in der Regel ein Modul für die telekooperative Befundung zur Verfügung. Beispiele für solche Programme finden sich in Kapitel 4.
Für die Teleradiologie lassen sich zwei große Anwendungsgebiete definieren: die *Teleradiologie nach Röntgenverordnung (RöV)* und die *Telekonsultation*. Unter ersterem wird die bildgebende Untersuchung eines Patienten durch einen Radiologen verstanden, der sich nicht am Ort der Durchführung befindet, sondern mittels elektronischer Datenübertragung mit diesem in Verbindung steht. So können auch kleinere Krankenhäusern insbesondere in der Nacht, an Sonn- oder Feiertagen, moderne bildgebende Verfahren anbieten, ohne dass immer ein Radiologe vor Ort sein muss.
Die zweite Form der Teleradiologie, die Telekonsultation, hat erheblich niedrigere gesetzliche Hürden zu überwinden, da sich hier immer ein verantwortlicher Radiologe am Ort der Bilderstellung und Untersuchung befindet. Dieser holt in besonders schwierig zu diagnostizierenden oder zweifelhaften Fällen eine sogenannte *Zweitmeinung* eines Experten für die Erstellung eines Befundes ein. [16]
In beiden Fällen wird der Bilddatensatz nach der Erstellung an den Radiologen oder Experten übermittelt und von diesem begutachtet. In der Regel wird der Befund anschließend per E-Mail, Fax oder Telefon zurückgeschickt. Insbesondere für die Telekonsultation existieren daneben jedoch auch Softwareprogramme, die einen Datenversand oder einen Zugriff auf Daten auf einem Webserver und eine anschließende gemeinsame kooperative Betrachtung der Daten über ein Netzwerk erlauben. Generell können an so einer *Telekonferenz* zwei oder mehr Kommunikationspartner, die sich nicht am gleichen Ort befinden, teilnehmen. Sie sehen dabei dasselbe auf ihren jeweiligen Monitoren und können die Anwendungen der anderen Teilnehmer fernsteuern, d. h. die Bearbeitungsfunktionen werden gleichzeitig bzw. quasi-synchron auf allen Seiten durchgeführt. Die Kommunikation selbst kann auf unterschiedlichen Wegen erfolgen, je nach System ist eine Video-, Voice-over-IP- oder Chatfunktion eingebaut.

Näheres zu den rechtlichen Rahmenbedingungen und Sicherheitsaspekten der Teleradiologie kann Abschnitt 2.1.3.1 entnommen werden.

2.1.2 Telematikinfrastruktur

Trotz permanent fortschreitender Digitalisierung und IT-Integration existiert in der deutschen Gesundheitslandschaft bis heute keine einheitliche, einrichtungsübergreifende IT-Infrastruktur, die eine bundesweite Kommunikation zwischen den vorhandenen Systemen der verschiedenen Organisationen und Einrichtungen im Gesundheitswesen ermöglicht. So ist z. B. die Krankengeschichte eines Patienten üblicherweise bei verschiedenen Ärzten abgelegt. Projekte zur Nutzung einer gemeinsamen Infrastruktur sind meist regional begrenzt und werden einrichtungsbezogen koordiniert. Um die Wirtschaftlichkeit, Qualität

Abbildung 2.3: Stufenweise Einführung der Funktionen der elektronischen Gesundheitskarte

und Transparenz der Patientenversorgung zu verbessern (siehe SGB V § 291a Abs. 1), plant der Gesetzgeber nun mit der Einrichtung einer *Telematikinfrastruktur* den Aufbau eines sicheren Kommunikationsnetzes für das Gesundheitswesen, das niedergelassene Ärzte, Zahnärzte, Krankenhäuser, Apotheken sowie die gesetzlichen und privaten Krankenkassen miteinander vernetzt und Zugangsportale für Versicherte bereitstellt. In SGB V §291 a Abs. 7 heißt es dazu:

> Der Spitzenverband Bund der Krankenkassen, die Kassenärztliche Bundesvereinigung, die Kassenzahnärztliche Bundesvereinigung, die Bundesärztekammer, die Bundeszahnärztekammer, die Deutsche Krankenhausgesellschaft sowie die für die Wahrnehmung der wirtschaftlichen Interessen gebildete maßgebliche Spitzenorganisation der Apotheker auf Bundesebene schaffen die für die Einführung und Anwendung der elektronischen Gesundheitskarte, insbesondere des elektronischen

> Rezeptes und der elektronischen Patientenakte, erforderliche interoperable und kompatible Informations-, Kommunikations- und Sicherheitsinfrastruktur (Telematikinfrastruktur).

Basiselement dieser neuen IT-Infrastruktur ist die *elektronische Gesundheitskarte (eGK)*, deren Einführung im Oktober 2009 in Nordrhein-Westfalen begann und bis Ende 2010 bundesweit abgeschlossen sein soll. Die eGK ist eine Erweiterung der 1995 eingeführten Krankenversichertenkarte und enthält zunächst einmal wie diese die sogenannten administrativen Daten wie z. B. Name und Geburtsdatum des Inhabers. Daneben ist die Karte mit einem Lichtbild versehen und beinhaltet auf der Rückseite die europäische Krankenversichertenkarte. Des Weiteren verfügt die neue Karte über einen programmierbaren Mikroprozessorchip, mit dem Daten verschlüsselt gespeichert und über eine Schnittstelle kommuniziert werden können.
Durch diesen Chip lässt sich die elektronische Gesundheitskarte flexibel gestalten, nach und nach sollen so neue Dienste wie beispielsweise das elektronische Rezept, die Arzneimitteldokumentation oder ein Notfalldatensatz zur Karte hinzugefügt werden. [17, 18]
Langfristig gesehen soll die Karte auch Anwendungen wie die *elektronische Patientenakte* unterstützen, in der Daten über den Behandlungsverlauf eines Patienten für eine fall- und einrichtungsübergreifende Dokumentation gesammelt und verwaltet werden sollen (siehe SGB V §291 a Abs. 3, `http://www.gesetze-im-internet.de/sgb_5/__291a.html`).

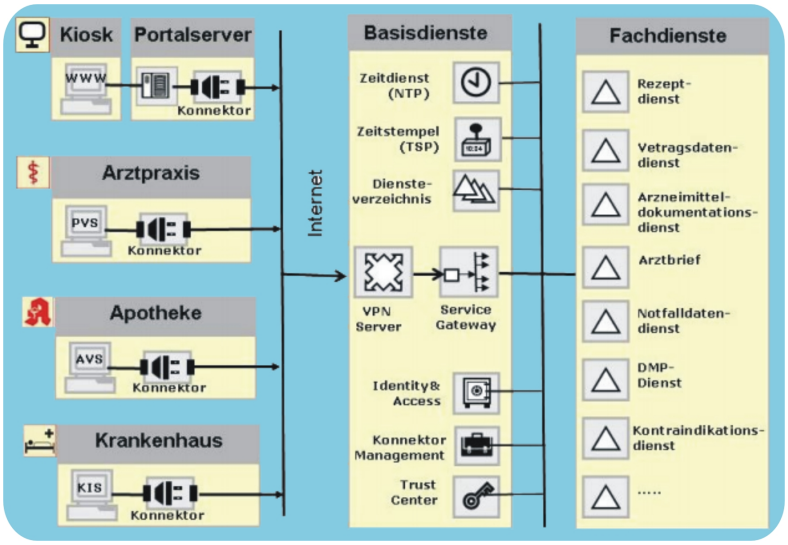


Abbildung 2.4: Aufbau der Telematikinfrastruktur in Deutschland

Folgende Komponenten gehören zur Telematikinfrastruktur (siehe Abbildung 2.4):

- die *Informationsinfrastuktur* bestehend aus:
  - zentralen Infrastrukturdiensten für den sicheren Zugriff und die Datenspeicherung wie z. B. Zertifikats-, Verzeichnis- und Berechtigungsdienste
  - Fachdiensten, z. B zur Bereitstellung der Versichertenstammdaten, Verarbeitung der Verordnungsdaten oder für freiwillige Anwendungen des Versicherten wie der Notfalldatendienst etc.
- die *Kommunikationsinfrastruktur* bestehend aus:
  - einer hochverfügbaren und breitbandigen Netzwerkinfrastruktur
  - dem *Konnektor*, der die Kommunikation zwischen den Primärsystemen (z. B. Arzt-PC), eGK, HBA/SMC und der Telematikinfrastruktur koordiniert
- die *Sicherheitsinfrastruktur* bestehend aus:
  - dem *elektronischen Heilberufsausweis (HBA)*, mit dem sich Ärzte und Apotheker beim Zugriff auf medizinische Daten ausweisen
  - der *Instituts-* oder *Secure Module Card (SMC)*, die die Authentifizierung von Arztpraxen, Krankenhäusern oder Apotheken übernimmt

Näheres zu den rechtlichen Rahmenbedingungen und Sicherheitsaspekten der Telematikinfrastruktur kann Abschnitt 2.1.3.2 entnommen werden.

2.1.3 Rechtliche Grundlagen, Datenschutz und Datensicherheit

Der Einsatz der Gesundheitstelematik soll zur Verbesserung der Kommunikation zwischen den einzelnen Institutionen des Gesundheitswesens beitragen und gleichzeitig die Behandlungsqualität optimieren. Dabei muss die hohe Sensibilität der zu verarbeitenden und zu kommunizierenden personenbezogenen Patientendaten beachtet werden. Den rechtlichen Rahmen, an dem sich die medizinische Datenverarbeitung zu orientieren hat, bilden hier das Datenschutzgesetz (BDSG) sowie die in der ärztlichen Berufsordnung und in § 203 StGB normierte Schweigepflicht.
In § 9 BDSG werden folgende Sicherheitsmaßnahmen für die technische und organisatorische Gewährleistung des Schutzes personenbezogener Daten definiert:

- *Vertraulichkeit:* nur Befugte dürfen personenbezogene Daten zur Kenntnis erhalten bzw. davon Kenntnis nehmen - die Daten müssen sicher einem Empfänger zugeordnet werden können
- *Authentizität:* die Daten müssen sicher einem Sender zugeordnet werden können

- *Integrität:* die Daten müssen ohne Übertragungsfehler oder Manipulation versendet werden bzw. müssen Übertragungsfehler und Manipulation erkennbar sein
- *Verfügbarkeit:* die technischen Komponenten (Hard- und Software) müssen einwandfrei funktionieren, damit die Daten zeitgerecht zur Verfügung stehen und ordnungsgemäß verarbeitet werden können
- *Revisionsfähigkeit:*alle Verarbeitungsschritte müssen lückenlos nachvollziehbar sein, so dass jederzeit festgestellt werden kann, wer wann was mit den Daten gemacht hat
- *Validität:* die Daten müssen aktuell in der für den Nutzungszweck angemessenen Qualität verarbeitet werden, bei Bilddaten betrifft dies insbesondere Qualitätsmerkmale wie Bildauflösung und Farbechtheit
- *Rechtssicherheit:* für jeden Verarbeitungsvorgang und dessen Ergebnis besteht Nachweispflicht
- *Nicht-Abstreitbarkeit von Datenübermittlungen:* Sender und Empfänger müssen eindeutig identifizierbar sein
- *Nutzungsfestlegung:* medizinische Datenverarbeitungssysteme müssen Nutzungsrechte festlegen können

Zur Realisierung dieser Sicherheitsziele sind für jedes medizinische Datenverarbeitungssystem auf Grundlage einer Bedrohungs- und Risikoanalyse die individuell erforderlichen Sicherheitsmaßnahmen zu ermitteln. Konkrete gesetzliche Vorgaben zur Umsetzung gibt es jedoch nicht. Softwareprogramme, die patientenbezogene Daten über ein Netz kommunizieren, sollten diese zur Wahrung der Vertraulichkeit mit starken kryptografischen Verfahren (siehe Abschnitt 3.3) verschlüsseln sowie Authentizität, Integrität und Revisionsfähigkeit durch elektronische Signaturen gewährleisten. [19]

2.1.3.1 Recht und Sicherheit bei der Teleradiologie

In § 2 der Röntgenverordnung (RöV) sind die Voraussetzungen für die Teleradiologie nach RöV definiert. Hier sind Qualitätssicherungsmaßnahmen auf technischer und organisatorischer Ebene festgelegt, z. B. benötigen Betreiber von Teleradiologieeinrichtungen immer eine Genehmigung der Aufsichtsbehörde, wenn kein Radiologe bei Untersuchungen unter Einsatz von Röntgenstrahlen persönlich anwesend ist. Daneben existiert mit

DIN 6868-159 seit März 2009 eine Norm für die Abnahme- und Konstanzprüfung in der Teleradiologie nach Röntgenverordnung.
Dennoch gibt es bis heute für den Versand von Bilddaten in der Teleradiologie keinen Standard. Oft werden personenbezogene Patientendaten zum Zweck des Einholens einer Zweitmeinung einfach unverschlüsselt an E-Mails angehängt. Die Deutsche Röntgengesellschaft empfiehlt als Minimalstandard für die Versendung DICOM-konformer Bilddaten (siehe Abschnitt 3.2) die Nutzung von *DICOM E-Mail*. Die Bilder werden dabei als MIME-Attachments an E-Mails angehängt und mit einem OpenPGP-Plug-In verschlüsselt [20, 21]. Durch diese Technik wird die Authentizität, Identität, Vertraulichkeit und Eindeutigkeit von Daten nach dem aktuellen Stand der Technik gewährleistet. [22]

2.1.3.2 Recht und Sicherheit der Telematikinfrastruktur

Für die technische Organisation und Realisierung des bundesweiten Einsatzes der elektronischen Gesundheitskarte zeichnet sich die *gematik - Gesellschaft für Telematikanwendungen der Gesundheitskarte mbH* verantwortlich, deren Gründung in § 291b SGB V gesetzlich festgeschrieben ist.
Die Grundlage zur Einführung der elektronischen Gesundheitskarte und damit verbundener Infrastrukturelemente wurde bereits 2003 mit dem *Gesetz zur Modernisierung der gesetzlichen Krankenversicherungen (GKV-Modernisierungsgesetz)* geschaffen. In § 291a SGB V ist festgelegt, dass die Karte neben einem verpflichtenden administrativen Teil über einen freiwilligen medizinischen Teil verfügt.
Des Weiteren regelt dieser Paragraph auch den Datenschutz und die Datensicherheit. Die Zugriffsrechte auf die vorhandenen Daten liegen allein beim Versicherten. Er bestimmt, welche Funktionen seiner Gesundheitskarte über die Versicherungsangaben und das elektronische Rezept hinaus genutzt werden sollen. Leistungserbringer dürfen die Gesundheitsdaten erst mit Einwilligung des Patienten speichern, verarbeiten und nutzen und müssen sich beim Zugriff auf die eGK mit ihrem elektronischen Heilberufsausweis identifizieren.

Nach [23] gelten die in der Gesundheitstelematik verwendeten SmartCards als das sicherste elektronische System außerhalb geschützter Rechenzentren. Einige der umgesetzten Sicherheitsmechanismen werden nachfolgend im Überblick beschrieben. Komponenten und Dienste der Telematikinfrastrukur müssen den Anforderungen der von der gematik herausgegebenen Richtlinien bezüglich Funktionsfähigkeit, Interoperabilität und Sicher-

heit genügen. Dabei sind die Sicherheitsanforderungen für den Einsatz kryptografischer Verfahren in der technischen Richtlinie TR-03116 des Bundesamtes für Sicherheit in der Informationstechnik (`http://www.bsi.bund.de`) verbindlich festgeschrieben. [24]
Sicherheitsmechanismen im Überblick:

- das *Zweikartenprinzip*: Sowohl Leistungserbringer als auch der Versicherte müssen sich über ihre jeweiligen Karten (eGK und HBA) und durch Eingabe einer zugehörigen PIN authentifizieren, um auf die Gesundheitsdaten der eGK zugreifen zu können

- die letzten 50 Zugriffe werden auf der eGK gespeichert

- die Daten auf der eGK werden kryptografisch mit einem hybriden System (siehe [20] und 3.3) nach heutigem Stand der Technik verschlüsselt (symmetrisches AES-Verfahren, asymmetrisches RSA-Verfahren, Schlüssellänge von 2048 Bit)

- sollen die Daten in die Telematikinfrastruktur eingepflegt werden, so übernimmt der Konnektor die Verschlüsselung, die Generierung einer Prüfsumme für die Kontrolle der Datenintegrität, die Sicherung der Datenauthentizität durch Signatur des Arztes und die Bereitstellung eines abhörsicheren Transportkanals

2.2 E-Learning / Einflüsse der Technologieentwicklung auf das Bildungswesen

Selbstverständlich haben die technologischen Veränderungen der letzten Jahre auch den Bildungssektor nicht unbeeinflusst gelassen. Mit dem Einsatz von Computern und Lernsoftware wird im schulischen und universitären Bereich versucht, das Lernen zu digitalisieren und damit die Ausbildung zu aktualisieren und zu erweitern [25, 26]. Aspekte wie E-Learning und der Gebrauch des Internet als didaktisches Medium spielen heute eine immer größere Rolle.

In diesem Kapitel werden zunächst in Abschnitt 2.2.1 Begriffe und Grundlagen zum Thema E-Learning beschrieben, während anschließend in Abschnitt 2.2.2 kurz das Fernstudienprojekt *medin* vorgestellt wird, das die in dieser Arbeit zu entwickelnde Software miteinbinden soll.

2.2.1 Begriffsdefinitionen E-Learning

Der Begriff *E-Learning* steht für *electronic learning* und beschreibt grundsätzlich alle Formen von Lernen, die durch den Einsatz von Informations- und Kommunikationstechnologie unterstützt werden. Damit ist er ein Synonym für den Einfluss des technologischen Wandels durch die digitale Revolution auf das Bildungswesen. [27, 28]
Erste breit verfügbare E-Learning-Technologien entwickelten sich parallel zur Verbreitung von Großrechensystemen und Fernsehen in den 1960er Jahren. Über spezielle Fernsehkanäle wurden Bildungsinhalte zum *Telelernen* vermittelt. Ein Beispiel dafür ist das bis heute existierende Telekolleg des bayerischen Rundfunks (`http://www.br-online.de/wissen-bildung/telekolleg/`). Mit der Entwicklung des PCs kam im Bereich des E-Learning dann das *Computer Based Training (CBT)* auf. Dabei wird mit Lernmodulen auf CD-ROM eine räumlich und zeitlich flexible Nutzung möglich.
Im Laufe der Zeit wurden die E-Learning-Angebote durch die fortschreitende Digitalisierung der verschiedensten Medienarten multimedialer. Heute werden Videos, Fotos, Animationen, interaktive Elemente uvm. zur Erstellung von E-Learning-Materialien genutzt. Daneben entwickelte sich im Zuge des Internet-Hype das E-Learning hin zum webunterstützten Lernen, dem sogenannten *Web Based Training (WBT)*. Die zur Verfügung stehenden digitalen Breitbandnetze machten eine Verteilung multimedialer Lerneinheiten durch das Internet möglich. Auf Webseiten abgelegt und verlinkt kann der Lernstoff so jederzeit von jedem Ort abgerufen werden. Web-basierte Lerntools bilden den größten Teil der heutigen E-Learning-Systeme. [10, 29]
Durch die Einbettung ins Netz und die Entwicklungen des Web2.0 haben sich vielfältige weiterführende Möglichkeiten der Kommunikation und Interaktion des Lernenden mit seinen Mitlernern oder des Lernenden mit dem Lehrenden ergeben. Mittlerweile werden E-Mail, News, Chats, Materialarchive und Diskussionsforen mit dem WBT verknüpft.
Nahezu jede Fernuniversität bietet ihre Studiengänge heutzutage in einer Online-Version an. In Schulen oder Universitäten wird E-Learning dagegen zumeist als Zusatzmedium eingesetzt. Lernende können im Rahmen ihrer Kurse über *Lernplattformen*, auch *Learning Management Systems (LMS)* genannt (Näheres dazu siehe [30, 31]), auf Unterrichtsmaterialien zugreifen. Abbildung 2.5 zeigt als Beispiel die moodle-Plattform des Instituts für Medizinische Informatik mit verschiedenen Angeboten zu den Kursen. Diese Verschränkung von Präsenzlehre und E-Learning wird auch *Blended Learning* genannt. Die E-Learning-Materialien haben in diesem Lehr- und Lernprozess folgende Funktionen:

- Unterstützung und Ergänzung bei der Präsentation der Lehrinhalte

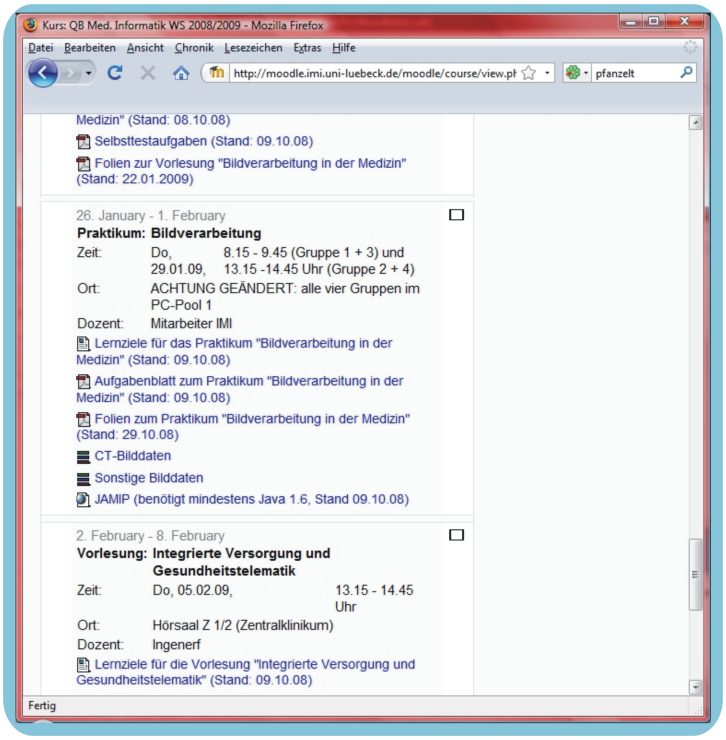


Abbildung 2.5: Die Lernplattform moodle des Instituts für Medizinische Informatik

- Vorbereitung des Unterrichts
- Erweiterung der Kommunikationsmöglichkeiten zwischen (entfernt) Lernenden und Lehrenden
- Einsatz zu Übungs-, Vertiefungs- und Testzwecken

2.2.2 Das Bmbf-Projekt medin

"Multimediales Fernstudium Medizinische Informatik", kurz *Medin* genannt, war ein von 2001 bis 2004 vom Bundesministerium für Bildung und Forschung (Bmbf) gefördertes Projekt im Rahmen des Programms *Neue Medien in der Bildung*. In Kooperation mit dem Institut für Multimediale und Interaktive Systeme und dem Institut für Medizinische Informatik wurde das Nebenfach Medizinische Informatik zum Informatikstudium der FernUniversität Hagen multimedial aufbereitet und online verfügbar gemacht.
Ausgehend von einer schriftlichen Fassung der Kurse wurden selbige in einem ersten Schritt in eine XML-Version konvertiert, die dann in HTML umgewandelt wurde. Um multimediale Lehr- und Lernmaterialen zu schaffen, wurden diese HTML-Seiten anschließend mit multimedialen Elementen wie Animation, Grafik, Video und Audio

angereichert.
Für die Kurseinheiten wurden neben einem einheitlichen Layout eine geeignete Navigationsstruktur, verschiedene Sichten auf die Kursmaterialien sowie die Integration und Kombination verschiedener Medien realisiert.

3 Grundlagen

In diesem Kapitel werden wichtige Grundlagen für das Verständnis und den Aufbau dieser Arbeit erläutert. Dabei wird in Abschnitt 3.1 zunächst auf das Thema *Medizinische Bildverarbeitung* eingegangen, in dessen Kontext das hier zu entwickelnde Lernsystem eingeordnet werden soll. Danach beschreibt Abschnitt 3.2 ausführlich das in der Medizin für digitale Bilddaten geltende Standardformat *DICOM*, das auch in dem hier zu entwickelnden System für die Anzeige medizinischer Bilddaten verwendet werden soll. Zuletzt werden in Abschnitt 3.3 verschiedene Verfahren zum verschlüsselten Datenversand vorgestellt, was für die Konzeption des Telekonferenzmoduls dieser Arbeit von Bedeutung ist.

3.1 Medizinische Bildverarbeitung

Medizinische Bilder sind heutzutage aus dem medizinischen Alltag nicht mehr wegzudenken. Sie sind zentrales Hilfsmittel der Diagnostik und Therapie in nahezu allen Bereichen, von der Vorsorgeuntersuchung über die Diagnosefindung und Therapie- oder Verlaufskontrolle bis hin zur computergestützten Vorbereitung und Durchführung von Operationen und strahlentherapeutischen Behandlungen. Die Erzeugung und Verarbeitung solcher Bilddaten mit modernsten Mitteln der Technik sind Teilbereiche des Gebiets der Medizinischen Bildverarbeitung, welches in diesem Abschnitt näher beschrieben wird.
Die Medizinische Bildverarbeitung lässt sich in fünf Bereiche unterteilen (siehe Abbildung 3.1) [32, 33]:

- Bilderzeugung
- Bildvorverarbeitung
- Bildanalyse
- Bildvisualisierung
- Bildverwaltung

Abbildung 3.1: Bereiche der medizinischen Bildverarbeitung

Dabei wird in der Stufe der *Bilderzeugung* das Bild zunächst mittels eines bildgebenden Verfahrens generiert und - sofern es sich nicht bereits um ein computerbasiertes Aufnahmeverfahren handelt - gegebenenfalls digitalisiert. Abschnitt 3.1.1 erläutert einige der wichtigsten Verfahren und Methoden. In den Bereich der *Bildvorverarbeitung* oder auch *Bildbearbeitung* oder *Bildverbesserung* (siehe Abschnitt 3.1.2) fallen dann jene Arbeitsschritte, die ein Bild hinsichtlich seiner späteren Auswertung verbessern, z. B. durch die Reduzierung des Rauscheinflusses. Anschließend folgt in der Kette der Bildverarbeitungsschritte und in Abschnitt 3.1.3 die *Bildanalyse* oder *Bildauswertung*. Hierunter versteht man Verfahren und Methoden, die zur Interpretation eines Bildes eingesetzt werden. Dabei soll ein bildbeschreibendes Ausgangsbild zu einem vorliegenden Originalbild generiert werden. Bei Röntgenbildern kann dies beispielsweise die hervorgehobene Darstellung und inhaltliche Zuordnung von Knochen oder Organen sein [33]. Die *Bildvisualisierung* benutzt dann verschiedene Methoden, um die Ergebnisse dieses Analyseprozesses darzustellen (Abschnitt 3.1.4). Als letzter Teilbereich der Medizinischen Bildverarbeitung wird schließlich in Abschnitt 3.1.5 die *Bildverwaltung* beschrieben; hierzu zählen alle digitalen Methoden und Techniken, die dem Zugriff, der Speicherung und der Übertragung medizinischer Bilder dienen. Das spezielle Format, in dem diese Daten vorliegen, wird in Abschnitt 3.2 gesondert betrachtet, da es für die Anzeige medizinischer Bilddaten eine besondere Herausforderung darstellt.

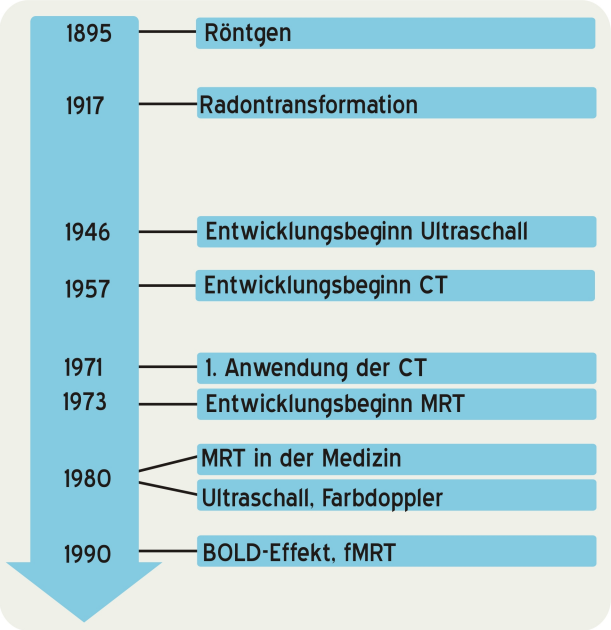


Abbildung 3.2: Zeitliche Entwicklung der wichtigsten bildgebenden Verfahren seit Röntgen

3.1.1 Bilderzeugung

Die Medizinische Bildverarbeitung ist geprägt von der Vielzahl unterschiedlicher Möglichkeiten zur Bilderzeugung und der damit zur Verfügung stehenden Vielzahl unterschiedlicher Bildarten. Auf die wichtigsten dieser medizinischen Bildtypen und die Art ihrer Generierung wird in den nachfolgenden Abschnitten kurz eingegangen, in Abbildung 3.2 ist die zeitliche Entwicklung dieser Verfahren zu sehen.
Wie so ein Bild digital repräsentiert wird, ist dann in Abschnitt 3.1.1.6 nachzulesen.

3.1.1.1 Röntgen

Mit der Entdeckung der *X-Strahlen* (englisch: X-rays), welche später in *Röntgenstrahlen* umbenannt wurden, beginnt am 8. November 1895 die Geschichte der bildgebenden Verfahren in der Medizin. Wilhelm Conrad Röntgen gelingt nicht nur die Entdeckung der Strahlen, sondern am 22. Dezember 1895 auch die erste Röntgenaufnahme von der Hand seiner Frau, bei der Knochen und Ehering klar zu erkennen sind (siehe Abbildung 3.3, Bild 1). Damit waren erstmals Abbildungen vom Inneren des Körpers möglich. Prinzip des Röntgenverfahrens sind elektromagnetische Wellen, die ausgehend von einem Röntgenstrahler den Körper durchleuchten und dabei je nach Durchdringungsstärke einen dahinter befindlichen Röntgenfilm schwärzen. Knochen absorbieren z. B. die

Röntgenstrahlung sehr stark und erscheinen daher auf Röntgenbildern als sehr helle Strukturen im Vergleich zu umgebendem Weichteilgewebe. Das entstehende Röntgenbild ist ein sogenanntes Projektionsbild, d. h. ein Bild, auf dem verschiedene Körperschichten überlagert zu sehen sind. Heutzutage ist das Verfahren zumeist digitalisiert, statt des Röntgenfilms wird eine wiederverwendbare Speicherfolie oder ein elektronischer Detektor benutzt.

3.1.1.2 Computertomographie (CT)

Bei der *Computertomograhie* handelt es sich um ein spezielles Röntgenverfahren, das überlagerungsfreie Querschnittsbilder verschiedener Körperregionen liefert. Dafür werden mit einer rotierenden Röntgenröhre aus den unterschiedlichsten Richtungen Röntgenbilder des zu untersuchenden Körpers oder Objektes erstellt, aus denen dann nachfolgend mit Hilfe der *Radontransformation* ein CT-Bild (Beispiel siehe Abbildung 3.3, 2. Bild v.l.) computergestützt berechnet wird. CT-Bilder liegen also von vornherein in digitaler Form vor und gaben damit einen wichtigen Anstoß für die weitere Entwicklung rechnergestützter Verfahren zur Bildverarbeitung. Obwohl Johann Radon die mathematische Grundlage für die Computertomographie mit seiner Radontransformation bereits 1917 lieferte, wurden erst 1971 die ersten CT-Aufnahmen gemacht. Maßgeblich verantwortlich für die Entwicklung der Computertomographie sind der Physiker Alan M. Cormack und der Elektrotechniker Godfrey Hounsfield. Nach letzterem ist auch die sogenannte *Hounsfield-Skala* benannt, eine Skala, die für den besseren Vergleich verschiedener CT-Bilder eingeführt wurde und dafür die gemessen Abschwächungskoeffzienten der Röntgenstrahlung in Bezug auf das Referenzmaterial Wasser normiert [33, 34]. Gleiche Gewebearten liegen so auf dieser Skala im selben Wertebereich, da sie die Strahlung gleich absorbieren. So ist es möglich pathologische Abweichungen von gesundem Gewebe festzustellen. Die Generierung von CT-Aufnahmen geht mit einer hohen Strahlenbelastung einher; im Vergleich zum Röntgen werden nicht nur eine sondern viele Röntgenaufnahmen gemacht, daher ist auch die Strahlenbelastung um ein Vielfaches höher.

3.1.1.3 Magnetresonanztomographie (MRT)

Eine weitere Methode, um überlagerungsfreie Schnittbilder einzelner Körperschichten zu erzeugen, stellt die Magnetresonanztomographie dar - eine Methode, bei der keinerlei Strahlenbelastung für den Patienten anfällt. Das Verfahren beruht auf einer synchronen Anregung oder auch Magnetisierung von Kernteilchen (meist Wasserstoff) und der anschließenden Zeitmessung, bis diese wieder in ihren Normalzustand übergehen. Eine

wesentliche Grundlage für den Bildkontrast sind die verschiedenen Relaxationszeiten der einzelnen Gewebearten sowie der unterschiedliche Gehalt an Wasserstoff-Atomen [35]. Im Gegensatz zu CT-Bildern zeichnen sich MRT-Bilder durch einen sehr hohen Weichteilkontrast aus, durch Variation von Untersuchungsparametern (Kontrastmittelgabe, Magnetfeldstärke, etc.) kann eine sehr hohe Detailerkennbarkeit im Organbereich erreicht werden, die Darstellung von Knochenstrukturen ist hingegen nur eingeschränkt möglich (siehe Abbildung 3.3, 3. Bild v.l.). Im Vergleich zu anderen bildgebenden Verfahren ist die Magnetresonanztomographie sehr zeitaufwendig und teuer. Entwickelt wird sie bereits seit den 1970er Jahren (Paul C. Lauterbur und Sir Peter Mansfield), eine breite medizinische Anwendung findet jedoch erst seit Mitte der 1980er Jahre statt. Die neueste Forschungsrichtung stellt die erst seit den 1990er Jahren entwickelte *funktionelle Magnetresonanztomographie (fMRT)* dar, mit der Stoffwechselvorgänge von Hirnarealen visualisiert werden können. Zunutze macht man sich dabei den sogenannten *BOLD-Effekt* (Blood Oxygen Level Dependency), also die Änderung der magnetischen Eigenschaften des Blutes in Relation zu seinem Sauerstoffgehalt.

3.1.1.4 Ultraschall - Sonographie

Wie der Name schon vermuten lässt, wird beim *Ultraschall-Verfahren* mit Ultraschallwellen gearbeitet. Diese auch *Sonographie* genannte Methode ist das häufigste bildgebende Verfahren in der Medizin - u. a. deshalb, weil es sehr kostengünstig ist und mit keinerlei Strahlenbelastung für den Patienten einhergeht. Das Grundprinzip geht auf militärische Anwendungen zurück (Sonarortung von U-Booten), jedoch werden in der medizinischen Diagnostik andere Schallfrequenzen benutzt. Ultraschallwellen sind Schallwellen oberhalb der menschlichen Hörgrenze mit einer Frequenz zwischen 20 kHz und 1 GHz, für die medizinische Diagnostik werden üblicherweise Frequenzen zwischen 1 und 40 MHz verwendet. Solche Ultraschallwellen werden von einem Schallkopf erzeugt und dann in das zu untersuchende Gewebe ausgesendet, wo sie sich innerhalb des Gewebes ausbreiten. An den Gewebegrenzen werden sie in Abhängigkeit von dem gewebespezifischen Schallwellenwiderstand unterschiedlich stark reflektiert, was als Signal wiederum von dem Schallkopf empfangen wird. Die Stärke dieser Reflexion wird von dem Ultraschallgerät als Grauwert interpretiert. Entwickelt wird das Verfahren bereits seit dem Ende der 1940er Jahre, die heute üblichen farbkodierten Doppler-Darstellungen sind jedoch erst seit den 1980er Jahren durch die Verfügbarkeit leistungsstarker Rechner möglich geworden.

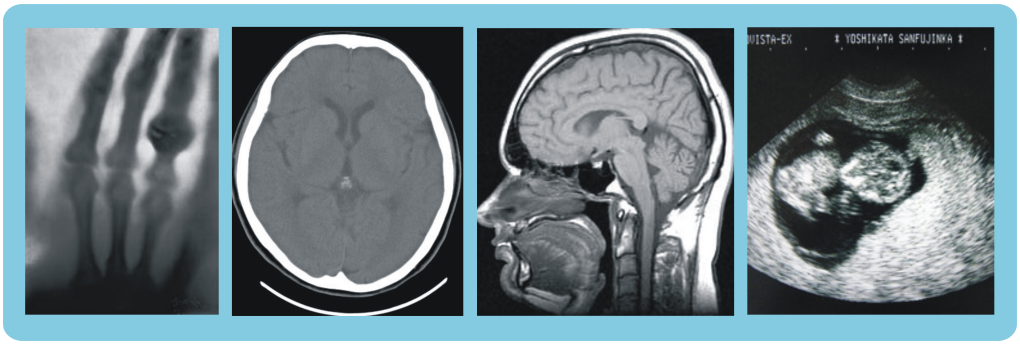


Abbildung 3.3: Bildbeispiele für die 4 wichtigsten bilderzeugenden Verfahren: Röntgen, CT, MRT, US (v.l.n.r)

3.1.1.5 Weitere bildgebende Verfahren

In der heutigen Medizin werden noch eine ganze Reihe weiterer Verfahren zur Erzeugung von Bildern eingesetzt. Während die zuvor erwähnten Methoden zu den *morphologisch bildgebenden Verfahren* zählen, also darauf abzielen, möglichst genaue Darstellungen von der Struktur des Körperinneren zu generieren, bilden die meisten anderen Verfahren oft einen Stoffwechselvorgang, eine Wärmeverteilung oder einen Verlauf ab. Es handelt sich somit meist um Methoden, mit denen eine Funktionsdiagnostik gemacht wird. Besonders hervorzuheben sind hier die nuklearmedizinischen Verfahren [36], bei denen radioaktiv markierte Stoffwechselprodukte in den Körper eingebracht werden, deren Strahlung dann von außerhalb mit einer speziellen Kamera aufgezeichnet wird. Auf diesen Bildern sieht man die Verteilung der radioaktiven Stoffwechselprodukte, wo und wie stark sie in Organen oder Knochen angereichert oder abgebaut wurden, und kann daraus Rückschlüsse auf die Funktionstüchtigkeit einzelner Organe ziehen. Zu den nuklearmedizinischen Verfahren zählen: *Positronen-Emissions-Tomographie* (PET), *Single-Photon-Emissions-Computertomographie* (SPECT) und *Szintigraphie*.
Folgende weitere Bildtypen sind in der Medizin üblich: *Fotografien*, *histologische Schnittbildserien*, Bilder der *optischen Kohärenztomographie* (OCT), *Thermographie-Bilder* und *Video*. Abbildung 3.4 zeigt eine Übersicht dieser Methoden mit typischen Beispielbildern angeordnet nach der Art des gemessenen Effekts.

Dieser Abschnitt hat einen Überblick über bildgebende Verfahren in der Medizin gegeben und die wichtigsten von ihnen kurz erläutert. Dabei ist deutlich geworden, dass es eine Vielzahl verschiedener Methoden zur Erzeugung medizinischer Bilddaten gibt und dass diese Methoden aufgrund der verschiedenen Herangehensweisen und Problemstellungen

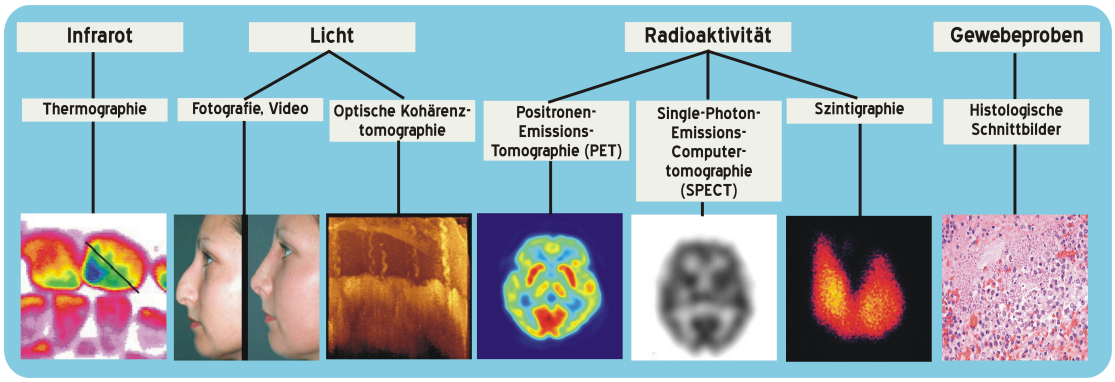


Abbildung 3.4: Übersicht über weitere bildgebende Verfahren in der Medizin

ganz unterschiedliche Bilddaten liefern, die sich auch in der Art ihrer Darstellung unterscheiden. Liefert das Röntgenverfahren anatomische Details vom Inneren des Körpers, so werden nuklearmedizinische Methoden wie die Szintigraphie oder die Positronen-Emissions-Tomographie hingegen angewendet, um Stoffwechselvorgänge zu beurteilen, also um eine Funktionsdiagnostik zu machen. Eine besonders detailgetreue Abbildung der Morphologie ist bei diesen Bildern nicht zu erwarten. Bei der weiteren Anwendung bildverarbeitender Verfahren dürfen daher das zugrundeliegende Bildmaterial und die zugehörige Aufgabenstellung natürlich nicht außer Acht gelassen werden.

3.1.1.6 Repräsentation digitaler Bilddaten

Ein digitales Bild lässt sich mathematisch als zweidimensionale Matrix mit x Zeilen und y Spalten beschreiben. Das Bild F entspricht dann der Funktion $F = f(x,y)$. Der einzelne Bildpunkt wird als *Pixel* (von engl. picture element) bezeichnet und gibt den *Grauwert* oder *Signalwert* des Bildes an des Stelle (x,y) an. Handelt es sich um Farbbilder, so kann ein Bild auch aus drei Matrizen bestehen, die jeweils den Rot-, Grün- und Blauanteil abdecken. In der Regel haben Bildpunkte eine quadratische Form, im besten Fall eine rechteckige, es gibt aber auch andere Formen, z. B. hexagonale Raster. Für die Qualität eines digitalen Bildes sind die *Ortsaufösung* oder *Rasterung* und die *Intensitätsauflösung* oder *Grauwertquantisierung* von entscheidender Bedeutung. Ist die Rasterung zu grob, so gehen möglicherweise Details verloren, ist die Rasterung sehr fein, so erhöht sich der Speicher- und Verarbeitungsaufwand. Ähnlich ist es bei der Intensitätsauflösung, bei zu wenigen zur Verfügung stehenden Grauwerten entstehen evtl. Artefakte oder ein Detailverlust. Abbildung 3.5 verdeutlicht das Problem. Tabelle 3.1 zeigt typische Grauwertquantisierungsstufen bei digitalen Bildern.

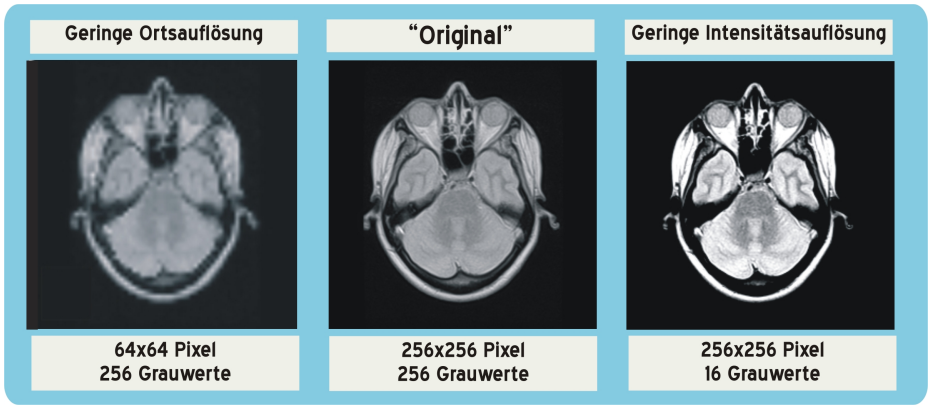


Abbildung 3.5: Die Bildqualität eines digitalen Bildes ist abhängig von der Orts- und Intensitätsauflösung

Grauwert-tiefe (Bit)	Anzahl Werte	Beschreibung
1	2	Binärbild
5	32	ungefähre maximale Anzahl an Grauwerten, die das menschliche Auge unterscheiden kann
8	256	Anzahl an Grauwerten, die ein handelsüblicher Monitor anzeigen kann
16	65536	üblich für medizinische Bilddaten, allerdings werden oft nur 12 Bit, also 4096 Grauwerte, verwendet
24	16777216	üblich für Farbbilder, z. B. 3*8 Bit in RGB-Bildern

Tabelle 3.1: Grauwertquantisierungsstufen

3.1.2 Bildvorverarbeitung

Unter Bildvorverarbeitung versteht man eine oder eine Folge von Bildverarbeitungsoperationen, die den Informationsgehalt eines Bildes nicht wesentlich verändern aber hinsichtlich einer anschließenden Auswertung verbessern. Ziel ist also eine Steigerung der Bildqualität. Dazu gehören Bildverarbeitungsoperationen, die - automatisch oder manuell ausgeführt - ohne Wissen über den konkreten Inhalt der einzelnen Bilder vorgenommen werden können. Dabei können z. B. fehlerhafte oder ungenaue Bilder so korrigiert bzw. restauriert werden, dass wichtige Informationen oder Strukturen sichtbar gemacht oder hervorgehoben werden. Natürlich ist auch die Bildvorverarbeitung stark von der Problemstellung und dem zu analysierenden Bildmaterial abhängig. Zu den meist ausgeführten Anwendungen in der Bildvorverarbeitung zählen die Reduzierung des Rauscheinflusses, die Elimination von Artefakten und Störeinflüssen, die Verbesserung des Bildkontrastes oder der Helligkeit, die Normierung der Bildgröße sowie eine Graustufenmodifikation.

Grundsätzlich wird bei der Bildvorverarbeitung zwischen 3 verschiedenen Arten von Operationen unterschieden: punktorientierten, lokalen und globalen. Wie der Name schon vermuten lässt, handelt es sich bei *punktorientierten Operationen* um Verfahren, die einzelne Bildpunkte isoliert von ihrer Umgebung verändern, jeder Bildpunkt wird hier also einzeln und nur in Abhängigkeit von seinem Grauwert und gegebenenfalls von seiner Position im Bild transformiert. Bei *lokalen Operationen* werden hingegen die benachbarten Bildpunkte zur Berechnung eines neuen Pixelwertes miteinbezogen und bei *globalen Operationen* werden schließlich für die Berechnung eines neuen Pixels alle übrigen Pixel des Ausgangsbildes benötigt.

3.1.3 Bildanalyse

Zu den wichtigsten Komponenten in medizinischen Bildsystemen gehören die verschiedenen Techniken zur Bildauswertung und Bildanalyse. Sie spielen eine entscheidungsunterstützende Rolle bei der computerbasierten Bewertung und Interpretation medizinischer Bilddaten. Ziel ist es, ein bildbeschreibendes Ergebnis zu einem vorliegenden Originalbild zu erhalten. Dafür müssen einzelne Objekte extrahiert und quantitativ beschrieben werden. Abbildung 3.6 zeigt ein Beispiel für die Phasen eines Bildanalyse-Systems anhand einer Hauttumor-Klassifizierung. Die Bildanalyse geschieht folglich in mehreren Teilschritten, den ersten bildet zumeist die *Segmentierung* des interessierenden Bildobjektes. Sie dient der Abgrenzung verschiedener, diagnostisch oder therapeutisch relevanter Bildobjekte wie Gewebe, Tumore, Läsionen, Gefäßsysteme etc. und ist damit Grundlage für eine weitergehende Analyse, Vermessung und 3D-Visualisierung von medizinischen Bildobjekten. Das Ziel der Segmentierung ist es, ein Objekt oder eine Struktur innerhalb eines Bildes hervorzuheben. Um dies zu erreichen, existieren eine Vielzahl von Algorithmen, die auf verschiedenen Ansätzen basieren. Allerdings ist der Einsatz von Algorithmen zur vollständigen, automatischen Segmentierung medizinischer Bilddaten nur eingeschränkt möglich und auf spezielle Bilddaten und Problemstellungen beschränkt, u. a. ist er abhängig von den verwendeten Modalitäten, den Aufnahmeparametern (z. B. Rauschen, Auflösung, Konstrastmittelgabe) oder den zu findenden Objekten. Für die Qualität eines Segmentierungsverfahrens sind die Robustheit gegenüber Rauschen, die Reproduzierbarkeit, die Geschwindigkeit und die Genauigkeit im Vergleich mit der Handsegmentierung eines Experten entscheidend. [34]
Auf Grundlage eines Segmentierungsergebnisses geschieht dann eine weitergehende Analyse. Dazu zählen standardmäßige Funktionen wie die Abstandsmessung zwischen zwei vorgegebenen Punkten oder die Berechnung eines mittleren Signalwertes in einer

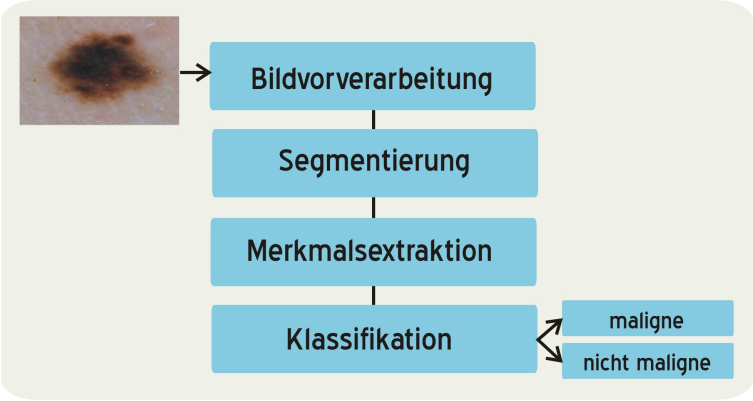


Abbildung 3.6: Phasen eines Bildanalyse-Systems am Beispiel der Klassifikation eines Hauttumors

gegebenen Region-Of-Interest. So sind bereits einfache numerische Beschreibungen möglich. Ein komplizierteres Verfahren zur quantitativen Beschreibung verschiedener Objekteigenschaften stellt z. B. die *Texturanalyse* dar. Unter diesem Begriff wird die Untersuchung von Texturen, also von Struktureigenschaften eines Bildbereichs, verstanden mit dem Ziel der Texturklassifikation, Textursegmentierung oder Textursynthese. Typisches Werkzeug der Texturanalyse ist die *Cooccurrence-Matrix*, die die auftretenden Häufigkeiten von Grauwertübergängen erfasst und anhand dieser Merkmalswerte für den Kontrast oder die Entropie berechnet. [37]
Werden die gefundenen Segmente und Bildmerkmale mit algorithmischen Verfahren bestimmten Klassen zugeordnet, erhalten sie also eine inhaltliche Bedeutung, so spricht man von *Klassifikation* und *Mustererkennung*.

3.1.4 Bildvisualisierung

In den Bereich der Bildvisualisierung fallen alle Arbeitsschritte, die sich mit der Anzeige und Darstellung der Bilder beschäftigen. Dazu gehören Farbraumtransformationen ebenso wie dreidimensionale Visualisierungstechniken, die oft genutzt werden, um Ergebnisse aus der Bildanalyse wie beispielsweise die Segmentierung eines Tumors aus CT-Schichtaufnahmen darzustellen. Dafür werden Techniken aus dem Bereich der Computergrafik wie z. B. die Oberflächenrekonstruktion, die Schattierung oder verschiedene Beleuchtungsmodelle eingesetzt.
Eine einfache Bildvisualisierungstechnik stellt die routinemäßig in der Radiologie eingesetzte *Fensterungsmethode* dar, mit deren Hilfe CT- und MR-Bilder in ihrer gesamten Grauwerttiefe betrachtet werden können.

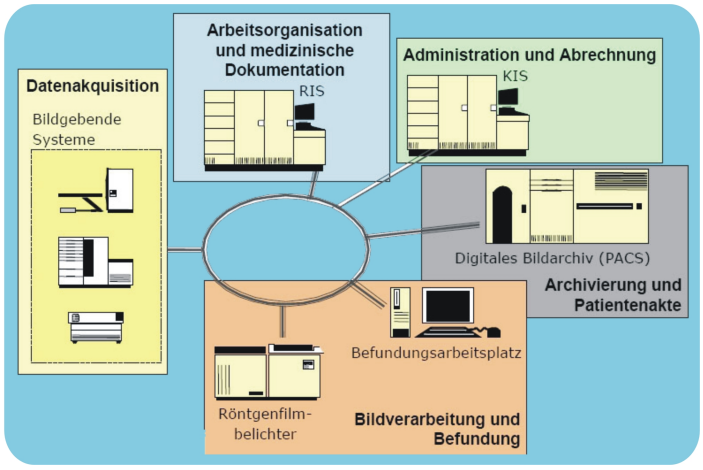


Abbildung 3.7: Bildverwaltung im Krankenhaus

3.1.5 Bildverwaltung

Zu dem Bereich der Bildverwaltung zählen alle digitalen Methoden und Techniken, die dem Zugriff, der Speicherung und der Übertragung von medizinischen Bilddaten dienen. Zentrales System ist hier zumeist das sogenannte *Picture Archiving and Communication System (PACS)*, ein Bildarchivierungs- und Kommunikationssystem auf der Basis digitaler Rechner und Netzwerke.

Nach ihrer Erzeugung werden digitale Bilddaten von PAC-Systemen erfasst und zusammen mit Informationen über die Identität des jeweiligen Patienten, über die klinische Fragestellung und über die Untersuchung auf einem zentralen Serversystem gespeichert. Wie Abbildung 3.7 verdeutlicht, ist ein PAC-System immer auch an die anderen Informations- und Kommunikationssysteme eines Krankenhauses wie z. B. Krankenhausinformationssystem (KIS) oder Radiologie-Informationssystem (RIS) gekoppelt. Bei allen modernen Systemen werden standardisierte Kommunikationsprotokolle und Speicherformate verwendet, die der nachfolgende Abschnitt 3.2 näher erläutert. So lassen sich die verschiedenen PAC-Komponenten und Diagnosegeräte herstellerunabhängig miteinander verbinden. Die Bilder können so von allen an das PACS angeschlossenen Bildbearbeitungs- und Bildbetrachtungsplätzen unabhängig von Zeit und Ort angefordert und zur Befundung, Diagnosestellung und für Forschung und Lehre auf Bildmonitoren dargestellt werden.

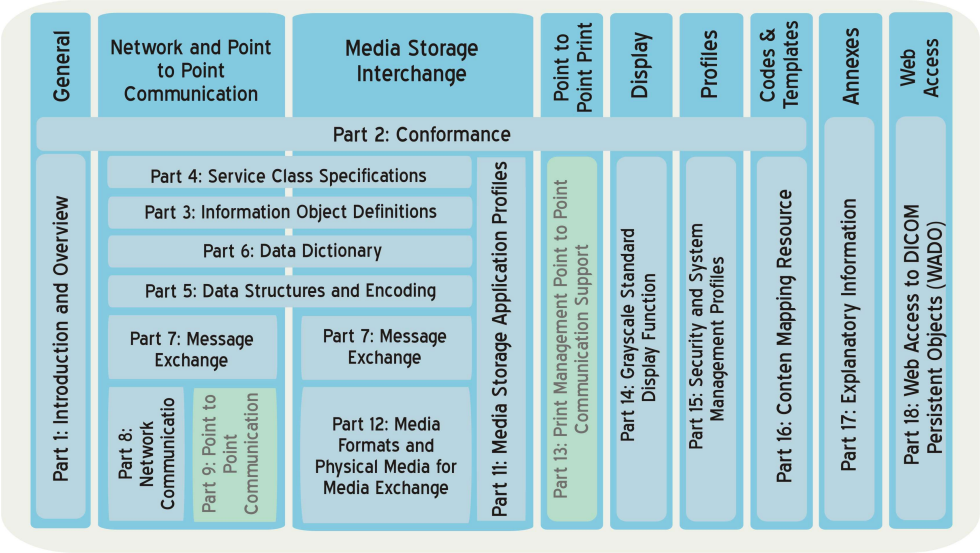


Abbildung 3.8: Der DICOM-Standard im Überblick

3.2 DICOM

DICOM steht für ***D**igital **I**maging and **Co**mmunications in **M**edicine* und ist ein weltweit offener Standard zum Austausch von digitalen Bildern in der Medizin. Dabei wird sowohl das Format zur Speicherung von Bilddaten, als auch das Kommunikationsprotokoll zum Austausch dieser Bilder standardisiert. DICOM hat sich im Laufe seiner Entwicklung zu einem unentbehrlichen Baustein für die Integration digitaler bildgebender und bildverarbeitender Systeme in der Medizin entwickelt. [38, 39]

Parallel zum zunehmenden Einsatz digitaler bildgebender Verfahren in der Medizin seit Mitte der 1970er Jahre (siehe dazu Abschnitt 3.1.1) entstand das Bedürfnis, digitale Bilder auch digital verwalten zu können. Da jeder Hersteller zunächst sein eigenes proprietäres Format entwickelte, war jedoch ein Austausch digitaler Bilder zwischen Geräten verschiedener Hersteller zunächst nicht möglich. Daher gründeten das American College of Radiology (ACR) und die National Electrical Manufacturers Association (NEMA) im Jahr 1983 eine gemeinsame Arbeitsgruppe zur Entwicklung eines Standardformats für die herstellerunabhängige Übertragung medizinischer Bilddaten. Dieser *ACR-NEMA* genannte Standard wurde 1985 in der Version 1.0 und 1988 in der weiterentwickelten Version 2.0 verabschiedet. Da er sich jedoch im Wesentlichen auf die Spezifizierung der Methoden zur Speicherung und Visualisierung medizinischer Bilddaten beschränkte und keine Netzwerkunterstützung bot, fand er nie die gewünschte Verbreitung. Seit der Verabschiedung der Version 3.0 im Jahre 1993 firmiert der Standard unter dem Namen

DICOM und erfüllt auch die Anforderungen für einen offenen Kommunikationsaustausch über TCP/IP-basierte Netzwerke. In Europa ist DICOM seit 1995 als formaler Standard akzeptiert (MEDICOM, ENV 12052). Der Inhalt des DICOM-Standards geht weit über die Definition eines reinen Austauschformats für medizinische Bilddaten hinaus, er beschreibt:

- Datenstrukturen für medizinische Bilder und bildbezogene Zusatzinformationen,
- netzwerkorientierte Dienste wie die Bildübertragung, die Abfrage eines Bildarchivs (PACS), die Druckausgabe, die Integration "RIS - PACS - bildgebende Modalität",
- Formate für den Datenträgeraustausch und
- Anforderungen an DICOM-konforme Geräte und Programme.

Auch wenn sich die Versionsnummer seit der Erstausgabe des DICOM3.0-Standards nicht mehr geändert hat, so wird der Standard doch fortwährend weiterentwickelt; derzeit existieren 26 Arbeitsgruppen (Stand: Juni 2008), die sich aus Mitarbeitern von Medizintechnik-Herstellern, Kliniken, Universitäten und anderen Forschungseinrichtungen zusammensetzen und DICOM um verschiedene Teilbereiche erweitern, wobei der Rahmen zusehends größer gefasst wird (z. B. Biosignale, Nuklearmedizin, 3D, Datenkompression, Structured Reporting etc.). Einen Überblick über die verschiedenen vom DICOM-Standard abgedeckten Bereiche liefert Abbildung 3.8.
Diese Ausführungen beziehen sich auf den DICOM3-Standard von 2008. Eine Kopie des Standards steht unter `http://medical.nema.org/dicom/2008` zur Verfügung.

3.2.1 Das DICOM-Dateiformat

Das sogenannte *DICOM File Format* (siehe [40] Part 10 und Part 6) beschreibt das Format einer einzelnen DICOM-Bilddatei. Wie Abbildung 3.9 zeigt besteht so ein einzelnes DICOM-File aus den Bilddaten selber und einer Liste von bildbegleitenden Informationen, dem sogenannten *Header*. Bei diesen bildbegleitenden Informationen handelt es sich um Angaben zum Patienten (z. B. Name, Geburtsdatum, Identifikationsnummer), zur bildgebenden Modalität und zur gemachten Aufnahme (z. B. Geräteangaben, Kalibrierung, Kontrastmittelgabe) und um Angaben zum eigentlichen Bild selber (z. B. Auflösung, Anzahl der Zeilen und Spalten, Anzahl der benutzten Bits).
Ein DICOM-File ist ein binäres Dateiformat, welches als Datenstrom (*Data Set*) bestehend aus einzelnen Datenelementen (*Data Elements*) definiert wird. Einem jeden Data Set ist die *DICOM File Meta Information* vorangestellt. Dabei handelt es sich um eine 128 Byte

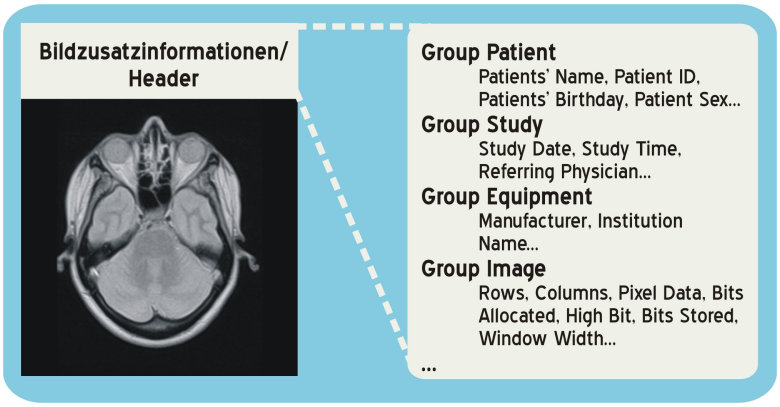


Abbildung 3.9: Aufbau eines DICOM-Bildes

große *DICOM File Preamble* für die Speicherung anwendungsspezifischer Informationen, gefolgt von dem 4 Byte langen ASCII-String "DICM" als DICOM-Präfix zur Erkennung eines DICOM-Files (DICOM-Dateien besitzen laut Spezifikation keine Dateierweiterung, auch wenn sich in der Praxis mittlerweile Endungen wie beispielsweise .img, .ima oder .dcm etabliert haben) und den *File Meta Elements*, die Informationen zum Auslesen des eigentlichen Data Sets beinhalten. Wie das Data Set sind auch diese File Meta Elements in Data Elements unterteilt (siehe Abbildung 3.10); hier wird u. a. die *Transfersyntax* zum Auslesen der Data Elements des Data Sets kodiert.

3.2.1.1 Das DICOM Data Element

Zentrales Informationsobjekt einer DICOM-Datei ist das Data Element. Alle im DICOM-Standard registrierten Datenelemente sind im sogenannten *Data Dictionary* abgelegt (siehe Part 6 des DICOM-Standards, Tabelle 3.2 zeigt einen Ausschnitt aus dem Data Dictionary). Ein einzelnes Data Element besteht wie in Abbildung 3.11 zu sehen ist aus drei bis vier Feldern:

- **Tag:** Der sogenannte Attribut-Tag ist ein eindeutiger Bezeichner für ein Informati-

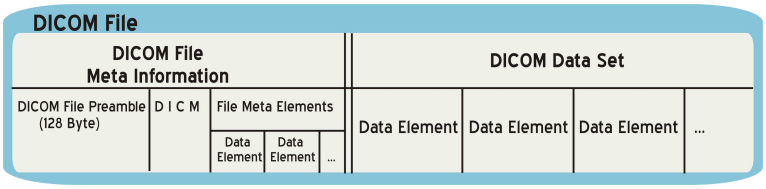


Abbildung 3.10: Der Aufbau eines DICOM-Files mit den vorangestellten DICOM Meta Informations zum Auslesen des DICOM Data Set

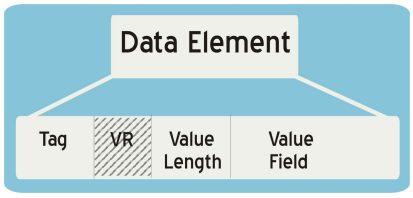


Abbildung 3.11: Der Aufbau eines DICOM Data Elements

Tag	Name	VR	VM
(0010,0010)	Patient's Name	PN	1
(0010,0020)	Patient ID	LO	1
(0010,0030)	Patient's Birth Date	DA	1
(0010,0040)	Patient's Sex	CS	1
(0010,1010)	Patient's Age	AS	1
(0010,1020)	Patient's Size	DS	1
(0010,1030)	Patient's Weight	DS	1
(0010,1040)	Patient's Address	LO	1

Tabelle 3.2: Beispiele aus dem Data Dictionary

onsobjekt. Er besteht aus einer Gruppen-Nummer und einer Element-Nummer, die jeweils 2 Byte lang und als hexadezimale Zahlen angegeben sind.

- **VR (optional):** Die Value Representation eines Data Elements beschreibt das Format und den Typ der enthaltenen Daten. Sie ist optional, die Transfersyntax gibt an, ob die VR mit einer Länge von 2 Byte angegeben ist oder nicht.
- **Value Length:** Hier wird die Länge des folgenden Datenfeldes angegeben. Ob dieses Feld eine Länge von 2 oder 4 Byte hat, ist abhängig vom Vorhandensein eines VR-Feldes bzw. wird durch die Transfersyntax bestimmt.
- **Value Field:** Hier sind die Daten des Datenelements enthalten.

Die Transfersyntax bestimmt wie die Data Elements auf Byte-Ebene kodiert werden und ist damit für die Speicherung und Übertragung von DICOM-Files von Interesse. Wie Abbildung 3.12 zeigt, gibt die Transfersyntax an, ob die in der Datei enthaltenen Data Elements ein VR-Feld besitzen (*Explicit VR*) oder ob das auszulesende Programm die VR aus dem Tag des Data Elements ermitteln muss (*Implicit VR*). Außerdem übermittelt die Transfersyntax die *Endianness* der Daten, also die Reihenfolge, in der ein in mehrere Bytes aufgeteiltes Datenwort gespeichert wird. Man unterscheidet hier zwischen *Little Endian* oder auch Little-Endian-First-Byte-Order (wörtlich "Kleines Ende zuerst"), wo das Byte mit den niederwertigsten Bits an der kleinsten Speicheradresse gespeichert

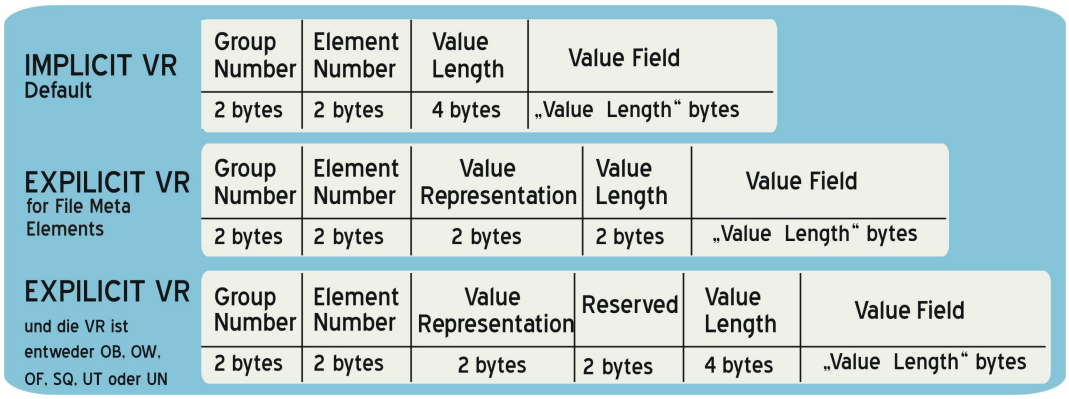


Abbildung 3.12: Der interne Aufbau eines DICOM Data Elements in Abhängigkeit von der VR

Transfersyntax UID	Beschreibung
1.2.840.10008.1.2	Implicit VR Little Endian: Default Transfersyntax für DICOM
1.2.840.10008.1.2.1	Explicit VR Little Endian: Transfersyntax für die File Meta Elements
1.2.840.10008.1.2.2	Explicit VR Big Endian
1.2.840.10008.1.2.4.50	JPEG Baseline (Process 1): Default Transfersyntax für Lossy JPEG 8 Bit Image Compression

Tabelle 3.3: Beispiele für die Transfersyntax in DICOM

wird, und *Big Endian* oder auch Big-Endian-First-Byte-Order (wörtlich “Großes Ende zuerst”), die das Byte mit den höchstwertigsten Bits zuerst speichert. Weiterhin kann die Transfersyntax Angaben zur Art der Bilddaten enthalten, d. h. das für die Kodierung der Bilddaten verwendete Format angeben (z. B. jpg). Festgelegt wird die Transfersyntax durch eine eindeutige UID, Tabelle 3.3 zeigt einige Beispiele. Die File Meta Elements werden standardmäßig immer mit der *“Explicit VR Little Endian”* kodiert. Ist in den File Meta Elements keine Angabe über die Transfersyntax zum Auslesen des Data Sets, so wird defaultmäßig die *“Implicit VR Little Endian”* benutzt. [41]

3.2.1.2 DICOM-Pixeldatenformat

Auf das Tag *“Pixel Data” (hex 7FE0, hex 0010)* folgen im Data Set die eigentlichen Pixeldaten. Jedes Pixel eines DICOM-Bildes ist in einer sogenannten *Pixelzelle* enthalten. Über das Format dieser Zelle geben drei Data Elements Auskunft:

- ***Bits Allocated*** (Tag (hex 0028, hex 0100)) legt die Größe eines Pixels fest.

- ***Bits Stored*** (Tag (hex 0028, 0101)) gibt die Anzahl der zur Darstellung wirklich notwendigen Pixel an.
- ***High Bit*** (Tag (hex 0028, 0102)) beschreibt, wo in Bezug auf Bits Allocated das höchstwertige Bit des Pixels anzuordnen ist.

Abbildung 3.13 zeigt ein Beispiel für so eine Pixelzelle. Zur Speicherung eines Pixels sind dabei 16 Bit angegeben (Bits Allocated = 16), zur Darstellung des Bildes werden aber nur 12 Bit genutzt (Bits Stored = 12) und das höchstwertige Bit liegt bei 11 (High Bit = 11). Die Tags *"Reihen" (hex 0028, hex 0010)* und *"Spalten" (hex 0028, hex 0011)*

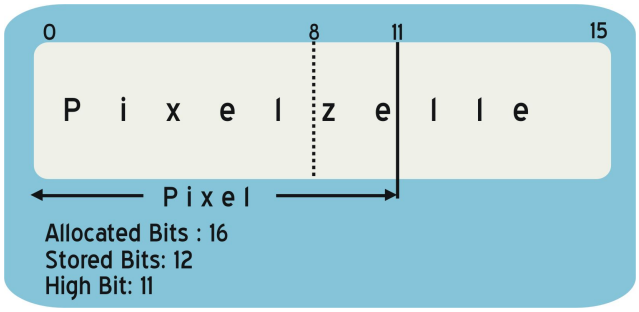


Abbildung 3.13: Beispiel für eine Pixelzelle in DICOM

geben Auskunft über die Größe der 2D-Pixelmatrix, die Bildebene beginnt immer mit dem ersten Pixel links oben, verläuft von rechts nach links und von oben nach unten und endet mit dem letzten Pixel rechts unten.

3.3 Verschlüsselung

Die *Kryptografie* beschäftigt sich mit mathematischen Verfahren zur Geheimhaltung von Nachrichten. Heutzutage sind kryptografische Verfahren im Bereich der elektronischen Kommunikation weit verbreitet. Hauptanwendungsgebiet ist dabei die Geheimhaltung von Nachrichten während des Transports, kryptografische Verfahren können aber auch für andere Anwendungen wie beispielsweise zur zweifelsfreien gegenseitigen Erkennung (*Authentifizierung*) oder für den Beweis der Unversehrtheit einer Nachricht (*Datenintegrität*) benutzt werden. Basisfunktionen der Kryptografie sind sogenannte *Chiffrierverfahren*, mathematische Algorithmen, die unter Verwendung eines *Schlüssels* aus einem Klartext einen Geheimtext erstellen. Der Schlüssel selbst ist im allgemeinen eine Bytesequenz, also eine Zeichenfolge bestimmter Länge. Benutzt man denselben Algorithmus aber verschiedene Schlüssel, um einen Text zu chiffrieren, so erhält man auch unterschiedliche

Geheimtexte. Bei einem guten Chiffrierverfahren ist es nicht möglich, den Klartext ohne Kenntnis des zugehörigen Schlüssels wiederherzustellen.
Kryptografische Verfahren sind unterschiedlich sicher, man spricht auch von *starker* und *schwacher* Kryptografie. Die Sicherheit eines solchen Verfahrens misst sich in der Rechenleistung und der Zeit, die erforderlich wäre, aus einem verschlüsselten Datensatz das Original ohne Kenntnis des zugehörigen Schlüssels wiederherzustellen. Da bisher noch kein Existenzbeweis für ein absolut sicheres Verfahren erbracht werden konnte, ist jedes System voraussichtlich nur auf begrenzte Zeit anwendbar; mehr Rechenleistung und neue Analyseverfahren können heute noch als sicher geltende Methoden schnell unsicher und damit unbrauchbar für den Einsatz machen.

3.3.1 Symmetrische Verschlüsselung

Unter *symmetrischen Verschlüsselungsverfahren* oder auch *Private Key Encryption* werden Chiffrierverfahren verstanden, bei denen für die Ver- und Entschlüsselung eines Datensatzes ein identischer Schlüssel benutzt wird. Der Sender A chiffriert seinen Text mit demselben Schlüssel, mit dem der Empfänger B ihn nach Erhalt wieder entschlüsselt. Dafür müssen beide Kommunikationspartner nicht nur den chiffrierten Text austauschen, sondern auch den zu benutzenden geheimen Schlüssel. Da jeder, der in den Besitz dieses Schlüssels gelangt, in der Lage ist, den von Sender A chiffrierten Text zu entschlüsseln, sollte der Austausch des Schlüssels nicht zur selben Zeit und auch nicht über denselben Datenkanal wie der des Chiffretextes stattfinden, womit sich die Schlüsselverteilung und -verwaltung als generelles Problem der symmetrischen Verfahren erweist. Von Vorteil sind die einfache Implementierbarkeit und Schnelligkeit. Der *Advanced Encryption Standard (AES)*, der *Data Encryption Standard (DES)* oder der *International Data Encryption Standard (IDEA)* sind Beispiele für symmetrische Verfahren. [42, 43]

3.3.2 Asymmetrische Verschlüsselung

Bei *asymmetrischen Verschlüsselungsverfahren* oder auch *Public Key Encryption* werden für die Ver- und Entschlüsselung zwei verschiedene Schlüssel benutzt. Jeder potentielle Kommunikationsteilnehmer ist dabei im Besitz eines zusammengehörigen Schlüsselpaares, bestehend aus zwei verschiedenen Schlüsseln, einem öffentlichen, dem sogenannten *Public Key*, mit dem der Klartext chiffriert wird, und einem geheimen, dem *Private Key*, der für die Dechiffrierung verwendet wird. Möchte Sender A eine geheime Nachricht an Empfänger B verschicken, die nur von diesem gelesen werden kann, so benötigt er für die Verschlüsselung der Daten den Public Key von B. Eine Dechiffrierung des so

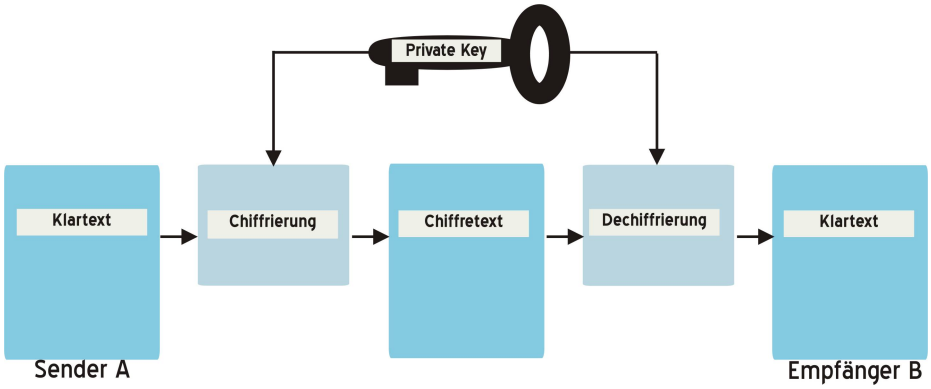


Abbildung 3.14: Symmetrische Verschlüsselung

entstandenen Geheimtextes ist nur mit Hilfe des Private Keys des Empfängers B möglich. So ist jeder, der Kenntnis des öffentlichen Schlüssels von B hat, in der Lage, eben diesem eine verschlüsselte Nachricht zukommen zu lassen, die dann nur von B entschlüsselt werden kann.

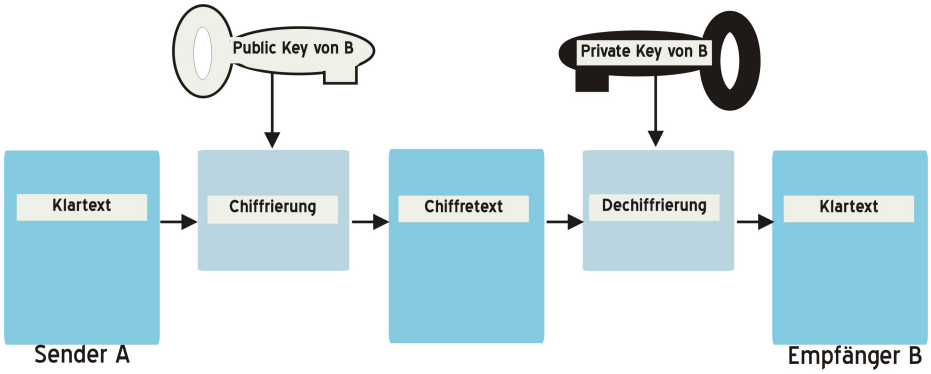


Abbildung 3.15: Asymmetrische Verschlüsselung

Das bei den symmetrischen Verfahren auftretende Problem der Schlüsseldistribution entfällt, da in die eigentliche Kommunikation nur öffentliche Schlüssel einbezogen werden. Sender und Empfänger müssen keinen geheim zu haltenden Schlüssel mehr über eine eventuell unsichere Verbindung austauschen, jeder Teilnehmer muss lediglich für die sichere Verwahrung seines Private Keys sorgen.
Schlüsselpaare können von einer sogenannten Schlüsselinstanz (Trusted Third Party) generiert und vergeben werden; der Private Key muss dabei von dem jeweiligen Schlüsselinhaber geheim gehalten werden, während der Public Key von der Schlüsselinstanz veröffentlicht wird. Das Konzept der asymmetrischen Verfahren wurde erstmals im Jahre 1975 von Diffie und Hellmann vorgestellt [44]. Das zugrundeliegende Prinzip der

asymmetrischen Chiffrieralgorithmen ist in der Regel ein methodisch ungelöstes mathematisches Problem, wie zum Beispiel die Faktorisierung, also die Zerlegung eines Produktes in seine Faktoren. Das Hauptproblem der asymmetrischen Verfahren bezüglich ihres praktischen Einsatzes liegt in ihrer Langsamkeit, sie sind im allgemeinen sehr viel langsamer als die symmetrischen Verfahren. Als das Standardverfahren gilt das 1977 von Rivest, Shamir und Adleman [45] entwickelte *RSA-Verfahren*.

3.3.3 Hybride Systeme

Wie in den vorangehenden Abschnitten bereits erwähnt, lassen sich symmetrische Verfahren besser in computerisierte Anwendungen integrieren und sind weitaus schneller als asymmetrische Verfahren, dafür besteht bei ihnen aber andererseits das Problem der Schlüsseldistribution und -verwaltung, was wiederum bei den asymmetrischen Verfahren nicht auftritt, die zudem auch sehr viel sicherer sind. In der Praxis führt dies dazu, dass zumeist sogenannte *hybride Systeme* eingesetzt werden; Systeme also, in denen sowohl symmetrische als auch asymmetrische Verfahren für die Verschlüsselung bzw. Datenübertragung verwendet werden.

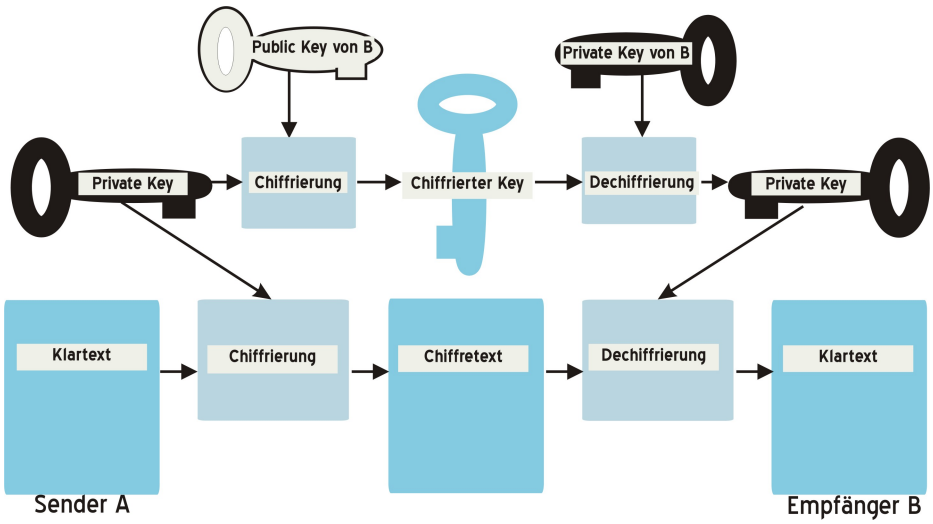


Abbildung 3.16: Hybride Verschlüsselung

Der grundsätzliche Ablauf stellt sich im allgemeinen folgendermaßen dar: Zu Beginn generiert der Sender A einen geheimen Schlüssel, den *Session Key*, mit dem die Daten nach einem symmetrischen Algorithmus verschlüsselt werden. Der Session Key selbst wird dann mit einem asymmetrischen Verfahren unter Nutzung des Public Keys des Empfängers B chiffriert und anschließend zusammen mit dem verschlüsselten Datensatz

übertragen. Nach Erhalt dechiffriert der Empfänger zunächst mit seinem Private Key und dem asymmetrischen Algorithmus den Session Key, mit dem er anschließend in der Lage ist, die eigentliche Nachricht zu entschlüsseln. Als Beispiel für ein solches hybrides System sei hier *Pretty Good Privacy (PGP)* erwähnt, ein System, das auch im Internet weite Verbreitung gefunden hat. Als symmetrischer Chiffrieralgorithmus wird dabei IDEA angewandt und als asymmetrischer das RSA-Verfahren. [46, 20, 21]

4 Existierende Ansätze

Die rasante technische Entwicklung der letzten Jahrzehnte hat gerade auch im Bereich der Medizinischen Bildverarbeitung für viele Neuerungen gesorgt. Heutzutage ist die digitale Bildverwaltung, -bearbeitung und -archivierung Standard und aus Krankenhäusern und Arztpraxen nicht mehr wegzudenken. Zahlreiche Hersteller offerieren Programme für die Bearbeitung medizinischer Bilder am PC oder bieten Bildarchivierungs- und Bildverwaltungssysteme an, teilweise mit integrierten Tools für telemedizinische Anwendungen. Daneben existieren verschiedene Angebote für die Lehre.

Da in dieser Arbeit ein Programm mit zwei verschiedenen Anwendungsschwerpunkten entwickelt werden soll - zum einen für den Einsatz als Lernsystem zur Vermittlung von Grundlagenalgorithmen der Medizinischen Bildverarbeitung und zum anderen für die Verwendung als einfaches, kostenfreies Telekooperationstool, gibt dieses Kapitel einen Überblick über verfügbare Systeme in diesen Bereichen.
In Abschnitt 4.1 wird dabei zunächst auf die Entwicklung von Teleradiologiesystemen mit integrierten Telekonferenztools eingegangen, bevor in Abschnitt 4.2 ein eigenständiges Programm für die telekooperative Betrachtung medizinischer Bilddaten vorgestellt wird. Abschnitt 4.3 beschäftigt sich dann mit frei verfügbaren DICOM-Viewern und in Abschnitt 4.4 werden kurz einige Tools für die Bearbeitung medizinischer Bilddaten vorgestellt.
Abschließend werden in Abschnitt 4.5 Programme und Plattformen gezeigt, die für die Anwendung als Lehr- oder Lerntools in der Medizinischen Bildverarbeitung konzipiert wurden.

4.1 Teleradiologiesysteme - Konzepte und Stand der Forschung

Die für die Teleradiologie, also für die Fernübertragung von medizinischen Bilddaten zu diagnostischen Zwecken (siehe Abschnitt 2.1.1.1), eingesetzten Systeme haben sich mit der technischen Entwicklung der letzten Jahre beständig verändert und weiterentwi-

ckelt. Zu Beginn der 1990er Jahre konnten bedingt durch die zunehmende Verfügbarkeit schneller Netze wie z. B. ISDN erste Teleradiologiesysteme entwickelt werden, die den Versand und die kooperative Befundung medizinischer Bilddaten im damals neuen DICOM-Format (siehe Kapitel 3.2) über Standardnetze ermöglichten. Mit Systemen wie *KAMEDIN - Kooperatives Arbeiten und Medizinische Diagnostik auf Innovativen Netzen* [47, 48] und *MEDICUS* [49] entstanden eigenständige Workstations für die Teleradiologie, mit denen räumlich voneinander getrennten Benutzern die Besprechung von gemeinsamen Bildmaterial ermöglicht wurde. Häufig angeboten wurde neben der direkten Audio- und Videokonferenzschaltung die Telekonferenz in Form von Synchronisation der Bildschirminhalte auf den einzelnen Systemeinheiten mit Zeigefunktion und Bildbearbeitungsoptionen. In der Regel mussten die Bilddaten vor so einer Konferenz an alle Teilnehmer versendet werden. Dieses sogenannte *Push-Modell* stellt die klassische Architektur von Teleradiologiesystemen dar, bei dem die Bilddaten in einem Peer-to-Peer-Netzwerk von einer Workstation zur anderen übertragen werden.
Ende der 1990er Jahre verfügte die Mehrzahl der Teleradiologiesysteme über eine DICOM-Schnittstelle zur direkten, digitalen Bilddatenübernahme, die eine netzbasierte Einbindung in radiologieinterne PACS-Systeme ermöglichte. Die Bildübertragung sowie die Kommunikation in einer kooperativen Telesitzung erfolgte TCP/IP-basiert, wodurch die Systeme sowohl über ISDN-Netze als auch in LANs via Ethernet eingesetzt werden konnten. Gleichzeitig fand eine Integration der Systeme in den klinischen Workflow statt, die isolierten Teleradiologieworkstations auf Basis proprietärer Protokolle wurden ersetzt durch allgemeine Workstations für Standardaufgaben im Rahmen der täglichen Routine mit eingebauten Teleradiologiefunktionen. [50]
Heutzutage hat sich dieser Trend weiter fortgesetzt, mit z. B. *CHILI* [51], *JiveX* (Produktgruppe der Firma VISUS), *Hipax* (Steinhart Medizinsysteme) oder *VIMED* (Meytec) existieren modular aufgebaute Produktfamilien, die den verschiedensten Anforderungen im Bereich der Bildbetrachtung, -kommunikation und -archivierung in der Radiologie und anderen bildverarbeitenden Fachgebieten gerecht werden. So gibt es beispielsweise für die Bildbetrachtung neben einfachen DICOM-Viewern zumeist anspruchsvolle 2D- oder 3D-Befundungstools, eigenständige PACS-Systeme für die Bildarchivierung und verschiedene Lösungen für die Teleradiologie, die oft die Möglichkeit zum Abhalten einer Telekonferenz umfassen.
Für eine schnelle Kommunikation von Bild- und Befunddaten - insbesondere vor dem Hintergrund der Zunahme der sektorübergreifenden und integrierten Versorgungsformen, werden in der Teleradiologie heute vielfach neben der einfachen Punkt-zu-Punkt-Übertragung Webserver- oder Webportal-basierte Lösungen eingesetzt. Bei dieser auch

Pull-Modell genannten Architektur (siehe Abbildung 4.1) wird die Bildverteilung über die Web-Technologie realisiert. Mittels eines innerhalb der Klinik sowie von einweisenden Hausärzten und externen Radiologen zugänglichen Webservers werden Daten zum Ansehen und zum Up- und Download zur Verfügung gestellt. Im Gegensatz zum Push-Modell, wo für jeden Teleradiologiepartner eine Verbindung eingestellt werden muss, muss hier bei allen Teilnehmern nur noch jeweils eine Verbindung zum Webserver konfiguriert werden.
Aktuell lässt der Aufbau der Telematikinfrastruktur (siehe Kapitel 2.1.2) mit der Entwicklung von krankenhaus- oder konzernübergreifenden Telemedizinakten wieder einen Paradigmenwechsel in der Teleradiologie erkennen. Vermutlich wird mit der Integration radiologischer Bilder in diese Akten die Teleradiologie zukünftig über die Telemedizinakte abgewickelt werden. [52, 53]

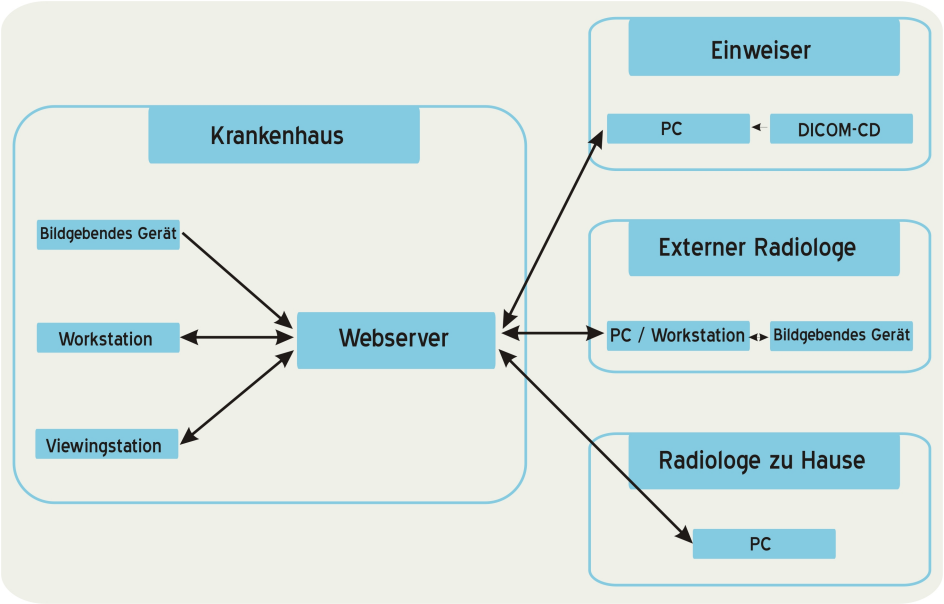


Abbildung 4.1: Webportal zur Bereitstellung von Bildern im Krankenhaus und zum bilateralen Austausch mit externen Partnern

4.2 Telekonferenztools

Software für Telekonferenzen findet sich hauptsächlich als Modul innerhalb der im vorigen Abschnitt erwähnten kommerziellen Teleradiologiesysteme. So bieten unter anderem die Systeme Chili und Hipax neben dem Datenversand auch die Möglichkeit zur kooperativen Befundung.

Frei verfügbare Tools für die kooperative Besprechung medizinischer Daten im Rahmen einer Telekonferenz, bei der zwei oder mehr Teilnehmer miteinander verbunden sind und dabei jeweils dasselbe auf ihren Monitoren sehen sowie die Anwendungen der anderen Teilnehmer fernsteuern können, gibt es jedoch keine.
Mit dem *Java DICOM Viewer* der Medizinischen Informatik der Fachhochschule Aachen wurde zwar im Jahr 2000 ein Java-Applet für die Teleradiologie zur kooperativen Bearbeitung und Konsultation entwickelt [54], das Projekt ist mittlerweile jedoch eingestellt. Eine kommerzielle Telekonferenzapplikation für Windows-Rechner existiert mit der *TeleConsult* genannten Software der Firma MedCom. Das Programm wird unter anderem im *MedNET*-Projekt eingesetzt, in dem durch den Aufbau eines gemeinsamen Netzwerks abgelegene und ländliche Regionen Lateinamerikas mit Gesundheitsleistungen versorgt werden sollen [55].

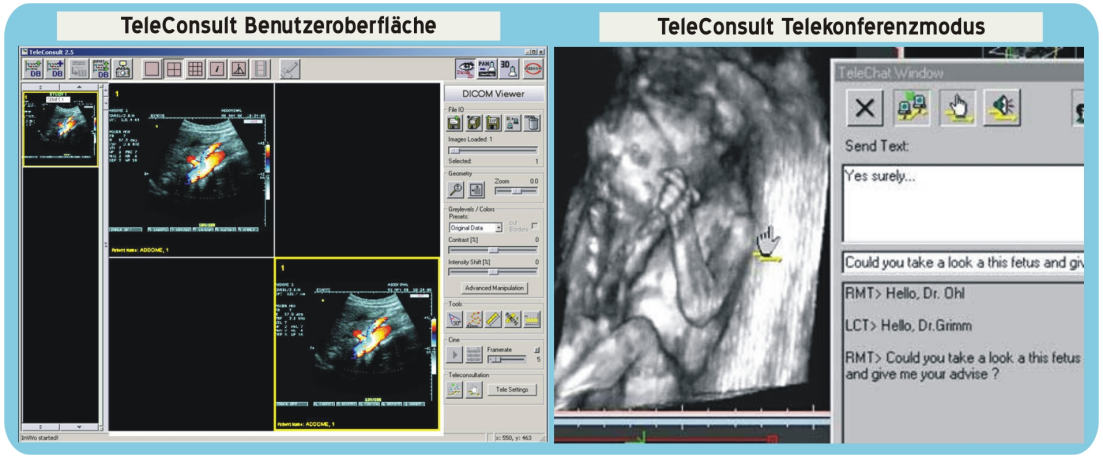


Abbildung 4.2: Die Softwareapplikation TeleConsult

Die TeleConsult-Anwendung stellt eine Kombination aus einem 2D/3D-DICOM-Viewer, einer Bilderzeugungssoftware für Ultraschalldaten, einem medizinischen Kommentierungswerkzeug und einem Telekommunikationstool dar. Abbildung 4.2 zeigt die Benutzeroberfläche, links ist eine Übersichtsdarstellung aller eingeladenen Bilder zu sehen, in der Mitte werden ein oder mehr Bilder im Detail angezeigt und rechts finden sich Funktionen wie Zoom oder Kontrastveränderung zur Bearbeitung der Bilder. Mit TeleConsult können Texte oder Bilder über verschiedene Kommunikationskanäle wie Internet, ISDN und analoge Telefonverbindungen im on- oder offline-Modus ausgetauscht werden. Eine Client-/Server-basierte Softwarearchitektur stellt die zentrale Datenhaltung und den Zugriff auf diese sicher. Im online-Modus besteht dann die Möglichkeit zur Telekonferenz mit einem entfernten Partner auf Basis dieses beiden zugänglichen Datenmaterials.

Neben der Übertragung der Mausbewegungen steht für die Kommunikation zusätzlich ein Chat-Window zur Verfügung (siehe Abbildung 4.2 rechts).

4.3 DICOM-Viewer

Als *DICOM-Viewer* wird eine Anwendung zur Visualisierung von DICOM-Bildformaten mit zugehörigen Metadaten bezeichnet, die im medizinischen Umfeld zur Betrachtung von Patientenbildern eingesetzt wird. Diese Viewer finden sich als Komponente in Teleradiologielösungen, sind aber auch als eigenständige Anwendungen verfügbar.

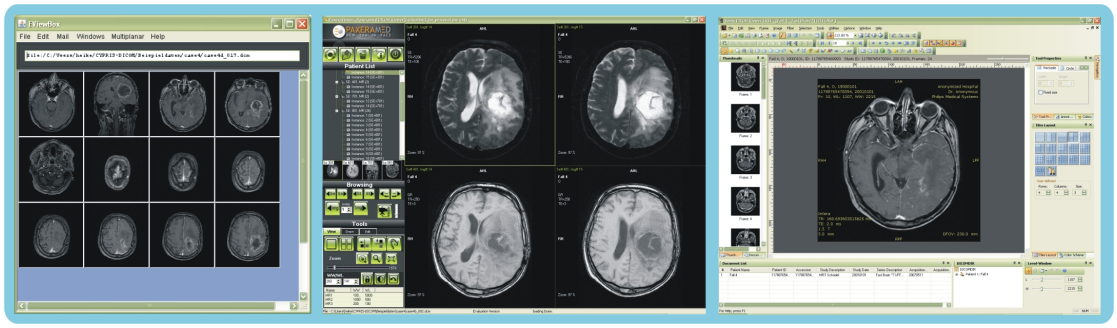


Abbildung 4.3: Beispiele für verschiedene DICOM-Viewer

Neben kommerziellen DICOM-Viewern wie z. B. dem *Rubo Medical Imaging Dicom Viewer* für 750 € oder dem *Sante DICOM Viewer* (ab 50 USD) gibt es auch kostenlos zur Verfügung stehende; eine Liste findet sich z. B. unter `http://www.cabiatl.com/mricro/dicom`. Das *I-do-Imaging-Portal*, auf dem freie Softwareapplikationen für die medizinische Bildverarbeitung vorgestellt werden, verzeichnet weitere kostenlose DICOM-Viewer. Tabelle 4.1 zeigt in alphabetischer Reihenfolge einen Auszug dieser frei im Internet zum Download zur Verfügung stehenden Viewer. Nicht mit aufgelistet sind hier die DICOM-Visualisierungstools, die sich auf ein spezielles medizinisches Anwendungsgebiet konzentrieren wie z. B. *CarDiCon*, ein DICOM-Viewer für kardiologische Filmsequenzen. Auch komplette PACS-Lösungen mit integrierten DICOM-Viewern wie die *Clear Canvas Workstation* oder *ConQuest* wurden in der Tabelle nicht berücksichtigt. Bei den angeführten DICOM-Viewern handelt es sich in der überwiegenden Mehrzahl um einfach gestaltete Programme, die sich durch Benutzerfreundlichkeit, leichte Bedienbarkeit und sinnvolle Funktionen zur Bildbearbeitung und -betrachtung wie z. B. die Fensterungsoperation, das Vergrößern und Verkleinern von Bildern, das Ausmessen einer Strecke in einem Bild, die Invertierung etc. auszeichnen. Darunter fallen ganz schlicht gehaltene Anwendungen wie die *eViewBox* oder das Programm *ezDicom*, die in erster

Name	Hersteller	Plattformen
Aeskulap	Open Source Project	Windows, Linux
DicomLite	3D-Head	Windows
DICOMScope	Offis Institut Oldenburg	Windows
DicomWorks	Puech, Boussel	Windows
eViewBox und eViewBoxApplet	Serge Derhy	alle
ezDICOM	Chris Rorden	Windows
FP Image	fpimage.com	Windows
Imread	David Rubinstein, University of Colorado	alle
iRAD	Lance Physher	MacOSX
JiveX DICOM Viewer	Visus	alle
Kradview	David Santo Orcero	Linux
Madena	USC Radiation Oncology	MacOSX
MEDISP DICOM Viewer	Medical Image and Signal Processing Lab	Windows
MicroDicom Viewer	Simeon Antonov Stoykov	Windows
MIView	Greg Book	Windows
Osiris	University Hosiptal of Geneva	Windows, MacOSX
OsiriXLite	OsiriX Imaging Software	MacOSX
PaxeraViewer	Millenmed, Egypt	Windows
Radscaper	Divinev	alle
Sante DICOM Viewer Free	Santesoft	Windows
SimpleDicom Viewer	UPMC - University Pittsburgh	Windows
Sobox Image Viewer	Sobox Image	Windows
Synedra View Personal	Synedra	Windows
TomoVision	TomoVision	Windows
Tudor DICOM Viewer	CRP Henri Tudor	alle

Tabelle 4.1: Eine Übersicht frei verfügbarer DICOM-Viewer

Linie der Visualisierung eines DICOM-Bildformats dienen und wenige Funktionen zur Bildbearbeitung bieten, ebenso wie aufwendiger gestaltete Tools wie *MicroDicom* oder der *PaxeraViewer*, die einen deutlich größeren Funktionsumfang haben, meist eine Übersichtsleiste der eingeladenen Bilder anzeigen können und neben der Einzelbildansicht über eine Vergleichsansicht von zwei oder mehr Bildern verfügen. Des Weiteren enthält die Tabelle mit dem *Sante DICOM Viewer Free*, dem *Synedra View Personal* oder *Osiris* auch Beispiele für sehr komplex gestaltete DICOM-Viewer, die aufgrund ihres Aufbaus und ihrer Multifunktionalität die Bedienung speziell für Einsteiger erschweren. Abbildung 4.3 zeigt Beispiele für unterschiedlich komplexe DICOM-Viewer.
Mit keinem der hier erwähnten DICOM-Viewer kann eine Telekonferenz wie in Abschnitt 2.1.1.1 definiert abgehalten werden (siehe dazu Abschnitt 4.2). Nur wenige Viewer sind plattformunabhängig oder stehen in mehreren Versionen für verschiedene Betriebssysteme zur Verfügung; für Windows-Plattformen gibt es die meisten DICOM-Viewer.

4.4 Bildverarbeitungsprogramme für die Medizin

Neben den DICOM-Viewern, deren hauptsächlicher Zweck die Visualisierung der DICOM-Bilddaten darstellt, gibt es mittlerweile eine Reihe frei verfügbarer Tools für die“High-End”-Bildverarbeitung dieser Daten. Diese Anwendungen bieten eine große Anzahl integrierter Funktionen und Werkzeuge, um Bilder - insbesondere medizinische DICOM-Bilddaten - interaktiv zu bearbeiten. Oft steht hinter einzelnen Funktionen oder Modulen eine spezielle medizinische Fragestellung.
Ein Beispiel für ein relativ einfach aufgebautes Bildverarbeitungsprogramm stellt das im wissenschaftlichen Sektor, besonders im Bereich der biologischen und medizinischen Bildverarbeitung, weit verbreitete Tool *ImageJ* dar. Es handelt sich um eine Java-basierte Open-Source-Software, für die viele Plugins und Erweiterungen existieren und die unter `http://rsbweb.nih.gov/ij/` kostenfrei zum Download zur Verfügung steht [56, 57].
Andere Tools wie beispielsweise das ebenfalls Java-basierte Programm *MIPAV - Medical Image Processing, Analysis and Visualization*, das wie ImageJ von den U. S. National Institutes of Health entwickelt und kostenfrei im Internet zur Verfügung gestellt wird, oder das von der Mevis Medical Solutions AG und dem Frauenhofer MEVIS - Institute for Medical Image Computing in Bremen herausgegebene Softwareprogramm *MeVisLab* bieten umfassendere Möglichkeiten der Bildverarbeitung, die weit über die grundlegender Algorithmen hinaus gehen. So besitzt MIPAV z. B. eine große Anzahl bereits eingebauter Funktionen für die quantitative Analyse sowie für die 2D-, 3D- und 4D-Visualisierung

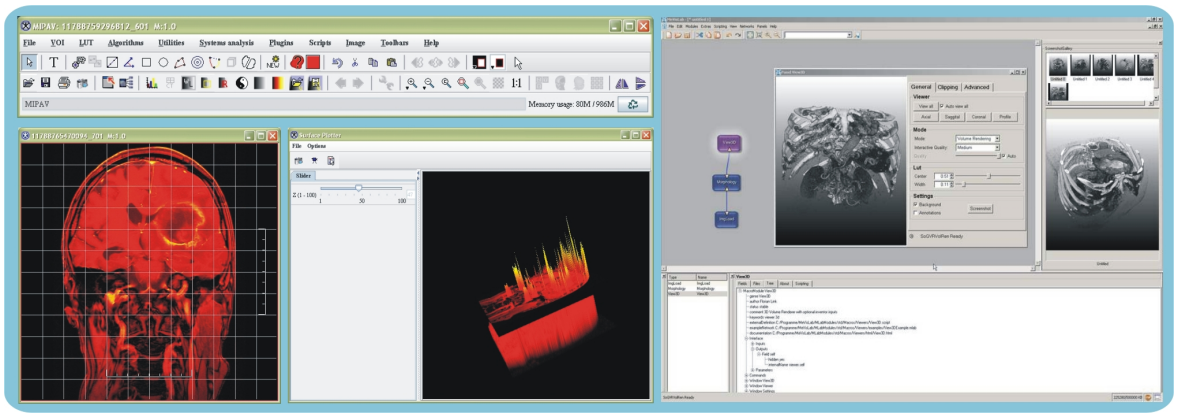


Abbildung 4.4: Beispiele für komplexe Bildverarbeitungsprogramme, links: MIPAV, rechts: MeVisLab

biomedizinischer Bilddaten. Neben grundlegenden Algorithmen wie der Fensterung oder der Filterung von Bilddaten beinhaltet das Tool also auch Module zur Erstellung von Oberflächenstrukturen und 3D-Modellen. Zu den Hauptanwendungen von MeVisLab zählen im diagnostischen Bereich die reproduzierbare und automatisierte Quantifizierung anatomischer Strukturen und pathologischer Prozesse, daher finden sich hier neben den Basisalgorithmen Verfahren für die 3D-Visualisierung, die Segmentierung, Registrierung und quantitative morphologische sowie funktionale Bildanalyse. Aufgrund ihrer Komplexität sind solche Programme in der Regel nicht einfach zu bedienen. Abbildung 4.4 zeigt die Benutzeroberflächen von MIPAV und MeVisLab.

4.5 Software und Lernplattformen für die (Medizinische) Bildverarbeitung

Die Medizinische Bildverarbeitung ist ein Teilgebiet der Medizinischen Informatik, die als Neben- oder Anwendungsfach im Informatikstudium an verschiedenen Universitäten und Fachhochschulen studiert werden kann. Genaueres kann den Webseiten des Berufsverbandes Medizinischer Informatiker `http://www.bvmi.de` entnommen werden. Daneben wird Medizinische Informatik und in diesem Rahmen auch Medizinische Bildverarbeitung als Querschnittsfach im Medizinstudium gelehrt. Dennoch gibt es für die Lehre relativ wenige Tools oder Plattformen, speziell zur Vermittlung grundlegender Algorithmen der Bildvorverarbeitung und Bildanalyse (siehe Kapitel 3.1) existieren kaum Programme oder Online-Angebote. Daher umfasst dieser Abschnitt auch Software und Lernplattformen, die für die Vermittlung der Bildverarbeitung im Allgemeinen konzipiert

wurden.
Als E-Learning-Angebote für die Lehre im Bereich (Medizinische) Bildverarbeitung existieren in erster Linie Online-Module, also Webseiten mit Bildern und Erläuterungen zum Thema. Daneben werden einige dieser Seiten durch spezielle Tools ergänzt, mit denen die vorgestellten Bildverarbeitungsalgorithmen ausprobiert werden können und die zum Teil auch für Übungsaufgaben genutzt werden. Die folgenden Unterabschnitte stellen drei dieser Online-Lernplattformen mit integrierten Tools näher vor. Dabei wird bei der Vermittlung der Lehrinhalte speziell auf das Vorhandensein eines "erklärenden Modus" für die Algorithmen geachtet.

4.5.1 BIO - Bildverarbeitung Interaktiv Online der FH Regensburg

Die Fachhochschule Regensburg bietet im Rahmen der Virtuellen Hochschule Bayern (siehe www.vhb.org) Studierenden, die an einer bayerischen Hochschule immatrikuliert sind, die Möglichkeit, entgeltfrei das netzgestützte Lehrangebot *BIO - Bildverarbeitung Interaktiv Online* zu nutzen. Eine Demoversion findet sich unter http://vhb.fh-regensburg.de/bio/bio_demo.asp. Es handelt sich dabei um ein multimediales Lernsystem über die Grundlagen der Bildverarbeitung, die Themengebiete sind nicht speziell auf medizinische Anwendungszwecke ausgerichtet. Unter einer grafischen Benutzeroberfläche sind verschiedene Medien zur Vermittlung der Inhalte integriert. Neben einem konventionellen Lehrbuch in elektronischer und gedruckter Form wurden audiovisuelle Tutorials als Lernprogramme realisiert, außerdem gibt es ein Java-basiertes interaktives Bildverarbeitungssystem zur Bearbeitung von Praktikums- und Übungsaufgaben. Dieses *IMAGINE* genannte Tool bietet 15 verschiedene Funktionsmodule, von der Histogramm-Operation über Filter-Module bis hin zur Trennung von Farbkanälen (siehe Abbildung 4.5). Ausprobiert werden können diese Operationen auf verschiedenen zur Auswahl stehenden Bildern im TIF-Format, darunter finden sich auch einige Abbildungen aus dem medizinischen Bereich wie z. B. ein CT-Scan. Es besteht jedoch nicht die Möglichkeit, eigene Bilder von der Festplatte in das Programm zu laden oder Bilder im DICOM-Format mit dem System zu betrachten. Auch eine Speicherung der Ergebnisse ist nicht möglich. Um die Bedienbarkeit des Tools zu erleichtern, wurden für alle Module Anleitungen in Form von Flash-Filmen realisiert. Einen "erklärenden Modus", der den Ablauf eines Algorithmus im Detail zeigt, gibt es jedoch nicht.

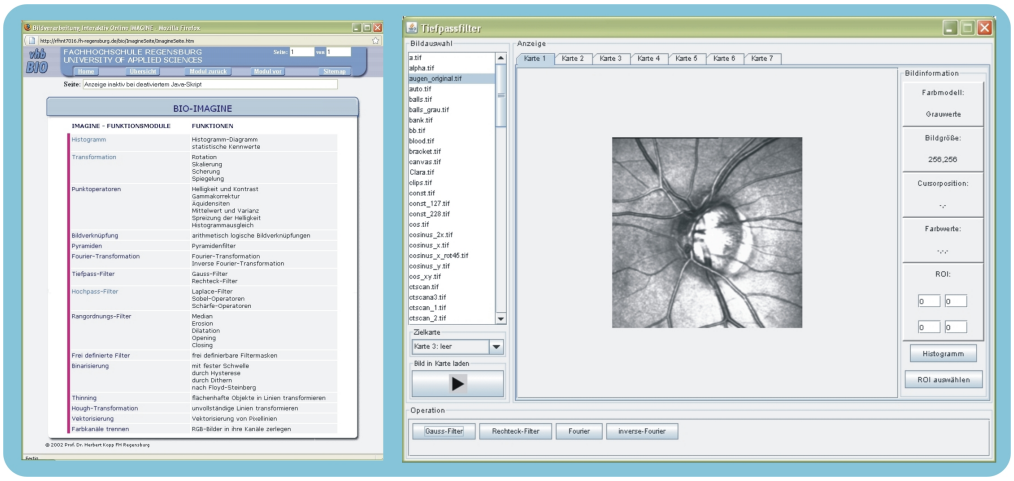


Abbildung 4.5: Das IMAGINE-Tool der FH Regensburg, links: Funktionsumfang des Tools, rechts: das Modul Tiefpassfilter

4.5.2 ODITEB - Open Distributed Textbook der TU München

Mit *ODITEB - Open Distributed Textbook* hat das Institut für Medizinische Statistik und Epidemiologie der Technischen Universität München ein Internet-basiertes interaktives Lehrbuch für die Zielgruppen Mediziner mit Schwerpunkt multimodale Tumordiagnostik und Informatiker mit Schwerpunkt Medizinische Bildverarbeitung entworfen. Kostenlos zur Verfügung gestellt wird das Programm unter `http://www.oditeb.de` seit 1998, derzeit wird an einer Nachfolgeversion gearbeitet, die bis Mitte 2010 fertig gestellt werden soll.

Den weitaus größten Teil des jetzigen Systems machen die Anwendungen für Mediziner aus. Die in ODITEB beschriebene Tumordiagnostik umfasst konventionelles Röntgen, MR- und CT-Bildgebung mit Originaldaten sowie digitalisierte Endosonographie- und Endoskopiebildsequenzen. Ungefähr 150 charakteristische Fälle können mit Hilfe von Expertenkommentaren erarbeitet werden. Die Visualisierung der Bilder erfolgt dabei in einem *Case Viewer* genannten Java-Applet. Dieser integrierte DICOM-Viewer gestattet dem Benutzer einfache Möglichkeiten zur interaktiven Bildbearbeitung wie z. B. frei einstellbare Hounsfield-Fensterungen [58].

Für Informatiker ist ODITEB in erster Linie ein online zur Verfügung stehendes Lehr- oder Nachschlagewerk über die Grundlagen der digitalen Bildverarbeitung für die Medizin. Die Kapitel beinhalten neben einer Einführung und den Grundlagen die Themen Bildtransformation, Bildverbesserung, Segmentierung, Merkmalsextraktion und Klassifikation. Abbildung 4.6 zeigt die Umsetzung des Bereichswachstumsverfahrens (siehe

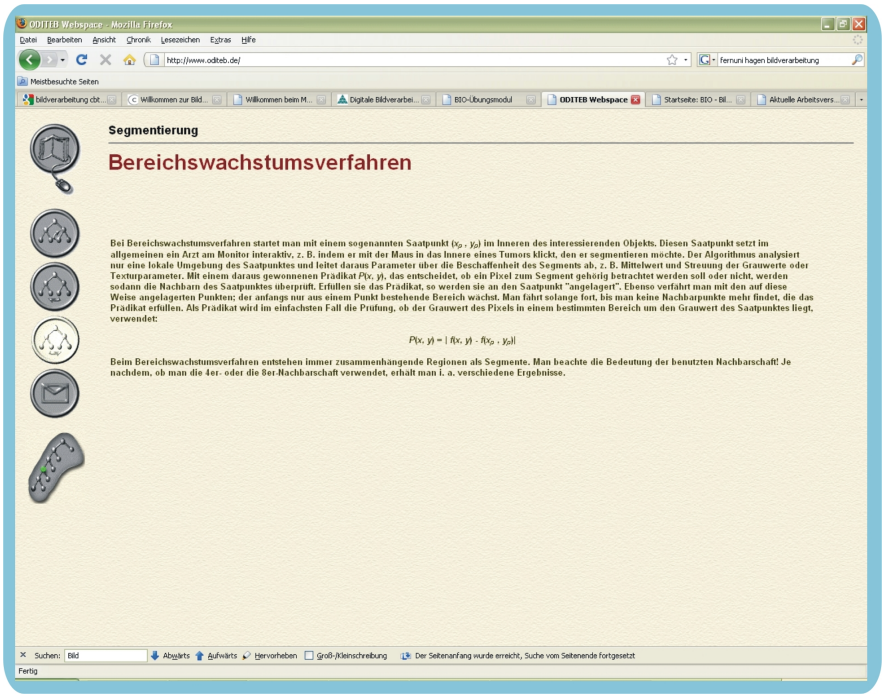


Abbildung 4.6: Die ODITEB-Seiten für Informatiker, exemplarisch ist das Bereichswachstumsverfahren dargestellt

Abschnitt 7.3.6). Alle Kapitel sind speziell auf die medizinische Bildverarbeitung ausgerichtet und beinhalten Beispiele aus dem medizinischen Alltag. Auch hier können mit dem Case Viewer speziell ausgewählte und bearbeitete Bilder zum besseren Verständnis einzelner Verfahren betrachtet werden, für die Faltung gibt es beispielsweise Vorher-Nachher-Ansichten. Der Ablauf einzelner Verfahren wird jedoch nicht in einem "erklärenden Modus" demonstriert, auch können keine eigenen Bilddaten eingeladen oder Ergebnisse abgespeichert werden. Generell steht nur eine recht eingeschränkte Auswahl an Bildverarbeitungsalgorithmen zum Ausprobieren zur Verfügung.

4.5.3 Lehrgang Digitale Bildverarbeitung der FH Jena

Unter `http://www.kmt.fh-jena.de` finden sich auf den Seiten des Instituts für Kommunikations- und Medientechnik verschiedene Module zum Thema Bildverarbeitung. Neben Begriffsdefinitionen und einer Inhaltsübersicht der Studieninhalte gibt es verschiedene Demos zu einzelnen Bereichen der Bildverarbeitung. Sie sind als JavaScript-Objekte oder in Form von Flash-Filmen in die Webseiten eingebettet. Besonders zu erwähnen ist hier das *DBV-E-Learning-Modul* (siehe Abbildung 4.7), mit dem Algorithmen zu den Themen Digitalisierung, Grauwertstatistik, Punktoperatoren, Lokale

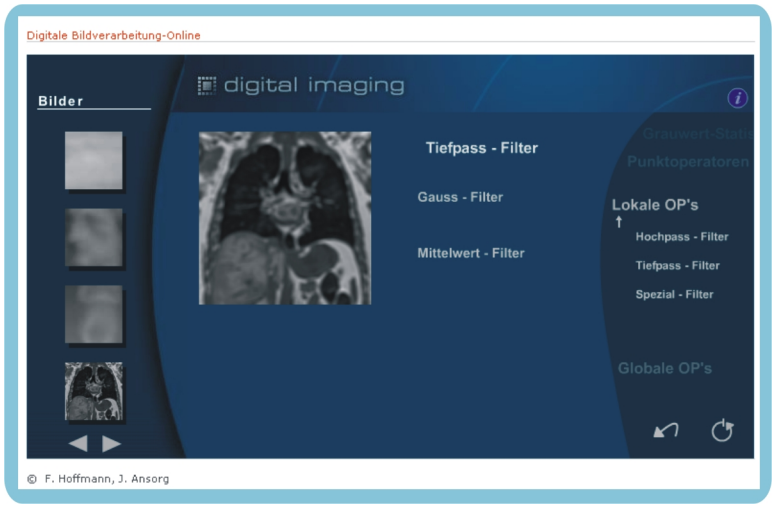


Abbildung 4.7: Das E-Learning-Modul für Bildverarbeitung der FH Jena

Operatoren und Globale Operatoren auf integrierten Beispielbildern ausprobiert werden können. Das Programm beinhaltet nicht die Möglichkeit, eigene Bilddateien hochzuladen oder Ergebnisse abzuspeichern. Spezielle Themen der medizinischen Bildverarbeitung werden nicht behandelt. Ein "erklärender Modus" steht jedoch für ausgewählte Gebiete zur Verfügung, so ist z. B. das Prinzip der Faltung (siehe Abschnitt 7.3.4) wie in Abbildung 4.8 zu sehen ist in einer extra Animation in ihrem Ablauf erläutert.

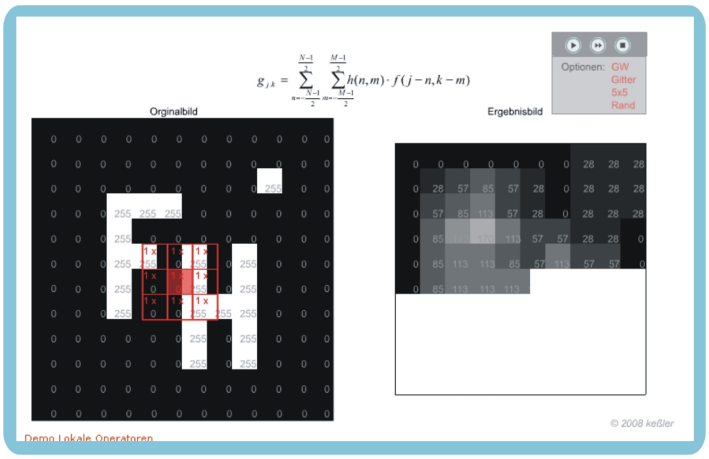


Abbildung 4.8: Der Demonstrationsmodus für lokale Operatoren auf den Seiten der FH Jena

5 Material und Methoden

In diesem Kapitel werden die für die Erstellung dieser Arbeit verwendeten Softwaretechniken vorgestellt, ein tabellarischer Überblick der eingesetzten Tools und Softwarebibliotheken findet sich in Abschnitt 6.7 des folgenden Kapitels.

Um die in der Zielsetzung geforderte Plattformunabhängigkeit zu gewährleisten, wurde Java als Programmiersprache für das hier zu entwickelnde Projekt ausgewählt. Die Sprache und für Aspekte der Aufgabenstellung wichtige Bibliotheken wie z. B. die *Java2D API* für Bilder, *Java RMI* für entfernte Methodenaufrufe oder die Zusatzbibliothek *JavaHelp* werden in Abschnitt 5.1 vorgestellt. Der darauf folgende Abschnitt 5.2 gibt dann einen Überblick über in Java mögliche Programmarten und dabei zu beachtende Besonderheiten, bevor in Abschnitt 5.3 eine Eigenkomposition des Instituts für Medizinische Informatik vorgestellt wird, mit der die Telekonferenzfunktionen für dieses Projekt realisiert werden.

5.1 Java

Java ist eine objektorientierte Programmiersprache, die eine umfangreiche Klassenbibliothek zur Erstellung lokaler, verteilter und internetbasierter Anwendungen zur Verfügung stellt.
Ursprünglich für die elektronische Konsumgüterindustrie zur geräteunabhängigen Steuerung von Fernsehern oder Waschmaschinen entworfen, konnte sich Java vor allem durch die Entstehung des World Wide Web in den 1990er Jahren etablieren. Die Plattformunabhängigkeit der Sprache sowie die Möglichkeit, Programme zu erstellen, die in einem Webbrowser ausgeführt werden können, erwiesen sich als optimal in Bezug auf die verschiedenen, an das Internet angeschlossenen Rechnerarchitekturen. Die Version 1.0 wurde von Sun im Mai 1995 publiziert, mittlerweile liegt Java bereits in der Version 6.18 vor, Version 7 ist in Vorbereitung.

Ein Java-Programm wird durch die sogenannte *Java-Plattform* von der Hardware eines Rechners abgeschirmt. Diese Plattform besteht aus der *Java Virtual Machine (JVM)* und

dem *Java Application Programming Interface (Java API)*, einer Sammlung vorgefertigter Softwarekomponenten wie z. B. Teilbausteinen für grafische Benutzeroberflächen, die in Bibliotheken unterteilt sind.
Mit der JVM benutzt Java für die Ausführung eines Programms einen Interpreter, der den zuvor mit einem Compiler erzeugten Java-Bytecode zur Laufzeit in die jeweilige Maschinenrepräsentation überführt. Der Bytecode kann auf allen Plattformen, für die eine Portierung des Interpreters zur Verfügung steht, ausgeführt werden (siehe Abbildung 5.1).

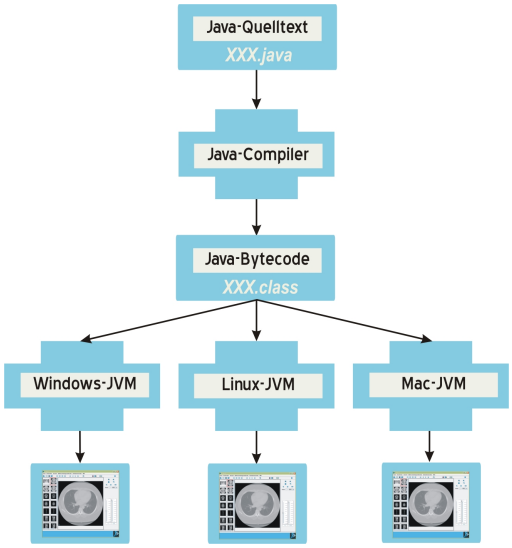


Abbildung 5.1: Die Ausführung eines Java-Programms

Performance-Problemen, die durch die Interpretation der Sprache zur Laufzeit entstehen, konnte durch die Entwicklung von schnelleren Rechner, Just-in-Time-Compilern oder Java-Prozessoren entgegengewirkt werden (Näheres dazu siehe [59]).

Auch wenn Java sich als Programmiersprache für große Anwendungsgebiete eignet (man spricht auch von einer *general-purpose language*), so gibt es dennoch Beschränkungen, die sich vor allem durch das Konzept der Plattformunabhängigkeit ergeben. Da alle Methoden auf allen Systemen lauffähig sein müssen, kann mit Java auf systemabhängige Eigenschaften oft nicht zugegriffen werden. Benutzerrechte etwa können von Java nicht erfragt oder modifiziert werden, da schon die Rechteverwaltungen von Unix und Windows völlig anders aussehen. [60]

In den folgenden Abschnitten werden Sprach- und Zusatzbibliotheken von Java beschrieben, die für die Erstellung dieser Arbeit verwendet wurden.

5.1.1 Java Swing für grafische Benutzeroberflächen

Die Bibliothek *Java Swing* ist seit der Version 2 fester Bestandteil von Java, sie enthält eine Sammlung von Klassen zur Entwicklung grafischer Benutzeroberflächen. Vorher war dies nur unter Benutzung des *Abstract Window Toolkits (AWT)* möglich, das eine

plattformabhängige Peer-Technik zur Darstellung von GUI-Elementen benutzt. Die Visualisierung der Java-Komponenten wird dabei vom zugrundeliegenden Betriebssystem übernommen. Ein einheitliches, plattformunabhängiges Look & Feel der Komponenten kann daher mit dem AWT nicht gewährleistet werden; gleiche Komponenten können auf unterschiedlichen Systemen unterschiedlich aussehen und sich unterschiedlich verhalten. Swing ist im Gegensatz dazu plattformunabhängig gestaltet. Alle Bestandteile sind vollständig in Java programmiert, das zugrundeliegende System und dessen GUI-Elemente haben keinen Einfluss auf das Aussehen und Verhalten der entwickelten Oberfläche. Swing-Komponenten benutzen einfache Grafikprimitiva, um sich selbst zu zeichnen und sehen daher auf jeder Plattform gleich aus. Des Weiteren umfasst Swing eine Reihe zusätzlicher Dialogelemente wie z. B. Fortschrittsanzeigen, Tabellen oder Bäume und die sogenannte *Model-View-Controller-Architektur* für die Komponenten, bei der die Speicherung und Verarbeitung der eigentlichen Daten getrennt von ihrer Darstellung erfolgt.

5.1.2 Bilder in Java

Standardmäßig können mit Java GIF-, PNG- und JPEG-Bilder geladen werden; sie können an verschiedenen Stellen eingebunden und angezeigt werden wie z. B. in Zeichengebieten (Canvas) oder als Icons in Schaltflächen.

Zum Laden der Bilder bietet die Java API mehrere sozusagen historisch gewachsene Möglichkeiten, die neueste wurde mit dem Paket `javax.imageio` in Version 4 eingeführt, um das Lesen und Schreiben von Grafiken zu vereinheitlichen. Für komplexe Grafikoperationen steht in Java die *Java2D API* mit einer umfangreichen Klassenbibliothek zur Bilddarstellung, Bildverarbeitung und Filterung zur Verfügung. Dazu gehören unter anderem Transformationen von Grafiken, Einbinden von Text und Bildern sowie die Einstellung von Zeichenstiftbreite, Linien- und Füllmuster oder Transparenz.

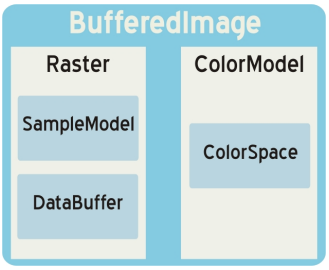


Abbildung 5.2: Das Modell des BufferedImage

Kernklasse des auch *Immediate Mode Model* genannten Java2D-Bildmodells ist das *BufferedImage*, das intern aus zwei Teilen besteht, dem Farbmodell des Bildes und den eigentlichen Bilddaten (siehe Abbildung 5.2). Durch den Zugriff auf diese Daten erlaubt BufferedImage die unmittelbare Manipulation der Pixel in einem Bild.

Neben der Java2D API gibt es mit dem *Java Advanced Imaging* eine mächtige Zusatzbi-

bliothek für die Bildverarbeitung in Java, die auch Dateiformate wie BMP, PNG, PNM und TIFF unterstützt. Da aber auch hier das Bildformat DICOM nicht unterstützt wird, wurde diese Bibliothek hier nicht eingesetzt.

5.1.3 Java RMI für entfernte Methodenaufrufe

Die *Remote Method Invocation (RMI)* ist seit der Version 1.1 eine Java-Standardbibliothek, die Mechanismen für die verteilte Anwendungsprogrammierung bereitstellt. Mit RMI kann der Aufruf einer Methode eines entfernten Java-Objekts auf einer anderen JVM erfolgen, was die Kommunikation über Netzwerke zwischen verschiedenen Rechnern ermöglicht. Dafür wurde eine Java-Variante des *Remote Procedure Calls* [61] realisiert, einem Verfahren, das den Zugriff auf entfernte Serverfunktionen wie lokale Methodenaufrufe aussehen lässt.
Neben RMI gibt es noch andere Standards, die dieses Prinzip implementieren. Dazu zählen vor allen Dingen die *Common Object Request Broker Architecture (CORBA)* [62], ein großes Framework für verteilte Softwarekomponenten und unterschiedliche Programmiersprachen sowie das *Distributed Component Object Model (DCOM)* von Microsoft. Da RMI Teil der Standardbibliothek von Java ist und die Kommunikation in dieser Arbeit zwischen zwei Java-Anwendungen erfolgen soll, bot sich hier die Verwendung von RMI jedoch an.

Die zugrundeliegende Kommunikationsarchitektur besteht aus drei Schichten: der *Transportschicht*, die die Netzwerkverbindung darstellt, der *Remote-Reference-Schicht*, die die Übertragungsprotokolle implementiert und der *Stub-Skeleton-Schicht*, die die Kommunikation der entfernten Objekte übernimmt.

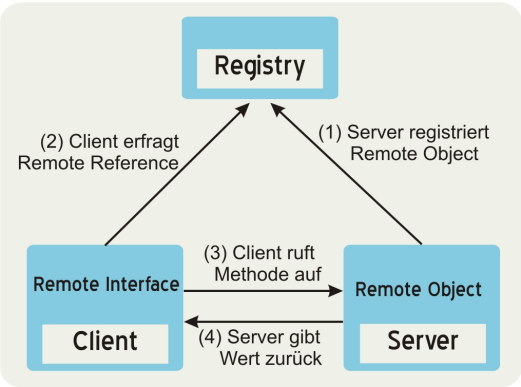


Abbildung 5.3: Der Ablauf bei RMI

Als Standardprotokoll gilt dabei für RMI das *Java Remote Method Protocol (JRMP)*, alternativ kann auch das *RMI IIOP (Internet Inter-ORB Protocol)* zur Integration in CORBA eingesetzt werden.
Für jedes entfernte Objekt, das auf einem Server laufen soll, muss ein *Remote Interface* existieren, das alle entfernten Methoden, die das Objekt bereitstellen soll, deklariert. Die Implementierung dieses Interfaces, das *Remote Object*, muss auf dem Server vorhanden sein. Zugriff auf das entfernte Objekt erhält der Client per *Remote Reference*, eine Referenz, die ihm von der *RMI Registry* übergeben wird. Abbildung 5.3 verdeutlicht den Funktionsablauf von RMI. Zunächst registriert der Server ein Remote Object bei der RMI Registry unter einem eindeutigen Namen. Der Client kann anschließend über diesen Namen bei der Registry die Remote Reference für das entfernte Server-Objekt erfragen und damit dann die entfernten Methoden aufrufen.

5.1.4 Java Cryptography Architecture für die Verschlüsselung

Um Daten verschlüsselt über ein Netzwerk versenden zu können, müssen kryptografische Verfahren eingesetzt werden (siehe dazu Abschnitt 3.3). In Java steht dafür die sogenannte *Java Cryptography Architecture (JCA)* zur Verfügung, eine Architektur, die Design Patterns und Entwürfe für kryptografische Algorithmen und Verfahren spezifiziert. Dabei erfolgt eine Trennung der Konzepte von ihrer Implementation. Unter den kryptografischen Konzepten werden in der JCA Klassen verstanden, die abstrakte Begriffe wie z. B. einen privaten oder öffentlichen Schlüssel, eine digitale Signatur, ein Zertifikat, einen Verschlüsselungsalgorithmus oder einen digitalen Fingerabdruck repräsentieren, ohne dabei jedoch ein konkretes Verfahren für deren Benutzung zu definieren. Die eigentliche Implementation stellen dann sogenannte *kryptografische Provider* zur Verfügung. Die JCA beinhaltet damit Klassen, mit denen man beispielsweise ein Objekt vom Typ Verschlüsselungsalgorithmus erzeugen kann, das aber zur eigentlichen Verwendung einen von einem Provider gelieferten Algorithmus, wie z. B. DES für die symmetrische Verschlüsselung, benötigt. Sun selber liefert mit der *Java Cryptography Extension* einen Standardprovider für grundlegende Operationen mit [63]. Diese Aufteilung wurde vorgenommen, weil bis in die späten 1990er Jahre kryptografische Verfahren in den USA unter das Exportverbot fielen. Auch heute noch bestehen Restriktionen, so darf der DES-Algorithmus beispielsweise nur mit einer maximalen Schlüssellänge von 64 Bit implementiert werden.
Neben dem Standardprovider von Sun stehen - insbesondere für die starke Verschlüsselung - viele andere Provider zur Verfügung, sowohl kommerzielle als auch frei verfügbare. Zu letzteren zählen unter anderem die hier eingesetzten *Bounty Castle Crypto APIs* für

Java.

5.1.5 Die Zusatzbibliothek JavaHelp für die Erstellung der Benutzeranleitung

Mit dem von Sun entwickelten *JavaHelp System* können Online-Hilfen und Dokumentationen für Java-Programme erstellt und eingebunden werden. Das System besteht aus einer umfangreichen, erweiterbaren Spezifikation und API sowie aus einer Referenzimplementierung. Es handelt sich hierbei um eine optionale Zusatzbibliothek, die als Open Source Projekt unter `http://javahelp.dev.java.net` verwaltet wird.
Ein Hilfesystem besteht in JavaHelp aus zwei Teilen: dem *Help Viewer* und dem *Help Set*. Letzteres sind die HTML-basierten Hilfetexte einschließlich Metadaten, die entweder lokal vorliegen oder über das Netzwerk geladen werden, wo sie dann mit dem Help Viewer, einem dreiteiligen Fenster mit Menubar, Navigationsfenster und Inhaltsbereich, angezeigt werden. Volltextsuche, Inhaltsverzeichnis und Index gemäß W3C Standards werden automatisch generiert. Durch die Trennung der Hilfetexte von der Java-Anwendung ist eine unabhängige Aktualisierung der Hilfe möglich.
Zur Erstellung des Help Set mit den Hilfeinformationen bietet sich die Nutzung eines Autorenwerkzeugs an. Hier wurde das im folgenden Abschnitt beschriebene Tool *DobuDish* benutzt.

5.1.5.1 DobuDish für die Erstellung von JavaHelp

Das *DocBook Publishing Framework DobuDish* erzeugt aus XML-Dateien nach *DocBook-Standard* wahlweise die Formate HTML, CHM, PDF oder JavaHelp. DocBook beschreibt dabei eine Dokumenttypdefinition (DTD) für XML-Dateien, die für die Erstellung von Büchern, Artikeln und Dokumentationen im technischen Umfeld ausgelegt ist. DocBook ist ein offener Standard, der von der *Organization for the Advancement of Structured Information Standards (OASIS)* gepflegt wird. [64, 65]

DobuDish ist ein Java-basiertes Programm, was ursprünglich für T-Systems entworfen wurde, heute aber frei verfügbar ist. Die Nutzung dieses Tools ermöglicht es, Anleitungen oder Dokumentationen ausgehend von im DocBook-Standard erstellten XML-Dateien in mehrere Formate zu überführen. So kann beispielsweise neben einer JavaHelp-Anleitung ohne zusätzlichen Aufwand auch eine PDF- oder HTML-Version gleichen Inhalts erstellt werden.
Für die Arbeit mit XML-Dateien empfiehlt sich zusätzlich die Nutzung eines XML-

Editors.

5.2 Application und Applet

Java erlaubt die Erstellung zwei verschiedener Arten von Programmen. Neben der normalen Anwendung - in Java *Application* genannt - gib es die in Webseiten eingebundenen *Applets*. Im Gegensatz zu den Applications müssen Applets nicht auf dem lokalen Rechner des Anwenders installiert werden, sondern können von dem Server, auf dem sie liegen, über das Netz übertragen werden, um dann auf dem lokalen Rechner in einem Java-fähigen Internet-Browser ausgeführt zu werden. Dafür muss der jeweilige Webbrowser über eine Java VM verfügen, die entweder direkt in den Browser integriert sein kann oder in Form eines *Java Plugins*, was die VM der auf dem Rechner installierten Java-Version aufruft, zur Verfügung stehen. Um immer die aktuelle Java-Technologie verwenden zu können, arbeiten heutzutage fast alle modernen Browser mit so einer externen VM, d. h. zur Ausführung von Applets muss Java auf dem Computer installiert sein. Für Browser mit integrierten VMs gilt das nicht, da der Browser selber die zur Ausführung benötigte Technologie bereitstellt. Beispiele dafür sind der Internet Explorer bis Version 5 und der Netscape Navigator bis Version 4, allerdings entsprechen die dort integrierten VMs früheren Java-Versionen.

5.2.1 Applets und Sicherheit

Im Gegensatz zu Applications, bei denen keinerlei Zugriffsbeschränkungen auf Systemressourcen bestehen, unterliegen Applets standardmäßig starken Sicherheitsrestriktionen. Sie laufen innerhalb des Browsers in einer sogenannten *Sandbox* ab und dürfen nicht auf lokale Dateisysteme zugreifen, fremde Programme starten oder Verbindungen zu anderen Rechnern aufbauen als zu dem, von dem sie geladen wurden.
Soll ein Applet dennoch Zugriff auf geschützte Ressourcen haben, beispielsweise um auf Bilddaten von der lokalen Festplatte zugreifen zu können, so müssen die Sicherheitsbestimmungen für das Applet aufgehoben werden. Dazu muss das Applet signiert werden, d. h. es wird digital mit einem Zertifikat unterzeichnet, was der Authentifizierung des Herstellers dient. Wird ein solches Applet in einen Browser geladen, so muss der Nutzer der Ausführung zustimmen, vorher ist kein Zugriff auf Systemressourcen möglich.
Zertifikate für die Signatur eines Applets werden von weltweit bekannten *Certification Authorities* wie beispielsweise *VerigSign*, *Thawte* oder der *Deutschen Telekom AG* ausgestellt. Die Kosten dafür liegen je nach Anbieter zwischen 549 USD und 895 USD bei

einer Geltungsdauer von 2 Jahren. Daneben besteht die Möglichkeit der Selbstzertifizierung mithilfe des zu Java gehörenden *Keytools*. Unter `http://wiklet.javacore.de/index.php/Signieren_von_Applets` kann nachgelesen werden, wie ein signiertes Applet mit einem eigenen Zertifikat erstellt werden kann.

5.2.2 Das Archivformat Jar

Mit dem *Jar-Dateiformat* existiert in Java seit der Version 1.1 eine Möglichkeit, Dateien wie im ZIP-Format komprimiert oder unkomprimiert zusammenzufassen. Java-Applets, die aus mehreren Klassen bestehen, können so als eine Datei in den Browser geladen werden. Anstatt vieler kleiner Server-Verbindungen für jede einzelne Klassendatei muss bei einem Jar-Archiv nur eine HTTP-Verbindung aufgebaut werden, was - trotz der Größe der Jar-Datei - wesentlich effizienter ist, da jeder Verbindungsaufbau überproportional viel Zeit kostet. Ein Vorteil, der durch die Datenkompression noch verstärkt werden kann. Neben den Klassendateien eines Applets oder einer Application beinhalten Jar-Dateien eine sogenannte *Manifest-Datei*, in der verschiedene Informationen über den Inhalt des Archivs stehen. Ist dort der Name der ausführbaren Hauptklasse der Java-Anwendung eingetragen, so kann diese JAR-Datei unter einigen Betriebssystemen wie z. B. Windows und Linux per Doppelklick direkt als Programm gestartet werden - man spricht dann von einem *executable Jar*. Auf allen Betriebssystemen lässt sich so eine Jar-Datei aus der Kommandozeile starten.

5.3 Das Telekooperationswerkzeug Jermes

Die Java-basierte Klassenbibliothek *Jermes* ist eine Eigenkomposition des Instituts für Medizinische Informatik [66, 67], mit der zu jedem Java-Programm eine telekonferenzfähige Variante erstellt werden kann. Das Programm wird dabei zunächst für den lokalen Gebrauch erstellt, bevor eine davon abgeleitete Unterklasse durch die Einbindung der Jermes API zu einer telekonferenzfähigen Version des Programms wird (siehe Abbildung 5.4).
Bei einer Telekonferenz einer Jermes-Anwendung werden die Aktionen der verschiedenen Kommunikationsteilnehmer über ein Netzwerk übertragen und in den jeweils verbundenen Anwendungen auch ausgeführt. Gleichzeitig werden für das Telepointing die Mauszeiger der jeweils anderen Teilnehmer zusätzlich zum eigenen in der Anwendung dargestellt. So kann jeder Benutzer die Anwendungen der anderen Teilnehmer fernsteuern.

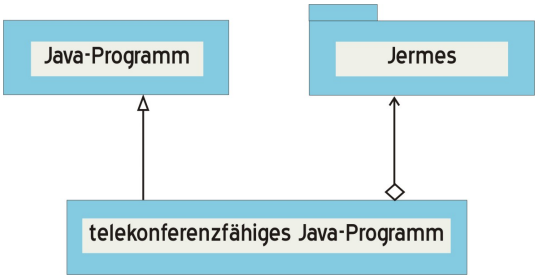


Abbildung 5.4: Durch Einbindung von Jermes wird ein Java-Programm telekonferenzfähig

Als Kommunikationsarchitektur liegt dabei standardmäßig eine Client-Server-Architektur zugrunde wie sie in Abbildung 5.5 zu sehen ist. Die Kommunikation der Teilnehmer erfolgt hier über einen Server-Thread, das Server-Objekt wird dabei als Thread in einer JVM zusammen mit dem Client-Objekt erzeugt. Einer der Clients ist also immer gleichzeitig der Server. Für den grundlegenden Kommunikationsaufbau und die Datenübermittlung wird in Jermes die Remote Method Invocation (siehe Abschnitt 5.1.3) benutzt.

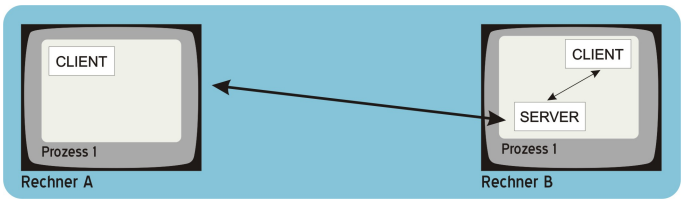


Abbildung 5.5: Die Kommunikationsarchitektur bei Jermes

Zur Synchronisation der Benutzeraktionen werden Java-Events übertragen. Dabei werden für die Darstellung der Mausbewegungen spezielle `MouseEventListener`-Objekte in den Komponenten einer Telekonferenzanwendung registriert, die die Mausposition an die Anwendungen der übrigen Teilnehmer übertragen und dort einen virtuellen Mauszeiger zeichnen. Genauso werden `EventListener`-Objekte für komponentenspezifische Events wie z. B. das Drücken eines Buttons registriert, die dann die entsprechenden Events an die Anwendungen der übrigen Teilnehmer übertragen, wo die Aktionen über die *Accessibility API* von Java, die die Bedienung eines grafischen Benutzerelements über eine abstrakte Schnittstelle ermöglicht, simuliert werden können. Abbildung 5.6 zeigt den prinzipiellen Ablauf einer Aktionsübertragung während einer Telekonferenz in Form eines UML-Kommunikationsdiagramms.

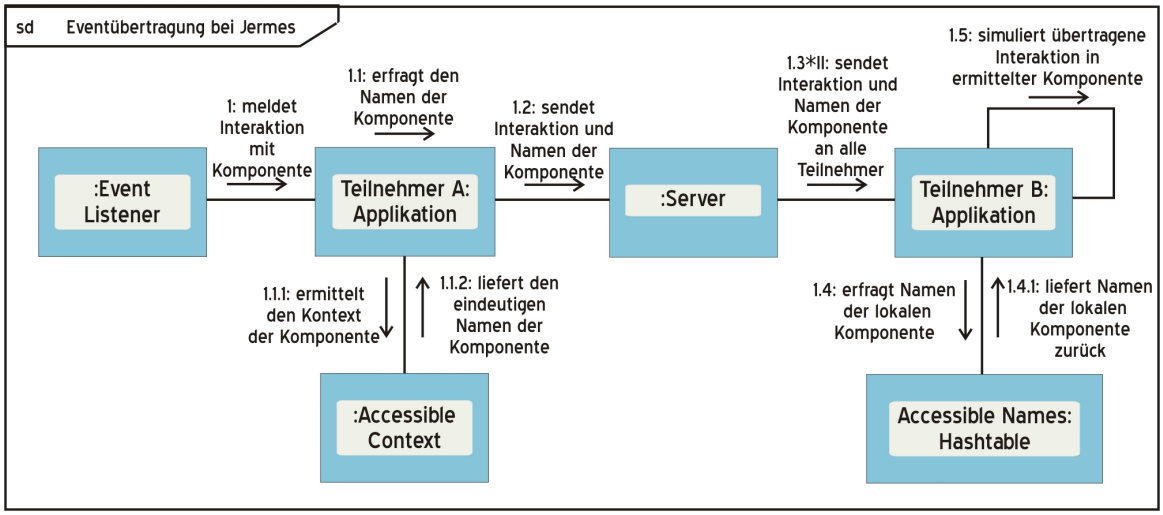


Abbildung 5.6: Der Ablauf einer Telekonferenz bei Jermes

6 JAMIP - Systementwurf und Implementierung

Dieses Kapitel beschreibt die mithilfe der in Kapitel 5 erläuterten Softwaretechniken erfolgte Umsetzung der Zielsetzung dieser Arbeit (siehe Abschnitt 1.2).

Zu Entwicklungsbeginn wurde dabei das hier zu realisierende Projekt unter dem Namen *CYPRIS* geführt, im späteren Verlauf jedoch in *JAMIP* - für ***Ja**va based **M**edical **I**mage **P**rocessing* - umbenannt. In dieser Ausarbeitung wird im Weiteren ausschließlich letzterer Name verwendet.
Wie die Langform dieses Namens vermuten lässt und Abschnitt 5.1 bereits zu entnehmen war, wurde Java als Programmiersprache für die zu entwickelnde Software ausgewählt, um von den oft heterogenen Systemen in Universität und Gesundheitswesen unabhängig zu sein und die geforderte Plattformunabhängigkeit zu gewährleisten.

Aus den weiteren in der Zielsetzung definierten Vorgaben, ergab sich für die Erstellung des Programms folgendes Vorgehen:
Zunächst musste als Basissoftware ein einfaches Bildverarbeitungstool entwickelt werden, was medizinische Bilddaten im DICOM-Format einlesen, anzeigen und mit Standardfunktionen bearbeiten kann. Dieser Teil der Programmentwicklung wird in Abschnitt 6.1 beschrieben.
Dabei wurde JAMIP wie in Abschnitt 6.2 beschrieben in zwei ausführbaren Varianten - als und als Application - realisiert, da es möglichst unabhängig von bestehenden Strukturen und bereits vorhandenen Tools einsetzbar sein sollte.
Anschließend wurde das Tool unter Einbindung der in Abschnitt 5.3 beschriebenen Jermes API zu einer telekonferenzfähigen Software. Dieser Entwicklungsschritt wird in Abschnitt 6.3 vorgestellt.
Die realisierten Telekonferenzfunktionen erforderten dann den Einbau eines zusätzlichen Moduls für den Datenversand, das in Abschnitt 6.4 vorgestellt wird. Außerdem wird hier die für den Datenversand über ein Netzwerk zu berücksichtigende Verschlüsselung der

Daten beschrieben.
Danach erfolgte die Umsetzung der in Ziel 2 definierten Vorgaben. In das bestehende Programm wurden Grundlagenalgorithmen aus dem Bereich der Medizinischen Bildverarbeitung integriert und lehr-/lernunterstützend aufbereitet. Details zu dem hier entwickelten Konzept und der Implementierung können Abschnitt 6.5 entnommen werden.
Eine abschließende Übersicht der bei der Implementierung dieses Projekts entstandenen Klassen und Pakete findet sich in Abschnitt 6.6 dieses Kapitels, bevor Abschnitt 6.7 zum Schluss einen tabellarischen Überblick über die bei der Erstellung von JAMIP eingesetzten Softwaretools und Programmiersprachen liefert.

6.1 Die Entwicklung des Basisprogramms

Dieser Abschnitt befasst sich mit der Konzeption des Grundsystems zur Anzeige und Bearbeitung medizinischer Bilddaten im DICOM-Format. Der Aufbau der grafischen Benutzeroberfläche dieses Programms soll dabei zu Lern- und Übungszwecken dem Aufbau üblicher Bilddarstellungssoftware für die Medizin wie z. B. DICOM-Viewern (siehe dazu Abschnitt 4.3) nachempfunden sein. Ebenso soll der Standardfunktionsumfang von JAMIP dem solcher Systeme entsprechen.
Es wurde daher ein Programm entwickelt, das Bilder im DICOM-Format einlesen, anzeigen und mit folgenden Standardfunktionen bearbeiten kann: Vergrößerung, Verkleinerung, Invertierung, Fensterung, Anzeige der Headerinformationen sowie Abstandsmessung und Mittelwertermittlung in ausgewählten Bildregionen. Die grafische Benutzeroberfläche beinhaltet ein Hauptfenster für die Einzel- und Mehrbildansicht medizinischer Bilddaten, ein Nebenfenster mit einer Übersichtsanzeige aller in das Programm geladenen Bilder, eine Menüleiste, über die die Funktionen des Programms gesteuert werden können, und eine Buttonleiste für die Ausführung der Standardfunktionen. Für die Oberfläche des Programms wurde ein einheitliches Design in Blautönen entworfen bestehend aus einem Logo und Icons für die Standardfunktionen.

Für dieses Basisprogramm wurde eine *JAMIP API* entwickelt, die sich in mehrere Java Packages unterteilt und es erlaubt, JAMIP leicht und effizient in seinen Funktionen zu erweitern. Ein Teil dieser Klassen stellt Grundlagen und Schnittstellen dar, während ein weiterer Teil Klassen für die Darstellung von Bedien- und Oberflächenelementen wie z. B. Fenstern, Slidern oder Dialogfenstern bereitstellt, um das JAMIP-Design für alle Elemente der grafischen Benutzeroberfläche herzustellen. Für die Entwicklung dieser GUI-Klassen wurden Elemente des Java Swing Pakets (siehe dazu Abschnitt 5.1.1)

benutzt.
Zu dem Basisprogramm gehören folgende Pakete:

- `jamip.action` enthält Aktionsklassen z. B. für das Einlesen von Bilddateien
- `jamip.gui` enthält mehrere Klassen um grafische Benutzerelemente im einheitlichen JAMIP-Design zu erstellen, dazu gehören u. a. die Klasse `JamipFrame`, die ein Fenster mit dem JAMIP-Logo im Rahmen öffnet oder die Klasse `JamipOptionPane`, mit der Dialogfenster mit bestimmten JAMIP-Icons erzeugt werden können
- `jamip.image` enthält Klassen, die für die Verwaltung und die Anzeige der Bilddaten erforderlich sind
- `jamip.util` enthält Klassen für mathematische Funktionen

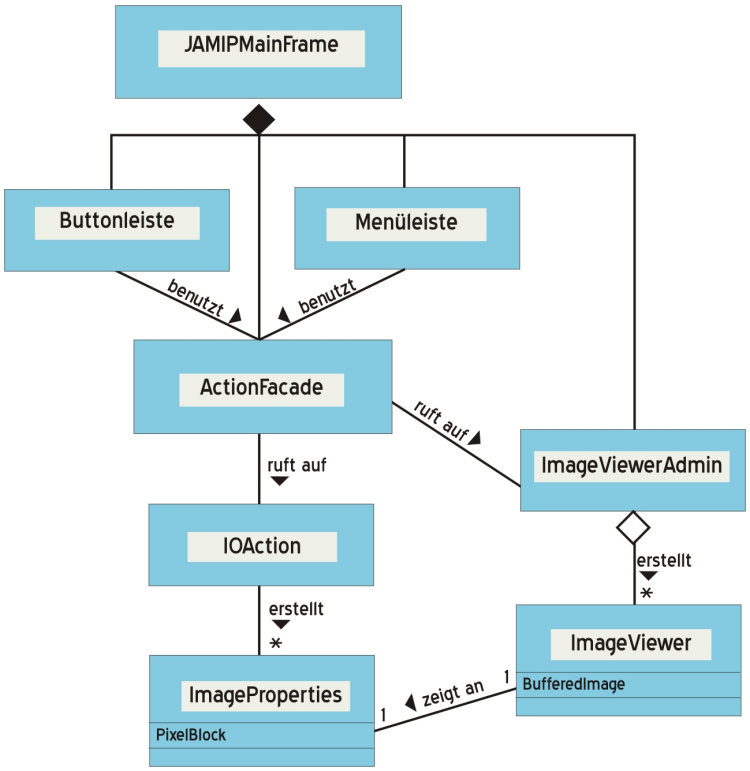


Abbildung 6.1: Das Klassendiagramm des Basissytems von JAMIP

Die oben beschriebene GUI des Programms wird durch die Klasse `JAMIPMainFrame` repräsentiert, ein von `javax.swing.JFrame` abgeleitetes Fenster, das alle Bedien- und Bildanzeigeelemente enthält. Dabei wird die Verwaltung der in das Programm

geladenen Bilder mit Übersichtsanzeige, Einzel- und Mehrbildansicht von einem in das Hauptfenster integrierten Panel, dem `ImageViewerAdmin`, übernommen.
Für die Aktionsverwaltung wurde das *Facade-Entwurfsmuster* implementiert. Der JAMIPMainFrame erzeugt daher ein Objekt der Klasse `ActionFacade`, um darüber alle Programmaktionen auszuführen. Die Eingabe einer Benutzerinteraktion über eines der Bedienelemente wird dabei an die ActionFacade weitergeleitet, die die für die Ausführung der Eingabe erforderlichen Subsystemklassen aufruft. Die verschiedenen Bedienelemente kommunizieren so lediglich mit der Fassaden-Klasse, die den Zugriff auf die komplexeren Programmstrukturen verwaltet. Bei Bedarf lässt sich ein Austausch dieser Elemente ohne großen Aufwand vornehmen, außerdem erleichtert dieses Vorgehen den Zugriff auf gemeinsam genutzte Operationen.

Soll ein vom Benutzer ausgewähltes Bild in das Programm geladen werden, so erzeugt die ActionFacade ein Objekt der Klasse `InOutAction`, die das Einladen der Bilddaten übernimmt. Es können sowohl JPG- und PNG-Dateien als auch DICOM-Files eingelesen werden. Der Einleseprozess für letztere wird in Abschnitt 6.1.1 gesondert beschrieben. Für jedes geladene Bild wird ein Objekt der Klasse `ImageProperties` erzeugt. Zentrales Element dieser Klasse ist der `PixelBlock`, eine Wrapperklasse für ein Bilddatenobjekt. Das zugrunde liegende Bild kann dabei eine beliebige Anzahl an Farbkanälen und Frames enthalten. Für die Anzeige des Bildes werden dann diese ImageProperties an den ImageViewerAdmin des Hauptfensters übergeben, der für die Visualisierung der Bilddaten ein Objekt der Klasse `ImageViewer` verwendet. Mit dieser von `javax.swing.JPanel` abgeleiteten Panel-Klasse wird aus einem PixelBlock-Objekt ein `BufferedImage` (siehe dazu Abschnitt 5.1.2) erzeugt. Für die Ausführung der Standardfunktionen auf einem Bild werden dann Klassen des Pakets jamip.util benutzt, wo Klassen für Funktionen wie den Zoom, die Invertierung oder die Fensterung auf einem PixelBlock implementiert sind.
Abbildung 6.1 zeigt das Klassendiagramm dieses Grundsystems von JAMIP.

6.1.1 DICOM-Daten in JAMIP

Um Bilddaten im in der Medizin üblichen DICOM-Format (siehe dazu Kapitel 3.2) anzeigen zu können, musste für diese Arbeit ein DICOM-Viewer realisiert werden. Dieses Format zählt nicht zu den standardmäßig in Java importierbaren Bildformaten. Es gibt mittlerweile jedoch z. B. mit *ImageJ* oder dem *PixelMed Java DICOM Toolkit* verschiedene DICOM-Zusatzbibliotheken für Java. Zum Zeitpunkt der Konzeption und

Implementierung dieses Programms war dies jedoch noch nicht der Fall, was die Entwicklung eines eigenen Einleseprozesses für DICOM-Daten erforderlich machte. Dabei wurde eine Bibliothek von Java-Klassen entwickelt, die genau den für die Visualisierung der Daten benötigten Teil von DICOM umfasst. Es wird hier davon ausgegangen, dass die Bilder als einzelne DICOM-Files vorliegen. Das für JAMIP umgesetzte Konzept wird im nachfolgenden Abschnitt beschrieben. Themen wie die Archivierung und Kommunikation im DICOM-Format wurden in dieser Arbeit nicht behandelt.

6.1.1.1 Das Einlesen eines DICOM-Files

Wie in Abschnitt 6.1 beschrieben, wird ein Bild in JAMIP von der Klasse ImageViewer verwaltet. Für die Anzeige wird hier das in der Java-Standardbibliothek enthaltene BufferedImage benutzt, das aus einem Pixelarray erzeugt werden kann. Für die Anzeige der Bilddaten eines DICOM-Files, dessen genauer Aufbau in Abschnitt 3.2.1 beschrieben wurde, müssen daher die Pixeldaten in ein Array ausgelesen werden. Daneben soll in JAMIP die Anzeige der Bildzusatzinformationen zu jedem DICOM-Bild möglich sein. Das entwickelte API-Package `jamip.dicom` besteht aus folgenden Klassen:

- `DicomParser` für das Einlesen der DICOM-Datei
- `DicomImage` für die Verwaltung des DICOM-Bildes
- `DicomHeader` für die Verwaltung der Bildzusatzinformationen
- `DicomData` für die Zusammenfassung von Header und Bild

In Abbildung 6.2 wird anhand eines UML-Kommunikationsdiagramms das Zusammenspiel dieser Klassen beim Einleseprozess eines DICOM-Files verdeutlicht. Es wurde dabei folgendes Vorgehen für den Umgang mit den DICOM-Rohdaten umgesetzt:

1. Das DICOM-File in ein Byte-Array einlesen
2. Prüfen, ob nach der DICOM File Preamble, das 4 Byte lange DICOM-Präfix “DICM” angegeben ist
3. Aus den File Meta Elements mit der “Explicit VR Little Endian” die Transfersyntax für das Data Set auslesen, Tag (hex 0002, hex 0010)
4. Je nach Transfersyntax die Data Elements auslesen
5. Die Pixeldaten auslesen und daraus das Bild erzeugen

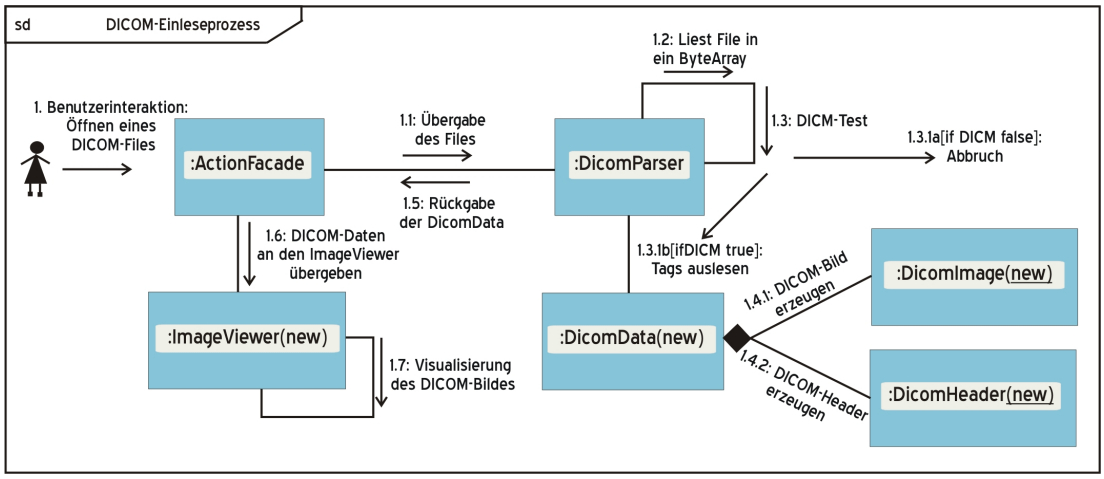


Abbildung 6.2: Der Einleseprozess für DICOM-Dateien in JAMIP

Der Ablauf in JAMIP gestaltet sich folgendermaßen: Nach Auswahl eines DICOM-Files durch den Benutzer wird dies zunächst von einem Objekt der mit DicomParser bezeichneten Klasse in ein Byte-Array eingelesen, da es sich bei DICOM um ein binäres Dateiformat handelt. Der DicomParser prüft dann zuerst, ob das 4 Byte lange DICOM-Präfix “DICM” angegeben ist, es sich also um eine DICOM3.0-Datei handelt, bevor mit dem Einlesen der Datei fortgefahren wird. Ist dies der Fall, werden mit der in den File Meta Elements angegebenen Transfersyntax die Headerinformationen und Pixeldaten des Bildes ausgelesen. Jetzt wird ein neues Objekt der Klasse DicomData erzeugt, in der die Elemente des DICOM-Files, also die bildbegleitenden Informationen und die Pixeldaten abgelegt werden. Dabei werden der Header und das eigentliche Bild getrennt voneinander in den Klassen DicomImage und DicomHeader verwaltet. Die DicomData werden dann über die ActionFacade dem ImageViewer übergeben, der aus den Pixeldaten des DicomImage ein BufferedImage für die Anzeige des Bildes und aus den Elementen des DicomHeaders ein Fenster mit den Bildzusatzinformationen erzeugt.

6.2 Application und Applet

Wie in Abschnitt 5.2 beschrieben existieren in Java mit Applets und Applications zwei Möglichkeiten zu Erstellung von Programmen. Da hier ein möglichst flexibel einsetzbares Tool implementiert werden sollte, wurde JAMIP in beiden Varianten erstellt. In dem Package `jamip.start` findet sich sowohl die Klasse `JAMIPApplet` als auch die Klasse `JAMIPApplication`. Beide rufen nach ihrer jeweiligen Initialisierung die ebenfalls in dieses Package integrierte Hauptklasse des Programms JAMIPMainFrame

auf. Für ihre Ausführung werden beide Programmvarianten mit dem Archivformat Jar (siehe dazu Abschnitt 5.2.2) komprimiert und in eine Datei zusammengepackt. So ist die Ausführung der JAMIP-Application unter Windows mit einem Doppelklick möglich und das JAMIP-Applet kann als eine Datei vom Server geladen werden.
Für die Realisierung des Applets mussten die in Abschnitt 5.2.1 beschriebenen Sicherheitsrestriktionen umgangen werden, da das JAMIP-Applet Zugriff auf die Systemressourcen erhalten sollte, um Bilddaten von der lokalen Festplatte laden zu können. Um die teuren Kosten für den Kauf eines Zertifikats einer Certifictaion Authority zu umgehen, wurde das hier erstellte Applet zunächst für Testzwecke mittels eines mit dem Java-Keytool selber erstellten Zertifikats signiert. Der nachträgliche Kauf eines Zertifikats ist jedoch jederzeit möglich.
Das Applet wurde für JAMIP so konfiguriert, dass es die Übernahme von Parametern aus einer HTML-Datei erlaubt. Dabei können die Dateinamen von Bilddaten, die sich auf dem Server befinden, auf dem auch das Applet liegt, übergeben werden. So ist es bei richtiger Ansteuerung des Applets durch die HTML-Seite möglich, das JAMIP-Applet mit bereits eingeladenen Bildern zu starten.
Da JAMIP mit jeweils aktuellen Java-Versionen erstellt wurde, können ältere Webbrowser mit einer bereits installierten JVM für die Ausführung des hier entstandenen Applets nicht genutzt werden. Die Installation eines Java Plugins ist daher erforderlich.

6.3 Die Realisierung der Telekonferenz

Für die Umsetzung der Telekonferenzfunktionen wurde in JAMIP die in Abschnitt 5.3 beschriebene Jermes API verwendet. Dafür wurde von der Hauptklasse JAMIPMainFrame eine Unterklasse `JAMIPTeleFrame` gebildet, die ein Objekt der Klasse `SystemFrame` aus Jermes miteinbindet und hierüber die Anmeldung des Programms über RMI als Server oder Client übernimmt. Die grafischen Komponenten der Hauptklasse werden dann über einen eindeutig zuordnenbaren Namen registriert, um so Events von außen an sie weiterleiten zu können. Da Elemente diesem Prozess nicht nachträglich unterzogen werden können, müssen bei dem Einsatz von Jermes alle grafischen Komponenten, die mittels Telepointing fernsteuerbar gemacht werden sollen, zu Beginn einer Konferenz feststehen. Aus diesem Grund ist es mit Jermes nicht möglich, ein bereits laufendes Programm mit Telekonferenzfunktionen zu versehen. Für diese Arbeit wurde daher unter dem Namen *JAMIPTele* eine telekonferenzfähige Variante von JAMIP erstellt und hier integriert, d. h. sie kann aus JAMIP heraus gestartet werden. JAMIPTele ist daneben auch eigenständig lauffähig und wurde sowohl als Applet als auch als Application realisiert.

Zum Starten einer Konferenz meldet sich der erste Teilnehmer als Server an und die übrigen registrieren sich bei diesem durch die Eingabe seiner IP-Adresse als Clients. Bei allen Benutzern erscheinen dann die Mauszeiger der anderen Teilnehmer in unterschiedlichen Farben.

Da auch Bildpanele nicht nachträglich telekonferenzfähig gemacht werden können, müssen die in einer Konferenz anzuzeigenden Bilder vor dem Aufruf des Hauptfensters ausgewählt und an dieses übergeben werden. Werden die Bilder erst während einer telekooperativen Sitzung aufgerufen und in die Oberfläche des Programms integriert, so kann in ihnen kein Telepointing stattfinden. Daher wurde für JAMIPTele die Dateiauswahl vor den Aufruf der fernsteuerbaren Benutzeroberfläche gesetzt, so dass das Programmfenster mit den bereits eingeladenen Bildern erscheint. Die Applet-Variante wurde auch hier so konzipiert, dass die Möglichkeit besteht, Bilddaten über eine HTML-Seite an das Programm zu übergeben. Steht ein gemeinsam zugänglicher Server mit Bildmaterial zur Verfügung, so lassen sich bei entsprechender Konfiguration der Webseite die Bilder durch Anklicken in JAMIPTele öffnen.

Für die Integration in das hier zu realisierende Projekt mussten an der Jermes API einige Änderungen und Erweiterungen vorgenommen werden, so wurde beispielsweise der SystemFrame, der die Verbindungen zu den Konferenzpartnern herstellt und anzeigt in seinem optischen Design an das Aussehen der JAMIP-Komponenten angepasst. Außerdem wurde aus Gründen der Bedienfreundlichkeit ein neues Anmeldefenster entworfen. Daneben ist in Jermes nur die Übertragung von Standard-Events implementiert, so dass dieser Teil der API im Verlaufe der Entwicklung von JAMIP - insbesondere für die Übertragung von Aktionen für Bildverarbeitungsalgorithmen - angepasst werden musste. Dennoch konnten nicht alle Funktionen vollständig für den Telekonferenzmodus übernommen werden, so ist beispielsweise eine Vergleichsansicht in JAMIPTele nicht möglich, da hierfür wieder grafische Komponenten nachträglich fernsteuerbar gemacht werden müssten. Die Hauptklasse JAMIPMainFrame wurde daher durch die boolesche Variable `remote` ergänzt, die im true-Zustand einige Elemente der Benutzeroberfläche nicht erzeugt. Des Weiteren wurde die Hauptklasse der JAMIPTele-Anwendung um ein Chatfenster für die zusätzliche Kommunikation während einer telekooperativen Sitzung ergänzt. Der JAMIPTeleFrame hat außerdem im Unterschied zum JAMIPMainFrame eine Namensleiste, die die Namen der miteinander verbundenen Teilnehmer in ihren jeweils zugeordneten Farben anzeigt. In denselben Farben erscheinen auch die Mauszeiger der Teilnehmer und die Kommentare im Chatfenster.

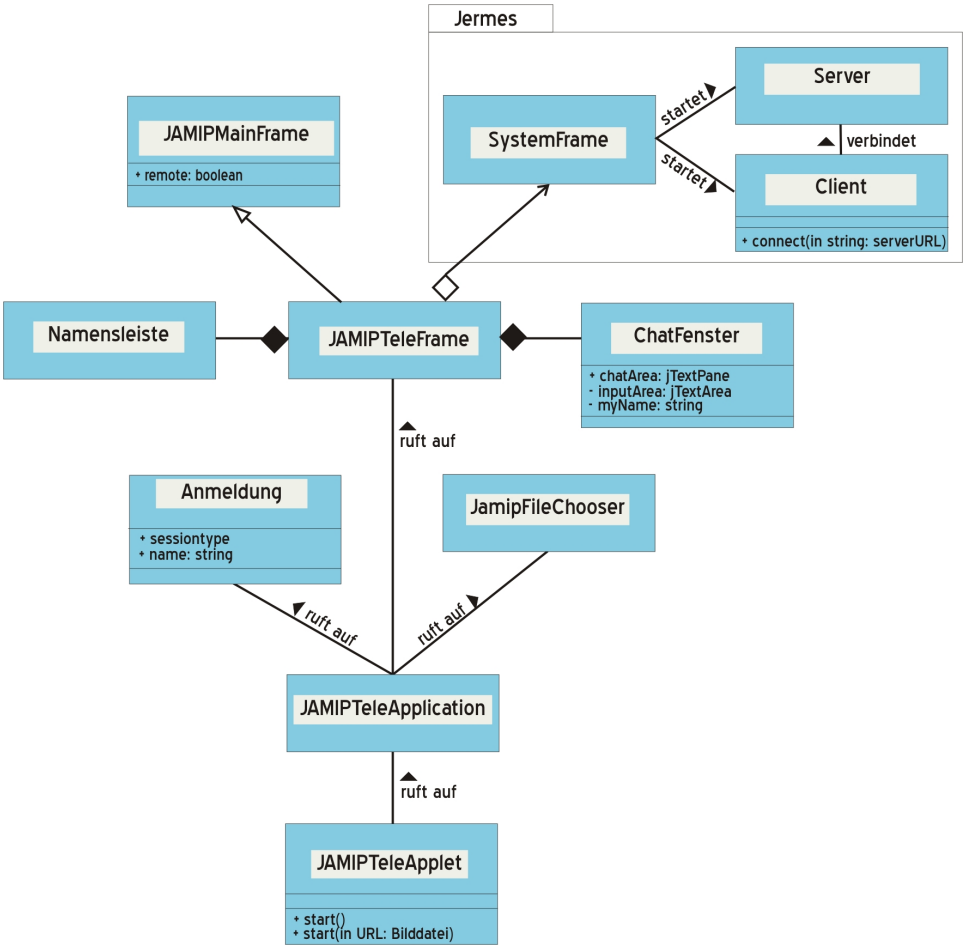


Abbildung 6.3: Das Klassendiagramm der Hauptklassen von JAMIPTele

JAMIPTele stellt somit eine fernsteuerbare Version von JAMIP dar, die sich in den zuvor genannten Punkten von diesem unterscheidet. Änderungen und Erweiterungen von JAMIP bewirken in der hier realisierten Umsetzung immer auch Änderungen bei JAMIPTele. Soll eine Funktion nicht übernommen werden, so lässt sich dies über die remote-Variable steuern. Einen Überblick gibt das in Abbildung 6.3 zu sehende Klassendiagramm von JAMIPTele.

6.4 Datenversand

Um Bilddaten im Rahmen einer Telekonferenz mit JAMIP betrachten zu können, müssen die Daten entweder jeweils lokal bei den Teilnehmern vorliegen oder zentral auf einem gemeinsamen Server vorhanden sein. Da gemeinsam zugängliche Server nicht immer zur Verfügung stehen und der Einsatz von JAMIP möglichst unabhängig von bestehen-

den Strukturen und Anwendungen gestaltet werden sollte, wurde daher zusätzlich der Datenversand über ein Netzwerk in das System integriert. So kann für die Ausführung des Telekonferenz-Programms als lokale Anwendung (Application) der Versand medizinischen Bildmaterials über alternative Programme wie z. B. DICOM-E-Mail erfolgen oder direkt aus JAMIP. Für lehr- oder lernbezogene Anwendungen ergibt sich so die Möglichkeit zum Bild- oder Ergebnisaustausch.

Da es sich bei medizinischen Bilddaten, insbesondere bei DICOM-Daten, die einen personenbezogenen Header beinhalten, um sensible Daten handelt, mussten für die Konzeption eines Moduls für den Datenversand auch datenschutzrechtliche Aspekte genau wie gesetzliche Grundlagen berücksichtigt werden (siehe dazu Abschnitt 2.1.3, insbesondere 2.1.3.1). Neben der Realisierung des eigentlichen Transfers der Daten war daher auch die Entwicklung eines Konzepts für die Verschlüsselung der Daten mit kryptografischen Methoden erforderlich.
Abschnitt 6.4.1 beschreibt zunächst die Versandarchitektur von JAMIP, bevor in dem darauf folgenden Abschnitt 6.4.2 das realisierte Verschlüsselungsverfahren vorgestellt wird.

6.4.1 Der Datenversand in JAMIP

Der Datenversand wird in JAMIP mithilfe der Java-Bibliothek RMI (siehe Abschnitt 5.1.3) für entfernte Methodenaufrufe realisiert. So können Bilddaten aus dem Programm heraus von dem Rechner des Senders zu dem Rechner Empfängers, auf dem ebenfalls eine Programm-Version von JAMIP läuft, über ein Netzwerk geschickt und dort abgelegt werden. Es wurde dafür das API-Package `jamip.transport` entwickelt, das aus folgenden Klassen besteht:

- `JamipClient` der Absender der Daten, der entfernte Methoden des Empfängers aufrufen kann
- `JamipServer` das RemoteInterface, das die entfernten Methoden definiert
- `JamipServerImpl` die Implementierung des RemoteInterface auf Empfängerseite

Der Ablauf des Datenversandes ergibt sich damit bei JAMIP folgendermaßen: Bei Start des Moduls für den Datentransport meldet sich jedes JAMIP-Programm automatisch bei der RMI-Registry als Server an, um Daten empfangen zu können. Um Daten an ein

anderes System senden zu können, muss das empfangende System ebenfalls bei der Registry als Server gemeldet sein. Jedes Programm ist damit gleichzeitig Client und Server, fungiert aber je nach gerade ausgeführter Aktion immer nur in einer der beiden Rollen. Der Absender ist der Client, der über die RMI-Registry und die IP-Adresse eine Verbindung mit dem Server als Empfänger herstellt. Dann kann der Client Methoden auf dem entfernten Server-Rechner aufrufen, die in dem RemoteInterface JamipServer beschrieben sind. Für den Datenversand wird hier die Methode `fileDownload()` aufgerufen, an die die ausgewählten Dateien als Parameter übergeben werden. Da entfernte Methoden nur serialisierbare Daten als Parameter erlauben, können die Bilder nicht als Dateien, in Java repräsentiert durch `java.io.File`, übermittelt werden, sondern müssen als Byte-Array übergeben werden. Auf Server-Seite erfolgt dann die Rückwandlung dieses Byte-Arrays in das File-Format und die Ablage der Bilddatei. Gleichzeitig wird Clientseitig eine Kopie der versendeten Daten erstellt, um das Wiederauffinden zu vereinfachen. Die Daten werden auf beiden Seiten in gleich lautenden Unterverzeichnissen relativ zum installierten JAMIP-Programm abgelegt.
Der gesamte Prozess des Datenversandes ist als Thread realisiert worden, so können gleichzeitig Daten versendet, empfangen und das Programm für die Bildverarbeitung genutzt werden.

6.4.2 Die Verschlüsselung der Daten in JAMIP

Für das im Rahmen dieser Arbeit entwickelte Programm wurde ein PGP-ähnliches hybrides System für den sicheren Transport medizinischer Bilddaten über ein Netzwerk unter Nutzung asymmetrischer und symmetrischer Verfahren konzipiert (siehe dazu Kapitel 3.3). Benutzt wird hierfür die Java Cryptography Architecture (JCA) und ein Provider für die starke Verschlüsselung (siehe Abschnitt 5.1.4). Dabei wurde zunächst das *Cryptix-Projekt* der Cryptix Foundation verwendet, bis dies 2005 eingestellt wurde. Aus diesem Grund werden nun in dieser Arbeit die *Bounty Castle Crypto APIs* für Java eingesetzt.

Der interne Ablauf der Verschlüsselung in JAMIP stellt sich folgendermaßen dar: Auf Senderseite wird vom System zunächst ein zufälliger neuer und einmaliger Session Key generiert, mit dem die ausgewählten Bilder unter Verwendung eines symmetrischen Chiffrieralgorithmus verschlüsselt werden. Der Session Key selbst wird dann mit einem asymmetrischen Verfahren und dem Public Key des Empfängers chiffriert, den dieser dem Sender dafür übermittelt. Dies wird per RMI über einen entfernten Methodenaufruf realisiert. Anschließend findet die eigentliche Transaktion statt. Der Sender schickt die

verschlüsselten Daten und den verschlüsselten Session Key an den Empfänger, der letzteren zuerst mit seinem Private Key und damit dann die eigentlichen Bilddaten entschlüsselt. Anders als bei den in Abschnitt 3.3.3 erwähnten Systemen, findet auch die Generierung des Schlüsselpaares für die asymmetrische Verschlüsselung des Session Keys hier jedes Mal neu statt. Nach Beendigung des Datentransports wird also nicht nur der Session Key gelöscht, sondern auch das benutzte Public-Private-Key-Paar, wodurch eine zusätzlich zu sichernde Aufbewahrung des Private Keys nicht mehr nötig ist.

Bei der Implementierung des Verfahrens für JAMIP wurden für die symmetrische Verschlüsselung des Datensatzes mit DES und IDEA zwei klassische Chiffrieralgorithmen ausgewählt, die alternativ zur Wahl stehen. DES ist ein bereits 1977 vom amerikanischen National Bureau of Standards entwickelter Algorithmus mit einer Schlüssellänge von 64 Bit für die kommerzielle Datenverschlüsselung. Der 1972 an der ETH-Zürich entwickelte IDEA-Algorithmus hat dagegen eine Schlüssellänge von 128 Bits, ist somit etwas langsamer, dafür aber auch sicherer als der DES-Algorithmus. Als asymmetrisches Verfahren zur Verschlüsselung des Session Keys wird in JAMIP das RSA-Verfahren mit einer Grundeinstellung der Zahlengröße von 1024 Bit verwendet (siehe dazu Abschnitt 3.3.2 und [45]).

Für die Versendung von DICOM-Bilddaten wurden zwei verschiedene Varianten für den Umfang der Verschlüsselung implementiert: das DICOM-File kann als Ganzes verschlüsselt werden oder es wird jeweils nur der Teil des DICOM-Files, der die personenbezogenen Headerinformationen enthält, verschlüsselt. Durch Letzteres verkürzt sich natürlich der Aufwand und damit auch die Übertragungszeit.

6.5 Die Bildverarbeitung

Laut Zielsetzung sollten in JAMIP zu Lehr- und Lernzwecken im Studienfach Medizinische Informatik Grundlagenalgorithmen aus dem Bereich der Medizinischen Bildverarbeitung integriert werden. Da sich die Software sowohl an der FernUniversität Hagen als auch in Präsenzveranstaltungen an der Universität zu Lübeck einbinden lassen sollte, wurden in erster Linie Bildverarbeitungsalgorithmen aus den hier stattfindenden Vorlesungen umgesetzt.
Um den Studenten mit dem Tool den Ablauf der Methoden im Detail verdeutlichen zu können, wurde für JAMIP neben der “normalen” Ausführung der Algorithmen der sogenannte *Animationsmodus* konzipiert. In diesem Modus wird der Ablauf eines Verfahrens

in seinen einzelnen Schritten visualisiert.
Da es sich bei den für JAMIP realisierten Verfahren um ganz unterschiedliche Bildverarbeitungsmethoden handelt, konnte hier kein einheitliches Konzept für die Erstellung des jeweiligen Animationsmodus umgesetzt werden. Grundsätzlich öffnet sich für den Animationsmodus eines Verfahrens immer ein neues, eigenes Fenster, in dem der Ablauf eines Algorithmus dann verdeutlicht wird; die konkreten Realisierungen können Abschnitt 7.3 entnommen werden.

Die Bildverarbeitungsroutinen befinden sich in JAMIP in dem Package `jamip.bildverarbeitung` und sind dann hier wiederum nach Methode in einzelne Pakete unterteilt. Die Verfahren arbeiten immer auf dem im Hauptfenster des Programms zu sehenden Bild bzw. auf den zugrunde liegenden Bilddaten, die ihnen von der ActionFacade übergeben werden. Ansonsten sind sie jedoch unabhängig von anderen Programmteilen implementiert. Die Schnittstelle zu den Verfahren stellt damit der ImageViewer dar mit den ImageProperties des Bildes, die die Pixeldaten des Bildes als Objekt der Klasse PixelBlock verwalten. Auf diesem PixelBlock finden die eigentlichen Operationen statt. Dafür wurde die abstrakte Klasse `AbstractFilter` entworfen, die die Basisklasse für jede Filteroperation auf einem PixelBlock bildet. Diese Klasse stellt die Filterkomponente, das Eingabebild, und die Filterparameter bereit, so dass jede konkrete Klasse auf gleiche Rahmenbedingungen aufbauen kann. Konkrete Implementierungen dieser Klasse unterscheiden sich so nur noch in ihrer Filterung, die mit der Methode `filterStep(PixelBlock in, PixelBlock out)` implementiert werden muss.

Da die Bildverarbeitungsroutinen im "normalen" Modus auf den Bildern im Hauptfenster von JAMIP ausgeführt werden, wurde ein Objekt der Klasse `OpList` zu den ImageProperties, die das eigentliche Bild repräsentieren, hinzugefügt, um Operationen wieder rückgängig machen zu können. Zu jedem Bild wird jetzt eingetragen, welches Verfahren bereits auf ihm ausgeführt wurde.

6.5.1 Bedienungsanleitung

Die Integration der verschiedenen Bildverarbeitungsalgorithmen und die dadurch bedingte teilweise nicht ganz einfache Bedienung des Tools erforderte die Konzeption einer Bedienungsanleitung für JAMIP. In dieser Anleitung werden wesentliche Voraussetzungen für den Einsatz des Programms beschrieben, die Benutzeroberfläche sowie die Bedienung sämtlicher Zusatzfenster für die Bildverarbeitung erklärt. Dabei wird davon

ausgegangen, dass die Verfahrensweisen der Algorithmen bekannt sind.
Zur Erstellung dieser Hilfe wurde die in Abschnitt 5.1.5 beschriebene Zusatzbibliothek JavaHelp verwendet, die Funktionen für die Generierung eines Hilfesystems mit Inhaltsangabe und Suchfunktionen aus vorliegenden Hilfetexten bietet. In JAMIP findet sich dieser Teil des Programms in dem Paket `jamip.help`.
Die Hilfetexte wurden für dieses Projekt im XML-Format nach der DocBook-DTD erstellt und anschließend mit dem in Abschnitt 5.1.5.1 beschriebenen Tool DoBuDish in das für die JavaHelp-Anleitung erforderliche Format konvertiert. Gleichzeitig konnte so aus den XML-Dateien auch eine PDF- und eine HTML-Version der Bedienungsanleitung generiert werden. Für die Erstellung der XML-Dateien wurde der XML-Editor *XMLSpy* der Firma Altova verwendet.

6.6 Übersicht der implementierten Klassen

Dieser Abschnitt zeigt in dem in Abbildung 6.4 zu sehenden Paketdiagramm eine vereinfachte Übersicht der in dieser Arbeit entstandenen Klassen und Pakete inklusive der erwähnten Zusatzbibliotheken.

6.7 Software-Übersicht

JAMIP wurde mit den Entwicklungsumgebungen *Eclipse* der Eclipse Foundation und dem von Sun herausgegebenen *NetBeans* implementiert. Diese Tools zählen zu den bekanntesten Open-Source-Umgebungen für die Java-Entwicklung und sind selbst ebenfalls in Java geschrieben.
Um mehrere Bibliotheken im Jar-Format in einer einzigen Jar-Datei zusammenzufügen, wurde das *Fat Jar Plugin* für die Entwicklungsumgebung Eclipse benutzt.
Klassen- und Kommunikationsdiagramme, die sich in der hier schriftlich vorliegenden Ausarbeitung finden, sind nach *UML2.0-Notation* erstellt.
Tabelle 6.1 gibt einen abschließenden Überblick über die für die Erstellung dieser Arbeit verwendete Software inklusive der Versionsangabe. Die erforderliche Software für die Ausführung des Programms wird in Kapitel 7 beschrieben.

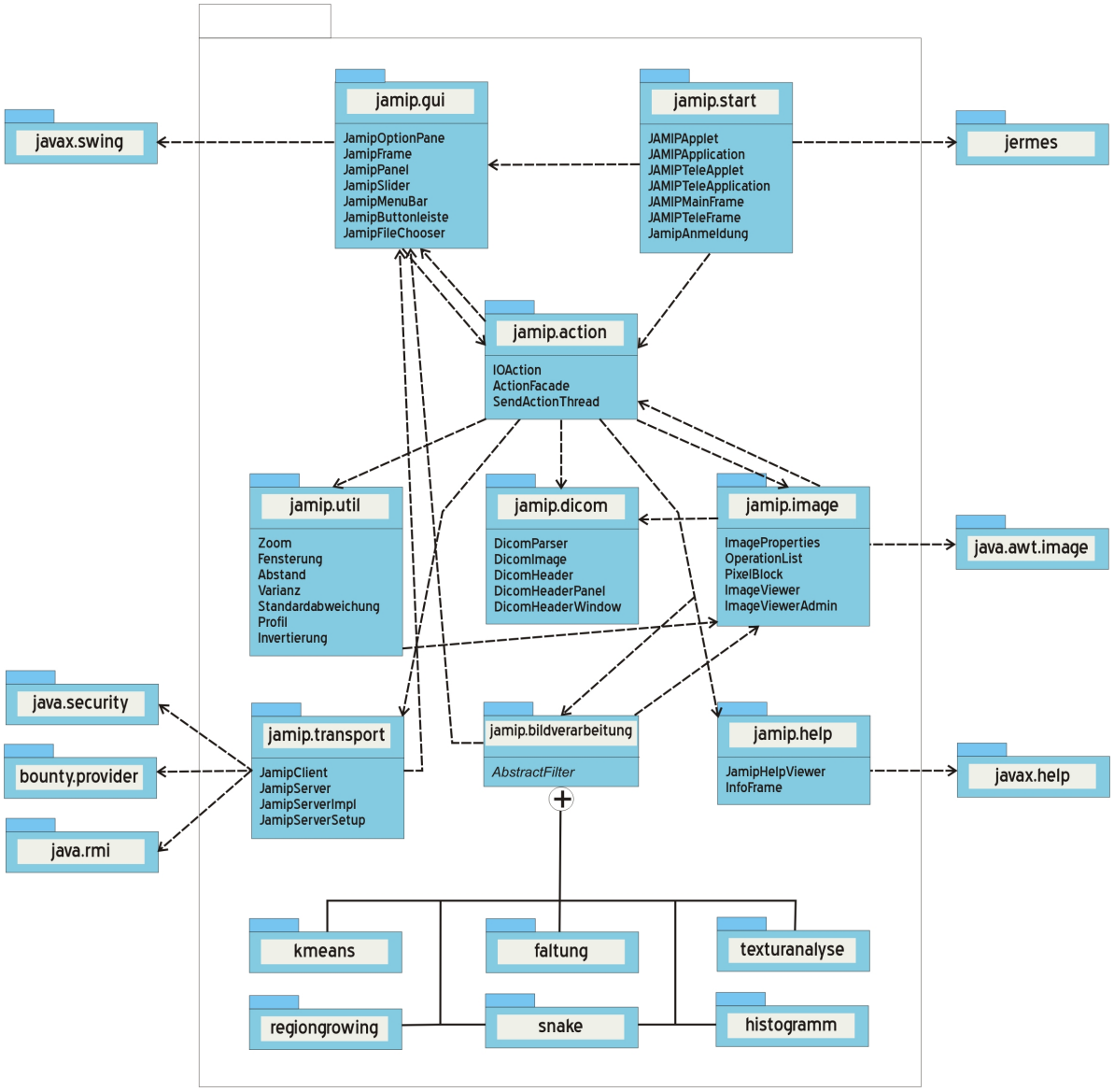


Abbildung 6.4: Die wichtigsten Klassen von JAMIP in der Übersicht als Paketdiagramm

Software	Version	Beschreibung
Java / JDK	1.6.0_05	Übersetzung des Java-Quellcodes und Java-Interpreter
Bouncy Castle Crypto API	1.45	Java-Implementierung von Verschlüsselungsalgorithmen
JavaHelp	2.0	System zur Erstellung von Hilfedateien
DocBook	4.5	Dokumenttypdefinition (DTD) für die Erstellung von Büchern, Artikeln und Dokumentationen
DobuDish	1.1.1	Tool zur Erzeugung von javahelp-, PDF- und HTML-Dateien aus DocBook-XML-Dateien
XMLSpy	2009 Standard Edition	XML-Editor der Firma Altova
Jermes	1.0	Telekonferenz-API
Netbeans	6.8	Entwicklungsumgebung
Eclipse	3.5	Entwicklungsumgebung
Fat jar Plugin	0.0.31	Eclipse Plugin zur Erstellung eines großen Jar-Files
Firefox	3.6	Internetbrowser

Tabelle 6.1: Die verwendete Software

7 JAMIP - die Realisierung

Mit JAMIP ist am Institut für Medizinische Informatik der Universität zu Lübeck ein Java-basiertes und damit plattformunabhängiges Software-Tool entstanden, mit dem Methoden und Algorithmen aus dem Anwendungsbereich der Medizinischen Bildverarbeitung erlernt und demonstriert werden können. Die Basis des Programms bildet ein Bildverarbeitungstool, was für die Anzeige und Bearbeitung medizinischer Bilddaten einen DICOM-Viewer sowie die für solche Viewer üblichen Standardfunktionen beinhaltet. Telekooperative Sitzungen zur Besprechung oder Bearbeitung dieser Daten sind mit dem integrierten aber auch eigenständig lauffähigen Telekonferenz-Modul JAMIPTele möglich.
Speziell für die Anwendung in der Lehre stehen einige der integrierten Bildverarbeitungsroutinen nicht nur in ihrer "normalen" Ausführung zur Verfügung, sondern sind im sogenannten Animationsmodus realisiert, der einen Algorithmus schrittweise multimedial animiert in seinem Ablauf erklärt. Zum besseren Verständnis wurden außerdem unterschiedliche Sichten auf die zugrunde liegenden Daten realisiert, so ist beispielsweise auch eine numerische Ansicht der Bilddaten möglich.
Zusätzlich können für den Einsatz in der Telemedizin oder in der Fernlehre mit JAMIP Daten verschlüsselt über ein Netzwerk verschickt werden.

Sowohl JAMIP als auch JAMIPTele sind frei verfügbare Tools, die jeweils in zwei Varianten - als Applet und als Application (siehe dazu Abschnitt 5.2) - zur Verfügung stehen und von der Webseite des Projekts `http://www.imi2.uni-luebeck.de/jamip` heruntergeladen bzw. gestartet werden können (siehe Abbildung 7.1). Die Java-basierte Anwendung ist in beiden Fällen als ausführbare jar-Datei realisiert worden, in die alle importierten Pakete wie javahelp (Abschnitt 5.1.5) für die Bedienungsanleitung oder die Zusatzbibliotheken für den verschlüsselten Datenversand integriert sind. Diese Versionen lassen sich auf jedem handelsüblichen PC installieren. Für die Entwicklung wurde Java in der Version 1.6 verwendet, zur Ausführung als Anwendung sollte daher auch zumindest die JRE in der Version 1.6 auf dem Computer verfügbar sein. Die Applet-Varianten von JAMIP und JAMIPTele müssen hingegen nicht installiert werden, sie laufen mit

Abbildung 7.1: Die Webseite des Projekts

Hilfe des Java-Plugins im Browserfensters ab. Auch dafür muss Java auf dem Rechner des Nutzers installiert sein sowie das erwähnte Plugin für den Browser. Für das Starten von JAMIP oder JAMIPTele als Applet eignet sich am besten ein Firefox-Browser der Version 3 oder höher. Da es sich hier um signierte Applets (siehe Abschnitt 5.2.1) handelt, die zur Anzeige von Bildern evtl. auf die Festplatte des Nutzers zugreifen, muss der Nutzer der Ausführung des Applets erst zustimmen (siehe Abbildung 7.2). Weitere Hinweise zum Programmstart und zu den Systemvoraussetzungen sowie eine ausführliche Bedienungsanleitung finden sich ebenfalls auf der Webseite des Projekts.

Da es sich bei medizinischen Bilddaten, insbesondere bei DICOM-Daten, die einen personenbezogenen Header beinhalten, um sensible Daten handelt, müssen sicherheitstechnische Aspekte wie der Schutz der Privatsphäre von Patient und Arzt, der Datenschutz sowie die ärztliche Schweigepflicht bei dem Umgang mit solchem Material und damit auch bei dem Einsatz von JAMIP berücksichtigt werden. Generell sollte daher sichergestellt sein, dass im Zusammenhang mit der Nutzung nicht-anonymisierter medizinischer Bilddaten nur autorisiertes Personal Zugang zu dem Programm erhält.

Abbildung 7.2: Zustimmung zur Ausführung des Applets erforderlich

Dieses Kapitel liefert eine ausführliche Beschreibung des in dieser Arbeit entstandenen Software-Systems und zeigt die Realisierung der Anforderungen aus Kapitel 1 und damit die Umsetzung des Systementwurfs, der in Kapitel 6 beschrieben wurde. Die Abschnitte 7.1 und 7.2 geben dabei einen Überblick über das Grundprogramm von JAMIP mit seiner Benutzeroberfläche und den Standardfunktionen. Anschließend stellt Abschnitt 7.3 dann die integrierten Operationen und Algorithmen aus der Medizinischen Bildverarbeitung und deren Umsetzung ausführlich dar, bevor in Abschnitt 7.4 die Funktionsweise des Datenversandes und der hier eingesetzten Verschlüsselungsmechanismen beschrieben wird. Auf das Telekonferenz-Modul JAMIPTele und seine Unterschiede zu JAMIP wird schließlich in Abschnitt 7.5 gesondert eingegangen. Im letzten Abschnitt 8.1 dieses Kapitels werden dann Einsatzszenarien und Anwendungsbeispiele beider Module beschrieben.

7.1 Die Benutzeroberfläche

Zur Anzeige und Bearbeitung medizinischer Bilddaten wurde die gängiger medizinischer Bilddarstellungssoftware nachempfundene und in Abbildung 7.3 dargestellte Benutzeroberfläche für JAMIP realisiert. Neben Bilddaten im in der Medizin üblichen DICOM-Format können auch Bilder im GIF-, PNG- oder JPG-Format mit dem System betrachtet und bearbeitet werden. Der Zugriff auf die Bilddaten kann je nach Art der Ausführung des Programms auf zwei Arten erfolgen:

- bei Ausführung als Application, also als lokal auf dem Rechner installierte Anwendung, erfolgt der Zugriff immer nur auf Bilddaten, die auch lokal zur Verfügung

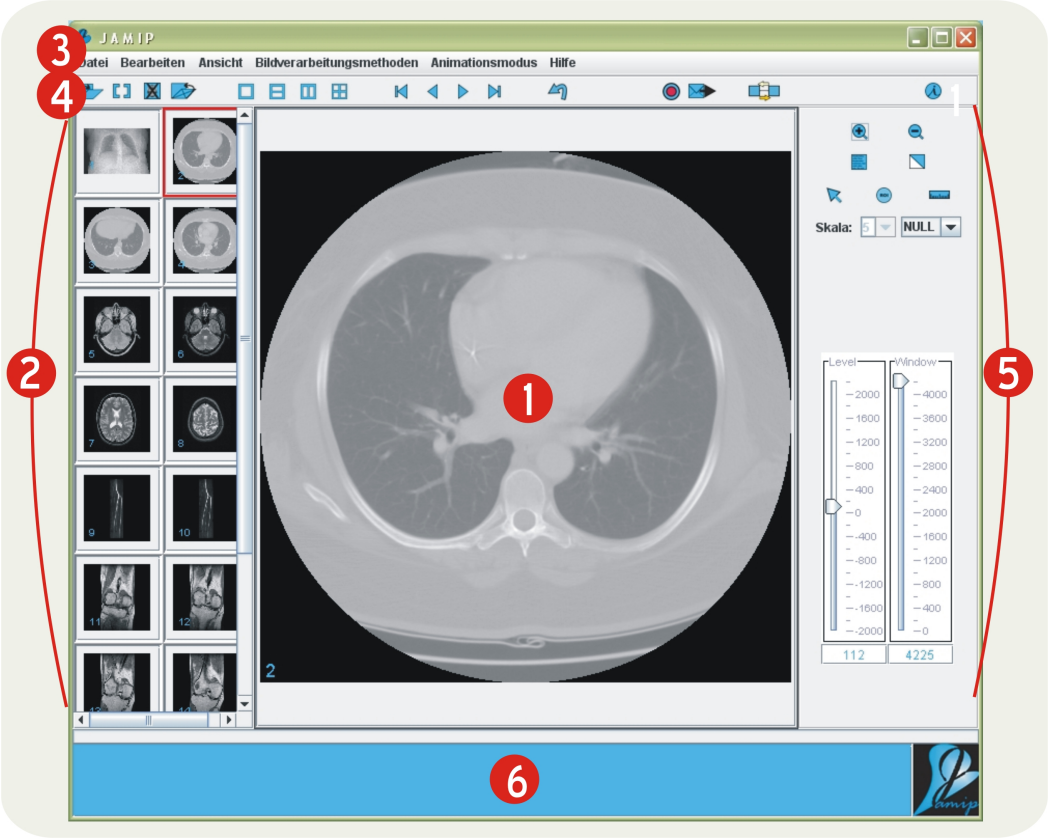


Abbildung 7.3: Die Benutzeroberfläche von JAMIP

stehen, das heißt jeder Nutzer des Programms kann immer nur auf die eigenen Dateien zugreifen, es können also nur lokal vorhandene bzw. lokal zugreifbare (also z. B. in einem Netzwerk verbundene) Bilddaten angezeigt oder zum Versand ausgewählt werden.

- bei Ausführung als Applet, kann der Nutzer sowohl auf seine lokal vorliegenden Bilddaten zugreifen, als auch auf die des Servers, von dem das Applet geladen wurde.

Im Folgenden wird nun die Abbildung 7.3 und damit der Aufbau der Benutzeroberfläche beschrieben:

- ❶ In der Mitte befindet sich das *Hauptfenster* des Tools, das die eingeladenen Bilddaten in Originalgröße anzeigt. Wahlweise können hier ein Bild, zwei Bilder vertikal nebeneinander oder horizontal übereinander oder auch vier Bilder angezeigt werden. Das (erste) Bild des Hauptfensters ist das sogenannte *aktuelle* Bild, auf dem

die ausgewählten Bildverarbeitungsroutinen und Standardoperationen ausgeführt werden.

- ❷ Die *Übersichtsanzeige* zeigt in einer vertikalen Übersichtsleiste alle in das Programm geladenen Bilder klein an. Durch Anklicken wird ein Bild im Hauptfenster in Originalgröße dargestellt. Das Bild, was auch im Hauptfenster zu sehen ist, wird in der Übersichtsanzeige rot umrandet dargestellt. Werden im Hauptfenster mehrere Bilder angezeigt, so ist in der Übersichtsanzeige das erste rot umrandet und die übrigen sind blau umrandet.

- ❸ Über die *Menüleiste* des Systems kann das Einladen und Bearbeiten der Bilder gesteuert werden. Eingeteilt ist sie in folgende Kategorien:

  – *Datei:* Über diesen Menüpunkt lassen sich Bilder öffnen, einfügen, schließen, importieren, exportieren oder versenden. Des Weiteren kann das Programm hierüber zum Datenempfang und -versand angemeldet sowie geschlossen werden. Das Programm ist im Wesentlichen für DICOM-Daten konzipiert, dennoch lassen sich auch JPG- und PNG-Dateien mit JAMIP öffnen.

  – *Bearbeiten:* Unter diesem Punkt verbergen sich Optionen zum Ändern der Bilder mittels Zoom oder Invertierung und zum Rückgängigmachen von Bildverarbeitungsoperationen. Außerdem erlaubt der Unterpunkt *Einstellungen* die Auswahl bestimmter Parameter für einzelne Funktionen und Bereiche, so kann hier beispielsweise die Art des Verschlüsselungsverfahrens festgelegt oder eine Berechnungsmethode für den Zoom ausgewählt werden.

  – *Ansicht:* Hier lässt sich die Headerinformation eines DICOM-Bildes anzeigen sowie die oben beschriebene Aufteilung des Hauptfensters vornehmen.

  – *Bildverarbeitungsmethoden:* Aufgeteilt nach *Methoden der Bildvorverarbeitung* und *Methoden der Bildanalyse* finden sich unter diesem Menüpunkt die in JAMIP integrierten Bildverarbeitungsmethoden. Einzelheiten zur Auswahl und zu den Verfahren werden in Abschnitt 7.3 erläutert.

  – *Animationsmodus:* Einige der ausgewählten Bildverarbeitungsmethoden können unter diesem Menüpunkt in ihrem Ablauf betrachtet werden.

  – *Hilfe:* Hierunter finden sich Informationen zur Programmversion sowie die mit JavaHelp realisierte (siehe dazu Abschnitt 6.5.1) Bedienungsanleitung des Systems, ein Beispiel ist in Abbildung 7.4 zu sehen.

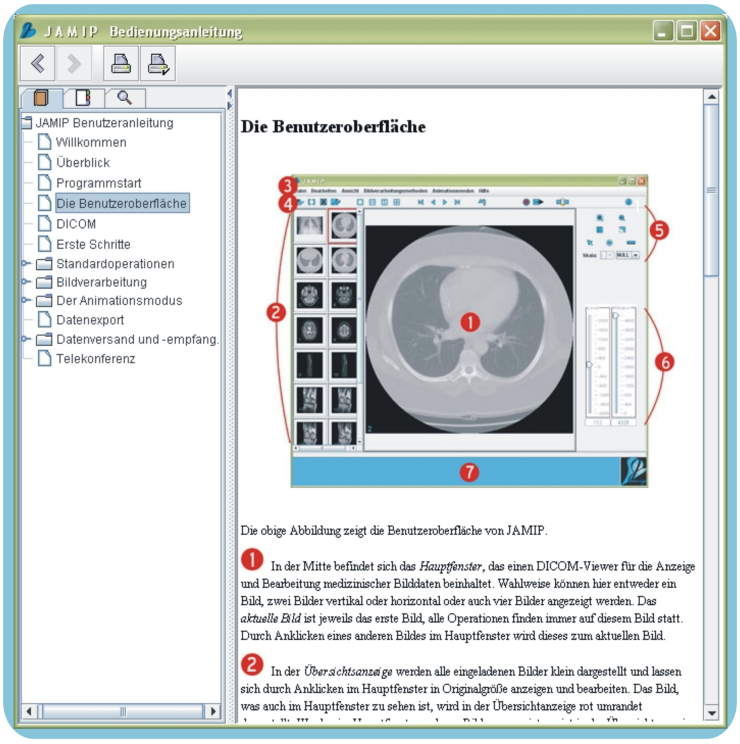


Abbildung 7.4: Die Bedienungsanleitung von JAMIP

- 4 Alternativ lässt sich ein Teil der Programmfunktionen über die Buttons der *Buttonleiste* steuern:
  - , , , Über diese Buttons lassen sich Bilddateien öffnen, einfügen, einzeln oder gesamt schließen.
  - , , , Die Aufteilung des Hauptfenster in ein Bild, zwei Bilder vertikal oder horizontal oder auch vier Bilder erfolgt mit diesen Buttons.
  - , , , Diese Buttongruppe ermöglicht das Durchsteppen durch die in der Übersichtsleiste angezeigten eingeladenen Bilddateien. Es wird also wahlweise das letzte, das vorherige, das nächste oder das erste Bild der Übersichtsleiste groß im Hauptfenster angezeigt.
  - Durch Drücken dieses Buttons wird die letzte Bildverarbeitungsoperation, die auf dem aktuellen Bild des Hauptfensters stattgefunden hat, wieder rückgängig gemacht.
  - , Diese Buttons dienen der Steuerung des Datenversandes. Durch Drücken des ersten Knopfes meldet sich das Programm zum Datenversand- und -empfang an. Sollen Bilddaten versendet werden, muss der zweite Knopf

gedrückt werden. Mehr zum Datenversand ist dem Abschnitt 7.4 zu entnehmen.

  - Hiermit wird das Telekonferenzmodul JAMIPTele gestartet.
  - Hinter diesem Button verbergen sich Informationen zur Programmversion.

- 5 Die obere Buttongruppe und die beiden Schieberegler repräsentieren die *Standardfunktionen* eines Telemedizinsystems; sie sind im folgenden Abschnitt 7.2 näher beschrieben.

- 6 In dem *Ausgabefenster* werden Fehlermeldungen angezeigt.

7.2 Die Realisierung der Standardfunktionen

Zu den Standardoperationen einer radiologischen Workstation oder eines Telemedizinsystems in Bezug auf die Bildbetrachtung zählen neben dem Einzel-, Doppel- oder Sequenzbildmodus, die Bildvergrößerung und -verkleinerung, die Invertierung der Farb- bzw. Grauwerte eines Bildes, die Fensterung eines Bildes mittels der Level-Window-Operation, das Hineinlegen eines Zentimetermaßes, die Abstandsmessung und die Mittelwertermittlung in einer ROI. Bei DICOM-Daten gehört zusätzlich noch die Anzeige der Headerinformationen zu den standardmäßig verfügbaren Funktionen. Für JAMIP wurden alle diese Operationen realisiert, sie sind über die Menüleiste oder über die Buttonleiste rechts neben dem Hauptfenster (siehe Abbildung 7.3) steuerbar.
Die ausgewählte Operation wird immer auf dem sogenannten *aktuellen Bild* des Hauptfensters durchgeführt; befinden sich mehrere Bilder im Hauptfenster, so ist dies immer das erste Bild links oben, es sei denn, eines der anderen Hauptfensterbilder wird zuvor durch Anklicken seines Rahmens, wodurch dieser leicht hervorgehoben dargestellt wird, ausgewählt und so zum aktuellen Bild. Die in einem Bild ausgeführten Operationen bleiben erhalten, auch wenn das Bild nicht mehr im Hauptfenster zu sehen ist. Ein Bild wird also immer mit den zuletzt auf ihm ausgeführten Operationen in der Großbildanzeige dargestellt. Alle weiteren Operationen werden auf dem aktuell angezeigten Bild ausgeführt. Ist dieses Bild also beispielsweise bereits invertiert, so wird die anschließende Operation für das invertierte Bild berechnet. Nachfolgend werden diese Funktionen in ihrer Realisierung für JAMIP kurz beschrieben, genauere Angaben können der Bedienungsanleitung für JAMIP entnommen werden.

Zoom Mit den Buttons und oder über die Menüleiste *Bearbeiten -> Zoom* wird das im Hauptfenster angezeigte Bild um den Faktor 0,1 vergrößert oder verkleinert. Des Weiteren kann im Menü unter *Bearbeiten -> Einstellungen -> Zoom* die Berechnungsart für den Zoom eingestellt werden. Ausgewählt werden kann zwischen einem Verfahren der bilinearen Interpolation, bei dem die Bildkonturen etwas verwischen, und dem standardmäßig eingestellten "Nächster-Nachbar"-Verfahren, was bei starker Vergrößerung die Pixelstruktur erkennen lässt.

Anzeige von Headerinformationen Über den Button oder über das Menü unter *Ansicht -> Bildzusatzinformationen* können zu einem DICOM-Bild ausgewählte Informationen aus dem Header angezeigt werden. In JAMIP werden hier neben Informationen zum Patient, zur Aufnahme und zum Aufnahmegerät auch Angaben über das Bild selbst angezeigt. Abbildung 7.5 zeigt ein Beispiel.

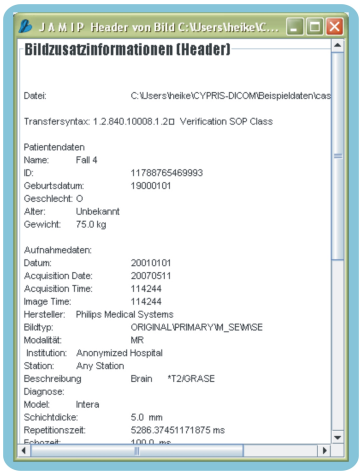


Abbildung 7.5: Ansicht eines Headers in JAMIP

Invertierung Mit dem Button oder über den Menüpunkt *Bearbeiten -> Invertieren* lässt sich das Bild invertieren. Bei einer Invertierung werden die Farben eines Bildes umgekehrt. Führt man die Invertierungsoperation ein zweites Mal aus, so erhält man wieder die Originalfarbwerte des Bildes.

Mittelwertermittlung Mit dem Button wird der Mittelwert in einer Region-Of-Interest (ROI) berechnet. Diese ROI wird durch Anklicken zweier Punkte im Bild bestimmt, die dann den Durchmesser eines Kreises bilden (siehe Abbildung 7.6, linkes Bild).

Berechnet wird der Mittelwert der Signalwerte dieses Kreises. Dabei wird immer auf die Originalinformation der Bilddaten und nicht auf die gefensterten Werte zurückgegriffen. Das Ergebnis der Berechnung wird direkt im Bild angezeigt.

Abstandsmessung Mit dem Button kann durch Anklicken zweier Punkte im Bild der Abstand zwischen diesen berechnet werden. Zwischen den Punkten wird eine Linie gezogen und das Ergebnis der Berechnung wird direkt im Bild angezeigt (Abbildung 7.6, mittleres Bild). Voraussetzung für das einwandfreie Funktionieren dieser Berechnung ist die entsprechende Angabe über den Abstand zwischen den einzelnen Pixeln im Header eines DICOM-Bildes; fehlt dieser Eintrag, so kann der Abstand nicht berechnet werden.

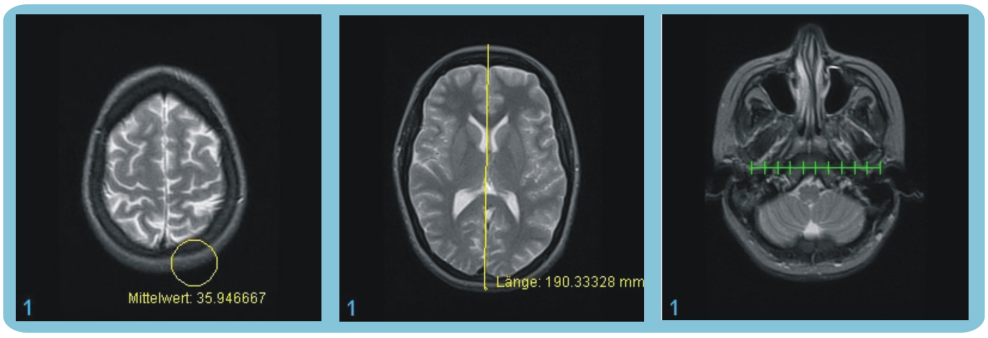


Abbildung 7.6: Beispiele für Abstand, Mittelwert und Skala in JAMIP

Skala Über diese Buttongruppe kann ein Zentimetermaß zum Hineinlegen in das Bild angezeigt werden. Dabei kann ausgewählt werden, ob das Maß oben, unten, rechts, links oder in der Mitte des Bildes erscheinen und ob es 3, 5, 10, 20 oder eine selbstgewählte Länge an Zentimetern haben soll (Beispiel in Abbildung 7.6, rechtes Bild). Auch hier muss für die Berechnung die entsprechende Angabe über den Abstand zwischen den einzelnen Pixeln im Header des DICOM-Bildes eingetragen sein.

Fensterung Als einfaches Beispiel für eine Bildvisualisierungstechnik (siehe Abschnitt 3.1.4) ist in JAMIP die in der Radiologie standardmäßig für die Anzeige von CT- oder MR-Daten verwendete *Fensterung* oder *Level-Window-Operation* integriert.
Dieses Verfahren wird eingesetzt, da die meisten in der Medizin mit digitalen Erzeugungsverfahren generierten Bilder eine Auflösung von 12 bit besitzen, was einer Grauwerttiefe von 4096 Signalwerten entspricht, die an herkömmlichen Monitoren mit nur 256 verschiedenen Grauwerten jedoch nicht geeignet dargestellt werden kann. Zur Vermeidung eines Informationsverlustes wird daher mit einer interaktiven Fensterung die Abbildung

der im Bild vorhandenen Signalwerte auf die zur Verfügung stehenden 256 Grauwerte manipuliert. Dafür wird zunächst innerhalb des Signalwertebereichs eine Intervallgröße, das sogenannte *Window*, und ein Intervallmittelpunkt, der *Level*, festgelegt. Anschließend werden die Signalwerte, die sich innerhalb des Intervalls befinden, linear auf die Grauwerte abgebildet, die übrigen Signalwerte werden auf die Werte 0 (schwarz) bzw. 255 (weiß) gesetzt. So lassen sich Gewebe mit Signalwerten in dem betrachteten Intervall kontrastreich visualisieren. Abbildung 7.7 zeigt ein CT-Bild bei unterschiedlicher Fensterung.

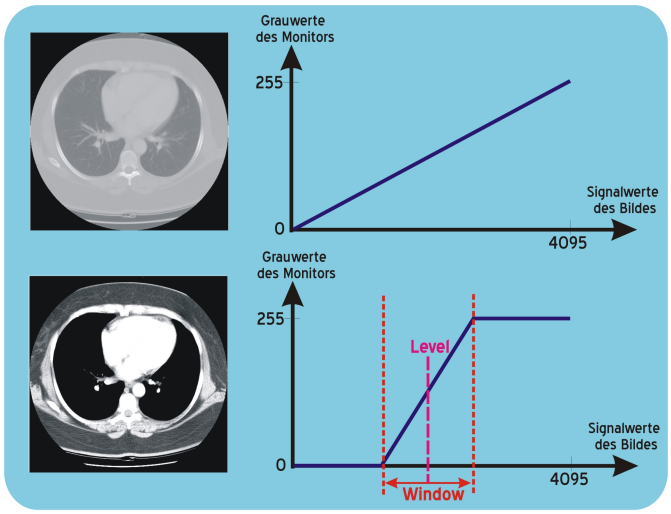


Abbildung 7.7: Die Fensterung/Level-Window-Operation

In JAMIP wird die Fensterung eines Bildes mit den beiden Schiebereglern, die sich rechts neben dem Hauptfenster befinden, durchgeführt (Abbildung 7.3). Der mit *Window* beschriftete Regler wird dabei für die Einstellung der Fensterbreite benutzt, er kann auf einer Skala zwischen Null und der Anzahl der im jeweiligen Bild enthaltenen Signalwerte hin- und hergeschoben werden. Der mit *Level* bezeichnete Schieberegler gibt den Punkt auf der Grauwerteskala des Bildes an, um den das Fenster gelegt werden soll. Die Werteskala dieses Schiebereglers beginnt immer bei dem kleinsten und endet mit dem größten in dem jeweiligen Bild vorkommenden Signalwert. Die Werte können alternativ auch per Hand in die Kästchen unter den Reglern eingetippt werden. Ausgeführt wird die Fensterung immer auf dem aktuellen Bild im Hauptfenster und gleichzeitig auf dem dazugehörigen Bild in der Übersichtsleiste. Alternativ kann im Menü unter *Bearbeiten -> Einstellungen -> Fensterung* eingestellt werden, ob die Fensterungsfunktion auf alle eingeladenen Bilder oder nur auf das aktuelle Bild angewandt werden soll.

7.3 Die Medizinische Bildverarbeitung in JAMIP

Die in JAMIP integrierten Bildverarbeitungsverfahren wurden für den Einsatz in der Präsenzlehre an der Universität zu Lübeck im Rahmen des Nebenfachstudiums der Medizinischen Informatik und für den Einsatz in dem Bmbf-geförderten Projekt "Multimediales Fernstudium Medizinische Informatik", welches in Zusammenarbeit mit der FernUni Hagen realisiert wurde, ausgewählt. Es handelt sich also um Methoden, die in der Präsenzlehre in der Vorlesung erklärt oder in der Online-Fassung des Studiengangs beschrieben werden. Speziell geht es dabei um Verfahren aus dem Bereich der Grundlagen der Medizinischen Bildverarbeitung. Mit JAMIP können diese Algorithmen ausprobiert oder auch vorgeführt werden und es kann der eigens dafür entwickelte Animationsmodus genutzt werden, um den genauen Ablauf eines Verfahrens zu verdeutlichen.
Wie in Abschnitt 3.1 definiert umfasst die Medizinische Bildverarbeitung, angefangen bei der Bilderzeugung bis hin zur Bildverwaltung, jegliche Anwendungsbereiche, die mit medizinischen Bilddaten zu tun haben. Die eigentliche Bearbeitung der Bilder findet jedoch in den Arbeitsschritten der Bildvorverarbeitung und Bildanalyse statt. In JAMIP sind grundlegende Algorithmen aus diesen Teilgebieten der Medizinischen Bildverarbeitung eingebunden; zu finden sind in der Menüleiste des Programms unter dem Punkt *Bildverarbeitungsmethoden* folgende:

- **Methoden der Bildvorverarbeitung**
  - Histogrammerstellung
  - Profilerzeugung
  - Grauwerttransformation
  - Filter
- **Methoden der Bildanalyse**
  - Schwellwertverfahren (Thresholding)
  - Bereichswachstumsverfahren (Regiongrowing)
  - Volumenwachstumsverfahren (Volumegrowing)
  - Histogrammbasierte Segmentierung
  - Aktive Konturmodelle - Snakes
  - K-Means-Clusteranalyse
  - Statistische Texturanalyse mit Cooccurrence-Matrizen

Das Ergebnis einer Bildverarbeitungsoperation wird standardmäßig übernommen, d. h. wird eine Operation wie z. B. ein Filter in einem Zusatzfenster ausgeführt, so wird das Ergebnis dieser Operation - in diesem Falle also das gefilterte Bild - an JAMIP zurückgegeben und das Ausgangsbild wird mit dem Ergebnisbild überschrieben. Eine Operation kann mit *Undo* (siehe Abbildung 7.3) rückgängig gemacht werden. Außerdem besteht die Möglichkeit in der Menüleiste unter *Bearbeiten -> Einstellungen -> Ergebnisrückgabe* auszuwählen, dass das Ergebnis nicht übernommen werden soll.

Nachfolgend werden die oben aufgeführten Methoden und Verfahren entsprechend dieser Aufzählungsreihenfolge und damit entsprechend ihres Auftretens in einer Bildverarbeitungskette sowie ihre Umsetzung in JAMIP näher beschrieben. Genauere Angaben zu den einzelnen Verfahren können [32, 33, 34] entnommen werden.

7.3.1 Histogrammerstellung

Die Erstellung eines Histogramms gehört zu den wesentlichen Elementen der Bildvorverarbeitung und ist Basis verschiedener weiterer Verfahren.

7.3.1.1 Theoretische Grundlagen zum Histogramm

Ein Histogramm ist eine diskrete Funktion, die die Häufigkeit und Verteilung der verschiedenen Grauwerte eines Bildes visualisiert. Über einer Achse, die den Wertebereich dieser Grauwerte darstellt, sind als Balken die einzelnen Häufigkeiten des Vorkommens des jeweiligen Wertes aufgetragen. Je höher der Balken über einem Wert ist, desto häufiger kommt dieser im Bild vor. So können die vorkommenden Grauwerte, der Kontrastumfang und die Helligkeit eines Bildes beurteilt werden.

7.3.1.2 Die Umsetzung des Histogramms in JAMIP

In JAMIP kann man sich das Histogramm eines Bildes durch Auswahl des Menüpunktes *Bildverarbeitungsmethoden -> Methoden der Bildvorverarbeitung -> Punktorientierte Verfahren -> Histogrammerstellung* darstellen lassen. Es erscheint in einem neuen, separaten Fenster. Standardmäßig ist immer zunächst das Histogramm des gesamten Bildes zu sehen, durch Aufziehen eines Rechtecks im Hauptfenster lässt sich aber auch das Histogramm einer so entstandenen kleinen ROI anzeigen; auch mehrere Histogramme verschiedener ROIs sind in einer Darstellung möglich. Durch Einstellung der unteren und oberen Grenze kann der Nutzer dann einen gewünschten Grauwertbereich auswählen, so dass z. B. vom Bildhintergrund herrührende Histogrammpeaks abgeschnitten werden und

so das Histogramm mit mehr Auflösung entlang der Y-Achse neu skaliert werden kann. Des Weiteren lässt sich ein Histogramm in JAMIP im normalen oder kumulativen Modus und als Balken- oder Liniendiagramm darstellen. Der Nutzer kann die Intervallgröße verändern und zwischen der Darstellung des Histogramms auf Basis der absoluten und auf Basis der relativen Anzahl der Grauwerte wechseln, ein Beispiel für ein Balkenhistogramm mit Intervallgröße 20 und mehreren in gelb, grün und blau eingefärbten ROIs ist in Abbildung 7.8 zu sehen.

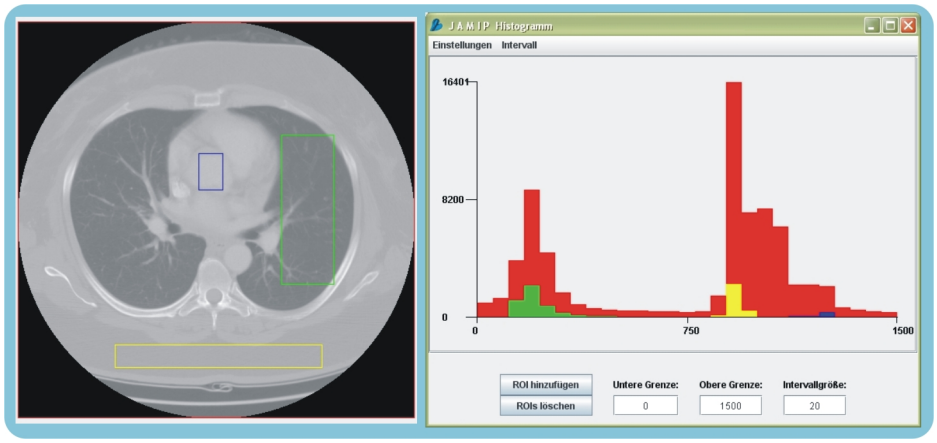


Abbildung 7.8: Das Histogramm in JAMIP; links: aktuelles Bild des Hauptfensters mit ausgewählten ROIs, rechts: Fenster mit mehreren farblich getrennten Histogrammen mit einer Intervallgröße von 20 und einem angepassten Grauwertbereich

7.3.1.3 Das Histogramm im Animationsmodus

Das Programm-Modul “Histogramm im Animationsmodus” wird durch Aufruf des Menüpunktes *Animationsmodus -> Histogramm* gestartet. Es erscheint ein neues, separates Fenster, wie Abbildung 7.9 zeigt bestehend aus einer Kopie des aktuellen Bildes neben dem Histogrammbereich. Der Nutzer wählt nun zunächst durch Aufziehen eines Rechtecks im Bild eine ROI aus, für die die Entstehung eines Histogramms dann angezeigt wird. Jeder Zwischenschritt entspricht dabei einem Pixel p der ROI, das in die Erstellung des Histogramms miteinbezogen wird. Dafür läuft während der Animation ein kleines gelbes Kästchen von links nach rechts und von oben nach unten über die ROI und markiert so das jeweils aktuelle Pixel, dessen Grauwert im Histogramm gezählt und durch einen roten Strich markiert wird. Beginnend mit einem anfangs leeren Histogrammfeld kann so beobachtet werden, wie schrittweise ein Histogramm entsteht.

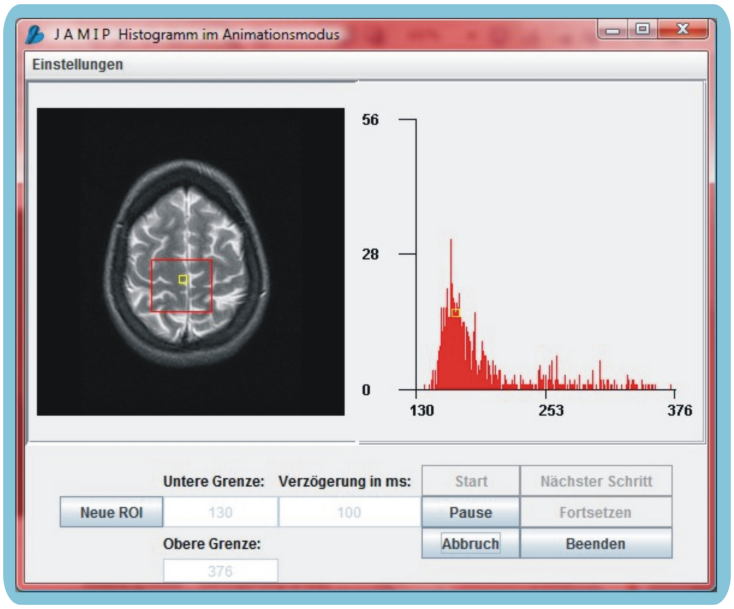


Abbildung 7.9: Die Histogrammerstellung im Animationsmodus

7.3.2 Profilerstellung

Ein Profil stellt einen Schnitt durch ein Bild entlang einer gegebenen Linie dar. Dabei kann als Linie eine Zeile oder Spalte des Bildes verwendet oder alternativ durch zwei frei im Bild gewählte Punkte definiert werden. Entlang dieser Linie werden dann die Grauwerte des aktuellen Bildpunktes ermittelt und als Funktion aufgetragen. Der so entstehende Plot zeigt den Grauwertverlauf des Bildes entlang dieser Linie.

7.3.2.1 Das Profil in JAMIP

Durch Auswahl des Menüpunktes *Bildverarbeitungsmethoden -> Methoden der Bildvorverarbeitung -> Punktorientierte Verfahren -> Profilerzeugung* wird dieses Modul in JAMIP gestartet, es gliedert sich in zwei nacheinander erscheinende Fenster. Über das erste Fenster (Abbildung 7.10 links) wird die Auswahl einer Linie im aktuellen Bild des Hauptfensters gesteuert, entlang derer dann im zweiten Fenster (Abbildung 7.10 rechts) das Profil als Linien- oder Balkendiagramm generiert wird.

7.3.3 Histogrammbasierte Grauwerttransformation

Medizinische Bilddaten, gerade CT-oder andere röntgenbasierte Aufnahmen, sind in der Praxis oft unterbelichtet, d. h. sehr dunkel und wenig kontrastreich. In solchen Bildern kommen viele Grauwerte der Grauwertskala überhaupt nicht vor. Um diese

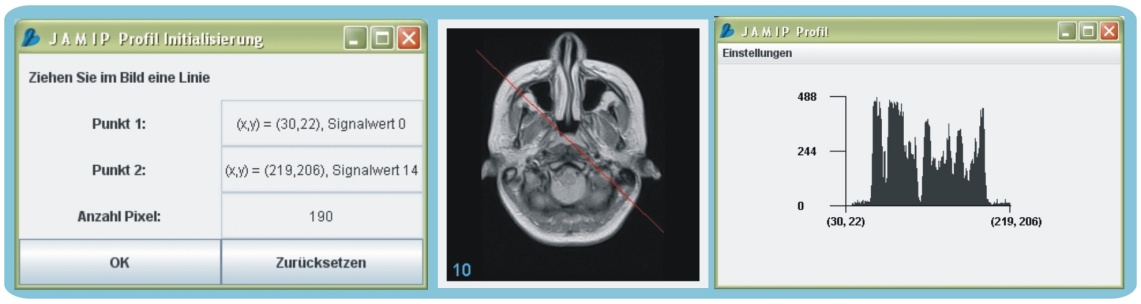


Abbildung 7.10: Erzeugung eines Profils in JAMIP; links: das Fenster zur Auswahl der Linie im Bild; Mitte: das aktuelle Bild des Hauptfensters mit der Profillinie; rechts: das Profil zur ausgewählten Linie

Aufnahmen zu optimieren, kann man eine lineare Dehnung bzw. *Histogrammspreizung* durchführen, wodurch eine Verbesserung des Kontrastes erreicht wird. Dieses Verfahren ist Histogramm-basiert und zählt zu den punktorientierten Verfahren der Bildvorverarbeitung.

7.3.3.1 Theoretische Grundlagen zur Grauwerttransformation

Bei dieser Operation wird der benutzte Grauwertbereich $g_{min}, ..., g_{max}$ auf den insgesamt zur Verfügung stehenden Grauwertbereich $G_{min}, ..., G_{max}$ abgebildet. Dies geschieht mit folgender Transformation, wobei mit $f(x,y)$ der Grauwert an der Stelle (x,y) im Originalbild gemeint ist:

$$T_{Histogrammspreizung}(x,y) = (G_{max} - G_{min}) * \frac{f(x,y) - g_{min}}{g_{max} - g_{min}}$$

Nicht immer erfolgt die Transformation linear, es können auch andere Transformationsfunktionen, z. B. quadratische, logarithmische oder exponentielle Funktionen, verwendet werden.

7.3.3.2 Die Umsetzung der Grauwerttransformation in JAMIP

Dieses Modul wird in JAMIP durch Aufruf des Menüpunktes *Bildverarbeitungsmethoden -> Methoden der Bildvorverarbeitung -> Punktorientierte Verfahren -> Grauwerttransformation* gestartet. Wie in Abbildung 7.11 zu sehen ist öffnet sich ein neues Fenster, das das originale neben dem transformierten Bild und darunter die zugehörigen Histogramme zeigt. Es ist möglich, die Transformation auf dem gesamten Bild oder innerhalb einer ROI anzuwenden, die im Ausgangsbild aufgezogen wird. Die ROI lässt sich dann über dem Originalbild hin- und herschieben, wobei die transformierte ROI des Ergebnisbildes

parallel mitverschoben wird. Weiterhin können die Intervallgrenzen für das Original- und Zielintervall verändert werden - initial werden diese beim Originalintervall auf das Minimum und Maximum der jeweiligen ROI und beim Zielintervall auf das Minimum und Maximum des Ausgangsbildes gesetzt. Auch besteht für die Transformation selbst die Auswahl zwischen der standardmäßig voreingestellten linearen, der quadratischen, der Quadratwurzel-, der Exponential- und der Logarithmusfunktion, wobei der Benutzer das Aussehen der Exponential- und Logarithmusfunktion durch Eingabe verschiedener Parameter verändern kann.

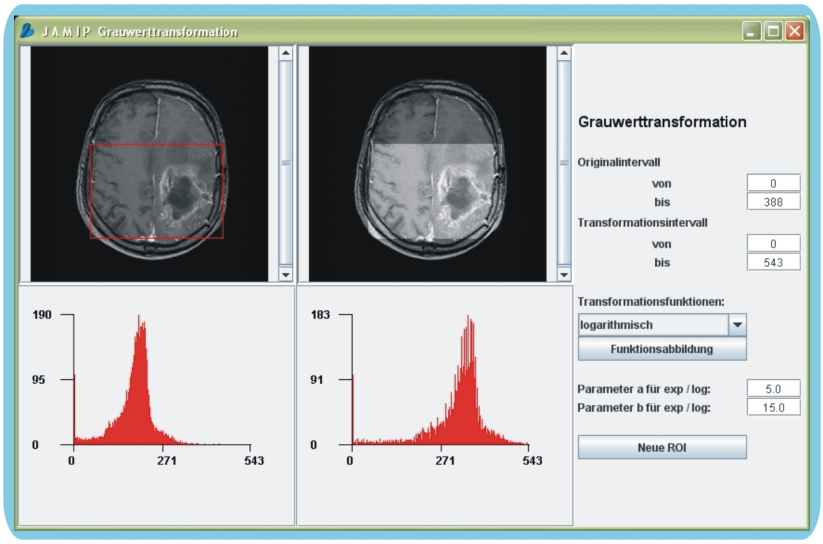


Abbildung 7.11: Die Grauwerttransformation; links oben: Originalbild mit verschiebbarer ROI, Mitte oben: Bild mit Transformationsergebnis in der ROI, unten: Histogramme der ROIs, rechte Seite: Parametereinstellung

7.3.4 Filter

Filter sind in der Regel lokale *lineare Operatoren*, für die es in der Bildvorverarbeitung mit der *Glättung* oder *Rauschunterdrückung* und der *Betonung von Kanten* zwei große Anwendungsgebiete gibt.

7.3.4.1 Theoretische Grundlagen zur Faltung

Filter lassen sich durch sogenannte *Masken* (engl. templates) beschreiben, deren Größe der zu betrachtenden Bildumgebung entspricht. Damit immer die Mitte des Fensters eindeutig ist, sind Maskengrößen im Allgemeinen ungerade, typische Größen sind: 3x3,

5x5 und 7x7. Das transformierte Bild T_{Filter} ergibt sich durch *Faltung* des Originalbildes

M1	M2	M3
M4	M5	M6
M7	M8	M9

Abbildung 7.12: Aufbau einer Maske: M1 bis M9 stehen hier für ganzzahlige Maskeneinträge, M4 ist der Maskeneintrag über dem zentralen Bildpunkt, der neu berechnet wird

f mit einer $(2m+1)x(2m+1)$-Maske g. Die allgemeine Gleichung dafür lautet:

$$T_{Filter}(x,y) = \sum_{i=-m}^{m} \sum_{j=-m}^{m} f(x+i, y+j) * g(i,j)$$

Praktisch gesehen ist das eine Aufsummierung der Multiplikationen der jeweils übereinanderliegenden Elemente der Bildmatrix mit denen der Maske. Ist die Belegung der Maske entsprechend gut gewählt, so können sich Effekte zur Nutzung in der Bildvorverarbeitung ergeben. So wird z. B. mit den in Abbildung 7.13 angegebenen 3x3-Masken das Bildrauschen reduziert. Ausgehend von der Annahme, Rauschpixel seien isolierte Pixel, die sich im Grauwert deutlich von ihrer Umgebung unterscheiden, wird durch Mittelung (Mittelwertfilter, links im Bild) bzw. durch von der Gauß'schen Normalverteilung abgeleiteten, gewichteten Mittelung (Gaußfilter, rechts im Bild) der Nachbarpunkte eines Pixels eine Bildglättung erreicht.

M = 1/9

1	1	1
1	1	1
1	1	1

G = 1/16

1	2	1
2	4	2
1	2	1

Abbildung 7.13: 3x3-Masken des Mittelwert- und des Gaußfilters

7.3.4.2 Umsetzung der Filter in JAMIP

In JAMIP sind sowohl Glättungs- als auch Kantenfilter eingebunden, sie finden sich im Menü unter *Bildverarbeitungsmethoden -> Methoden der Bildvorverarbeitung -> Lokale Operationen -> Filter*. Dabei gehören der Mittelwertfilter, der Gauß-3x3- und der Gauß-5x5-Filter zu den integrierten Glättungsfiltern, während mit dem Differenz-, dem symmetrischen Differenz-, dem Prewitt-, dem Sobel- und dem Laplace4- sowie dem Laplace8-Operator einige Beispiele für Kantenfilter in das System eingebunden wurden.

Zusätzlich kann der Benutzer mit einem Editor Filter frei definieren und anwenden, dieses Modul wird in Abschnitt 7.3.4.4 gesondert beschrieben.

Nach dem Start eines Filters erscheint das in Abbildung 7.14 zu sehende Fenster, das zum Vergleich links eine Kopie des aktuellen Bildes des Hauptfensters neben dem gefilterten Ergebnisbild auf der rechten Seite anzeigt. Über die Zoomleiste können parallel beide Bilder vergrößert werden. Des Weiteren lassen sich über den Menüpunkt *Einstellungen* die Randbehandlung auswählen, der Filter in den Bearbeitungsmodus übernehmen sowie die Einstellung und Anzeige eines Subfilters vornehmen.

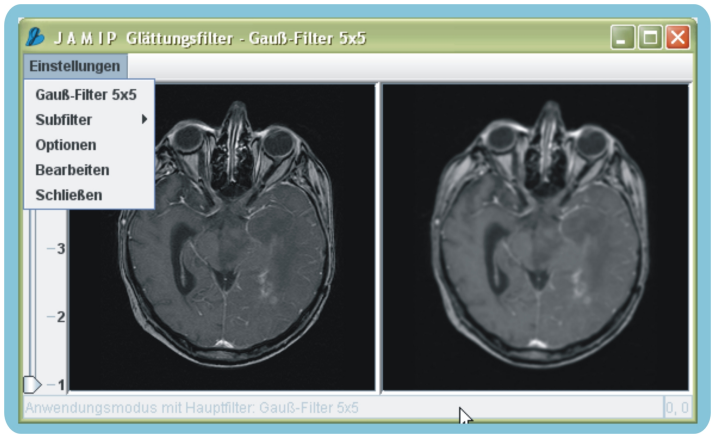


Abbildung 7.14: Das Filter-Modul in JAMIP bei Anwendung eines Gauß5x5-Filters

7.3.4.3 Die Filter im Animationsmodus

Der Animationsmodus für die Filter, der durch Aufruf des Menüpunktes *Animationsmodus -> Filter* nach Auswahl eines Kanten- oder Glättungsfilters startet, bietet mehrere Ansichten zur Verdeutlichung des Ablaufs der Faltung.

In der *Animationsansicht* wird der Ablauf der Faltung auf einem kleinen, zuvor vom Nutzer ausgewählten Bildausschnitt demonstriert. Es erscheint das in Abbildung 7.15 zu sehende Fenster, in dem oben links der Ausschnitt des Ausgangsbildes zu sehen ist, darunter erscheinen die zugehörigen numerischen Pixelwerte. Die Berechnung der Faltung wird in dem Kasten in der Mitte des Fensters angezeigt. Nach jedem Schritt werden die errechneten Werte unten rechts in die zu Beginn leere Pixelwertabbildung des Ergebnisbildes eingetragen, während oben rechts der passende Grauwert als Farbe erscheint. Zur Veranschaulichung der Berechnung der einzelnen Pixelwerte kann zwischen dem Step-

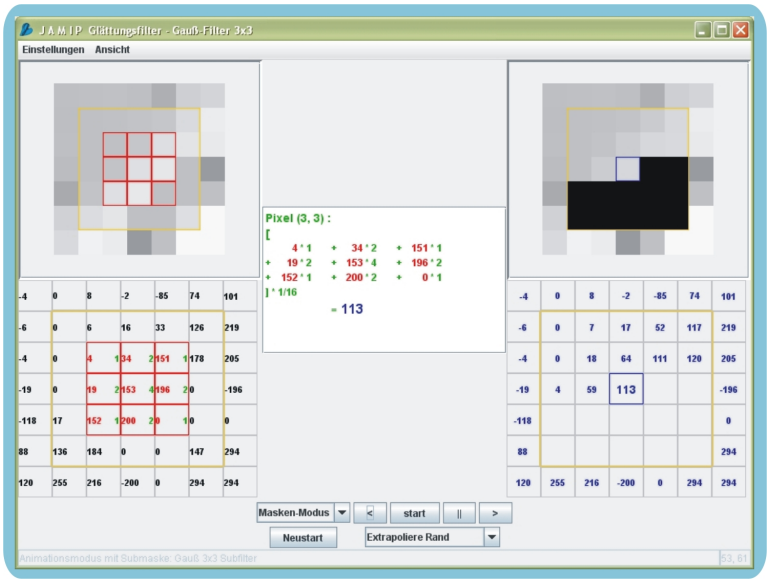


Abbildung 7.15: Der Animationsmodus bei der Filterung; links: vergrößerte ROI des Originalbildes und deren Pixeldaten, Mitte: Berechnung der Faltung, rechts: vergrößerte ROI des Ergebnisbildes und deren Pixeldaten

und dem Masken-Modus sowie zwischen automatischem Ablauf in drei verschiedenen Tempi und schrittweisem Verlauf per Hand umgeschaltet werden. Außerdem sind für die Randbehandlung die drei gängigsten Alternativen implementiert und dem Anwender zur Auswahl zur Verfügung gestellt.

Neben der Darstellung des animierten Ablaufs der Faltung erlauben die *Numerische*

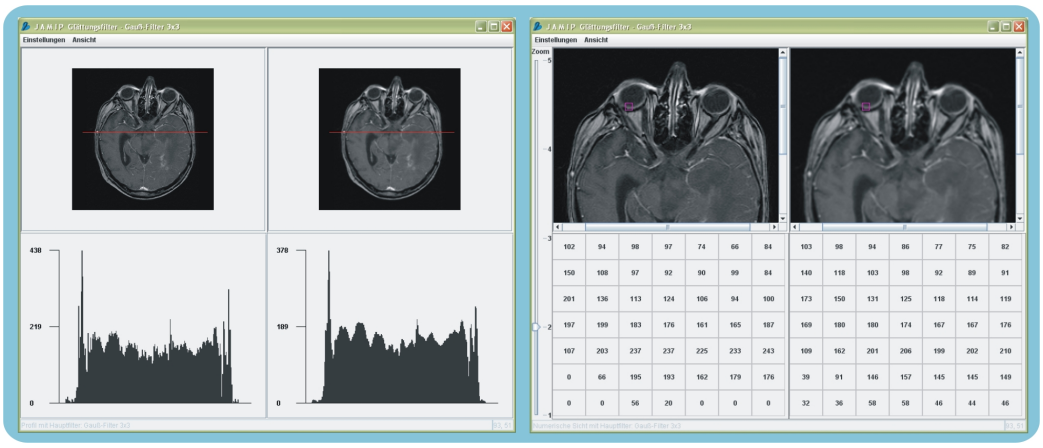


Abbildung 7.16: Profilansicht und Numerische Ansicht bei der Filterung

Ansicht und die *Profilansicht* einen Vergleich des Originals mit dem gefilterten Ergeb-

nisbild. Dafür kann sich der Nutzer, wie in Abbildung 7.16 rechts zu sehen ist, einmal die numerischen Pixelwerte in einer selbstgewählten ROI anzeigen und zum anderen die Profile anhand einer selbstdefinierten Linie erstellen lassen.

Gelegentlich ist es in der Medizinischen Bildverarbeitung sinnvoll, das Original- mit dem

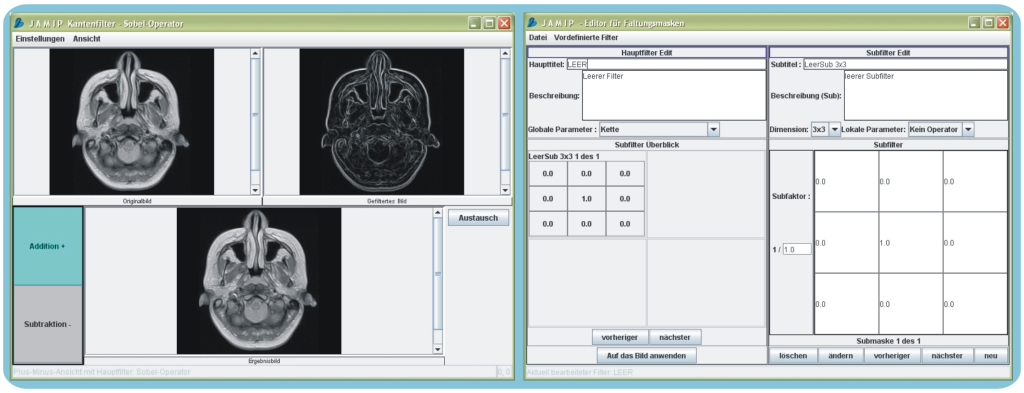


Abbildung 7.17: Weitere Ansichten bei der Filterung; links: Plus-Minus-Ansicht bei der Filterung, rechts: Eingabemaske für die Benutzerdefinierten Filter

gefilterten Bild mittels Addition oder Subtraktion zu verknüpfen; um ein kantenverstärktes Bild zu erhalten, muss z. B. das Ausgangsbild zu seinem kantengefilterten Ergebnisbild hinzuaddiert werden. Zu diesem Zweck findet sich in JAMIP die in Abbildung 7.17 links gezeigte *Plus-Minus-Ansicht*. Das Ergebnis so einer Addition oder Subtraktion kann in das Hauptprogramm von JAMIP übernommen werden.

7.3.4.4 Benutzerdefinierte Filter in JAMIP

Als weitere Besonderheit bietet JAMIP die Möglichkeit, Filter mittels eines Editors für Filtermasken frei zu definieren. Das Modul startet unter *Bildverarbeitungsmethoden -> Methoden der Bildvorverarbeitung -> Lokale Verfahren -> Filter -> Benutzerdefinierte Filter*. Bereits implementierte Filter können mit diesem Programmfeature in ihrem Aufbau betrachtet oder verändert werden, neue Filter können erstellt und für den späteren Gebrauch gespeichert werden. Dabei ist sowohl die Größe der Filtermasken als auch die Anzahl der hintereinander zu verwendenden Filter frei wählbar. Abbildung 7.17 zeigt rechts den Editor nach Aufruf dieses Moduls.

7.3.5 Schwellwertverfahren

Bei dem globalen Schwellwertverfahren handelt es sich um ein pixelorientiertes Segmentierungsverfahren, welches häufig zum Einsatz kommt, um z. B. ein Bildobjekt von dem Bildhintergrund zu trennen.

7.3.5.1 Theoretische Grundlagen zum Schwellwertverfahren

Nach Festlegung eines Schwellwertbereiches $[t_{min}, t_{max}]$ wird jedes Pixel aufgrund seines Grauwertes $f(x,y)$ einer von zwei Klassen zugeordnet, entweder liegt es innerhalb des Bereiches oder nicht. Das Ergebnis dieser Operation ist eine Binarisierung des Bildes, d. h. das Ergebnisbild $T_{Schwellwertverfahren}$ enthält nur noch die Werte 0 und 1. Formal lässt sich dies folgendermaßen beschreiben:

$$T_{Schwellwertverfahren}(x,y) = \begin{cases} 1 & : \quad t_{min} < f(x,y) \leq t_{max} \\ 0 & : \quad sonst \end{cases}$$

Da die Bildpunkte nur in Abhängigkeit ihres Grauwertes einer Klasse zugeordnet werden, müssen keine zusammenhängenden Regionen entstehen.

7.3.5.2 Das Schwellwertverfahren in JAMIP

In JAMIP wird das Verfahren durch Auswahl des Menüpunktes *Bildverarbeitungsmethoden -> Methoden der Bildanalyse -> Segmentierungsverfahren -> Schwellwertverfahren* gestartet. Es erscheint ein zusätzliches Fenster mit zwei Schiebereglern zur Einstellung des unteren bzw. oberen Grenzwertes (siehe Abbildung 7.18). Das Ergebnis des Schwell-

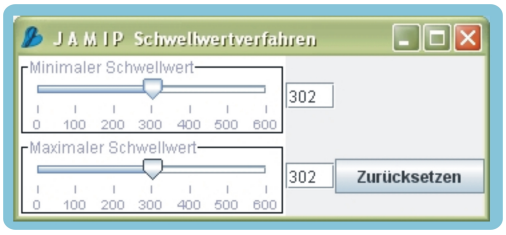


Abbildung 7.18: Das Steuerungsfenster für das Schwellwertverfahren

wertverfahrens wird im aktuellen Bild des Hauptfensters dargestellt. Zu Beginn sind beide Regler auf den mittleren Grauwert des Bildes voreingestellt. Der genaue Wert kann den Textfeldern rechts neben den Reglern entnommen werden. Durch Verschieben der Regler werden minimaler und maximaler Grenzwert verändert; außerdem können diese Werte über die Textfelder genau eingegeben werden. Der Button `Zurücksetzen` setzt das

aktuelle Bild des Hauptfensters wieder auf das Ausgangsbild zurück. Abbildung 7.26 zeigt ganz links ein Beispiel für die Anwendung des Schwellwertverfahrens in JAMIP, das hier die Trennung der Lungen vom restlichen Objekt ermöglicht.

7.3.6 Bereichswachstumsverfahren (Regiongrowing)

Mit dem *Bereichswachstumsverfahren* oder *Regiongrowing* wurde in JAMIP ein weiteres in der Medizinischen Bildverarbeitung häufig zur Segmentierung einzelner Gewebe genutztes Verfahren eingebunden.

7.3.6.1 Theoretische Grundlagen zum Bereichswachstumsverfahren

Das Bereichswachstumsverfahren ist ein halbautomatisches Bildsegmentierungsverfahren, bei dem ausgehend von einem vom Benutzer gewählten Saatpunkt Pixel immer dann zu dem Segment hinzugefügt werden, wenn sie einem sogenannten Homogenitätskriterium genügen. Das Verfahren verfolgt den nachbarschaftsorientierten Ansatz, d. h. erfüllt ein Punkt das Homogenitätskriterium und gehört damit zum Segment, so werden auch seine Nachbarn auf Homogenität überprüft und gegebenenfalls dem Segment zugeordnet. So entsteht anders als beim Schwellwertverfahren immer eine zusammenhängende Region. Das Homogenitätskriterium ist dabei als Ähnlichkeitsmaß zu verstehen; es gibt an, wie ähnlich sich die Pixel eines Segmentes sein sollen. Ein einfaches Homogenitätskriterium ist z. B. die Angabe eines Grauwertintervalls $[g_{min}, g_{max}]$, in dessen Grenzen sich das Segment befinden soll. Ist $f(x,y)$ der Grauwert des Pixels an der Stelle (x,y), dann wird das Kriterium formal für alle Pixel, die betrachtet werden, beschrieben durch:

$$T_{Regiongrowing}(x,y) = \begin{cases} (x,y) \in Segment & : \quad g_{min} \leq f(x,y) \leq g_{max} \\ (x,y) \notin Segment & : \quad sonst \end{cases}$$

Die Qualität eines Segmentierungsergebnisses mittels des Bereichswachstumsverfahrens hängt in der Praxis stark von der Wahl und Einstellung des Homogenitätskriteriums sowie von dem Saatpunkt ab.

7.3.6.2 Die Umsetzung des Bereichswachstumsverfahrens in JAMIP

Nach Aufruf des Menüpunktes *Bildverarbeitungsmethoden -> Methoden der Bildanalyse -> Segmentierungsverfahren -> Bereichswachstumsverfahren (Regiongrowing)* erscheint in Abbildung 7.19 zu sehendes Fenster, mit dem das Bereichswachstumsverfahren in JAMIP gesteuert wird. Dafür wählt der Benutzer zunächst nach Drücken des Buttons `Saatpunkte setzen` im aktuellen Bild des Hauptfensters durch Anklicken einen

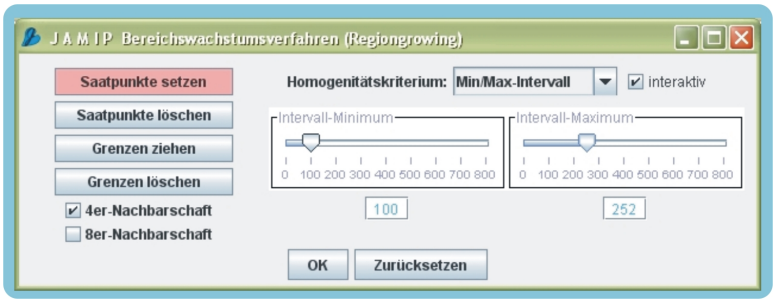


Abbildung 7.19: Das Steuerungsfenster für das Bereichswachstumsverfahren

oder mehrere Saatpunkte aus. Es stehen dann drei verschiedene Homogenitätskriterien und damit Möglichkeiten zur Intervalldefinition zur Auswahl: Zum einen können mit `Min/Max-Intervall` der minimale und der maximale Grauwert über Schieberegler direkt eingestellt werden, zum anderen lassen sich mit `auto. Min/Max` die Intervallgrenzen unter Berücksichtigung der Standardabweichung σ und des Mittelwerts m der Saatpunkte und ihrer jeweiligen Nachbarn berechnen. Die Intervallgrenzen sind dabei folgendermaßen definiert: $i_{min} = m - s \cdot \sigma$ und $i_{max} = m + s \cdot \sigma$, wobei mit s ein vom Nutzer anzugebender Skalierungsfaktor gemeint ist. Die dritte Variante `Euklidische Distanz` verwendet selbige, d. h. für die Bildpunkte des Segments gilt: $\sqrt{(G(x,y)-m)^2} < r$, wobei r ein einzugebender Radius ist und m den Mittelwert des Saatpunktes und seiner Nachbarpunkte bezeichnet.
Hat der Nutzer eines der Homogenitätskriterien ausgewählt, standardmäßig ist das Min/Max-Intervall voreingestellt, so kann er sich durch Drücken des `OK`-Buttons das Ergebnis im Bild anzeigen lassen. Die Saatpunkte erscheinen dabei als gelbe Kreuze, während das Segment selbst rot und der Rand des Segments grün eingefärbt werden. Ein Beispiel für die Segmentierung eines Lungenflügels ist in Abbildung 7.26 zu sehen. Für eine optimale Nutzung und Steuerung des Verfahrens ist es in JAMIP möglich, Grenzen zu ziehen und wieder zu löschen, zwischen der Betrachtung der 4er- oder 8er-Nachbarschaft zu wählen und gleichzeitig mit der Änderung eines Homogenitätskriteriums das Wachsen eines Segments zu beobachten.

7.3.6.3 Das Bereichswachstumsverfahren im Animationsmodus

Im Animationsmodus für das Bereichswachstumsverfahren - aufrufbar unter *Animationsmodus -> Bereichswachstumsverfahren* erscheint ein Extrafenster, in dem eine Kopie des aktuellen Bildes neben einem vergrößerten Bildbereich des vom Benutzer definierten Saatpunktes dargestellt wird. Nach Auswahl und Einstellung eines Homogenitätskriteriums

wird der Ablauf des Algorithmus' schrittweise durch eine Betrachtung der einzelnen Pixel und ihrer jeweiligen Zuordnung ausgeführt. Dabei werden alle Pixel, die das Kriterium erfüllen, rot und ihre jeweils noch zu betrachtenden Nachbarn hellblau eingefärbt. Pixel, die schon betrachtet wurden und nicht zum Segment gehören, werden grün markiert und bilden so den Rand. Der Animationsmodus und weitere Details der Einfärbung sind in Abbildung 7.20 zu sehen. Nach Beendigung des Verfahrens ist das gefundene Segment rot mit grünem Rand sowohl im Bild als auch im vergrößerten Bildbereich zu sehen. Zur Veranschaulichung des Ablaufs lässt sich der Regiongrowingalgorithmus im Animationsmodus automatisch in drei verschiedenen Schnelligkeitsstufen oder schrittweise per Hand ausführen. Des Weiteren ist die Größe des Bildausschnittes auswählbar, außerdem können statt der Farb- die numerischen Werte angezeigt werden.

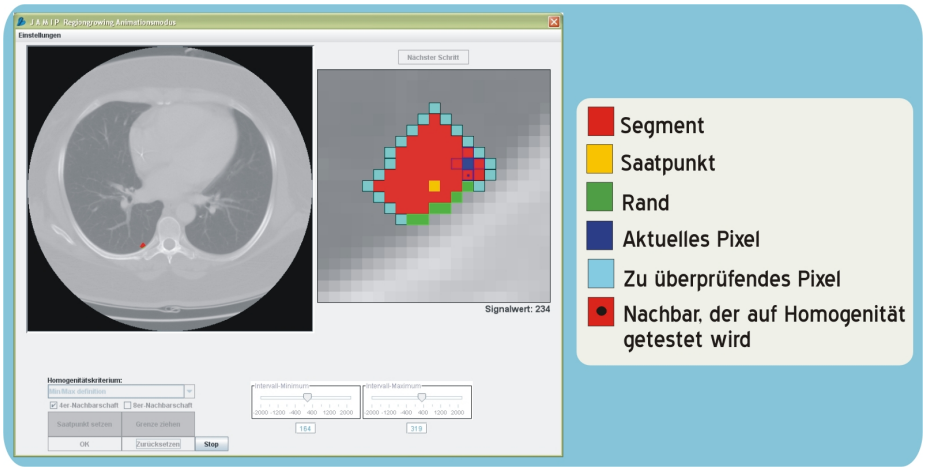


Abbildung 7.20: Der Animationsmodus beim Bereichswachstumsverfahren; links: Bild mit benutzerdefiniertem Saatpunkt, rechts: vergrößerter Saatpunktbereich, ganz rechts: Einfärbung der Pixel

7.3.6.4 Das Volumenwachstumsverfahren

Bei diesem Verfahren handelt sich um die 3D-Variante des Bereichswachstumsverfahrens, die durch die Betrachtung räumlicher Nachbarschaften erreicht wird. Grundlage zur Ausführung dieses Moduls sollte daher eine Schichtaufnahme sein. Eingesetzt wird das Volumenwachstumsverfahren in erster Linie zur 3D-Segmentierung medizinischer Bildobjekte mit dem Ziel der 3D-Visualisierung oder der Volumenbestimmung.
Das Verfahren wird durch Auswahl des Menüpunktes *Bildverarbeitungsmethoden -> Methoden der Bildanalyse -> Segmentierungsverfahren -> Volumenwachstumsverfahren (Volumegrowing)* gestartet. Auch hier erscheint ein zusätzliches Fenster zur Steuerung

des Verfahrens, was sich hauptsächlich darin von dem Steuerungsfenster für das Bereichswachstumsverfahren unterscheidet, dass hier zusätzlich die Nummern der Bilder auswählbar sind, die in die Berechnung miteinbezogen werden sollen; standardmäßig werden zunächst alle in das System geladenen Bilder für das Verfahren ausgewählt.
Das Ergebnis der Berechnung wird in den markierten Bildern als rotes Segment eingefärbt. Ein Beispiel für ein mit JAMIP durchgeführtes Volumenwachstumsverfahren auf einer CT-Schichtbildserie des Oberkörpers ist in Abbildung 7.21 zu sehen.

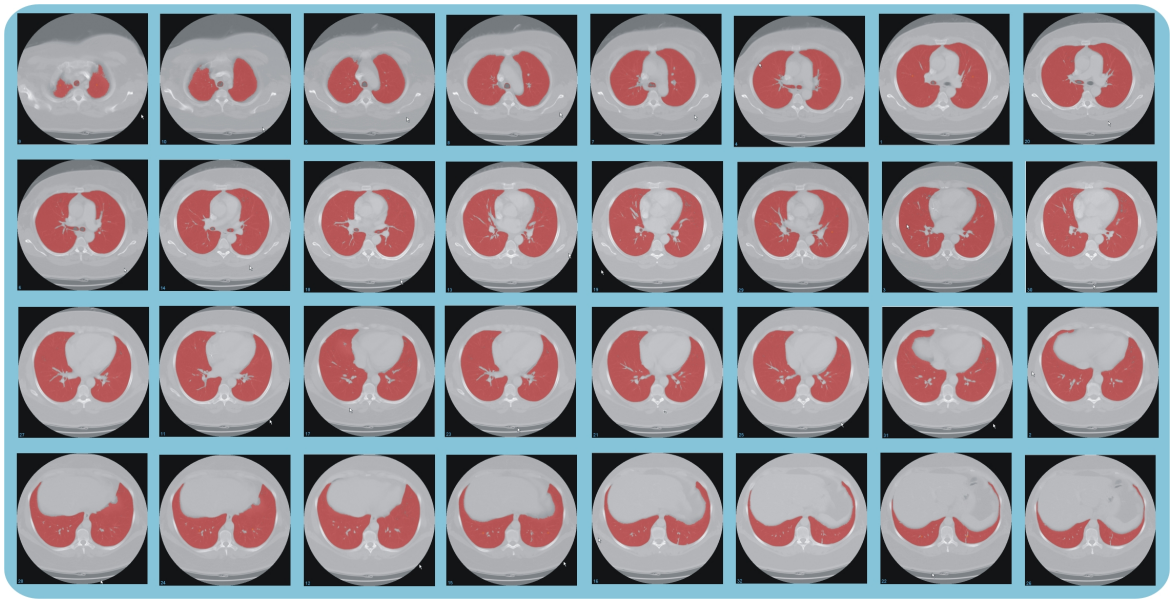


Abbildung 7.21: Segmentierungsergebnis eines Volumenwachstumsverfahrens auf einer CT-Serie

7.3.7 Histogrammbasierte Segmentierung

Eine andere Möglichkeit zur Bildsegmentierung bietet sich dem Anwender in JAMIP mit der histogrammbasierten Segmentierung.

7.3.7.1 Theoretische Grundlagen zur histogrammbasierten Segmentierung

Ziel der histogrammbasierten Segmentierung ist die Trennung verschiedener Strukturen unter Zuhilfenahme des Histogramms, also der Verteilung der Grauwerte des Bildes. Die Voraussetzung für ein gelungenes Ergebnis ist hier, dass unterschiedliche Strukturen auch unterschiedliche Grauwerte aufweisen und somit als voneinander abgrenzbare Anhäufungen im Histogramm wiederzufinden sind. Im einfachsten Fall ergibt sich so die Trennung eines Bildobjektes vom umgebenden Bildhintergrund, im komplexeren Fall

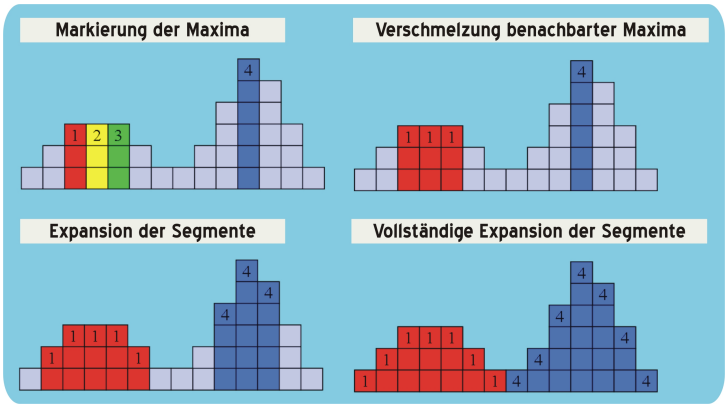


Abbildung 7.22: Ablauf der histogrammbasierten Segmentierung

z. B. die Abgrenzung verschiedener Organe oder Organstrukturen voneinander. Weisen verschiedene Strukturen ähnliche oder gleiche Grauwerte auf, so ist dies Verfahren zur Segmentierung nicht geeignet.

Abbildung 7.22 zeigt eine Illustration des Algorithmus' für die histogrammbasierte Segmentierung. In dem Histogramm H des Bildes werden alle Maxima gesucht. Jedes neue Maximum entspricht dabei zunächst einem neuen Segment. Dann werden direkt benachbarte Segmente sukzessive miteinander verschmolzen, bevor schließlich die Expansion der verbliebenen Segmente erfolgt. Anschließend wird das Ergebnis im Bild visualisiert, ein Beispiel ist in Abbildung 7.26 zu sehen.

7.3.7.2 Die Umsetzung der histogrammbasierten Segmentierung in JAMIP

Auch für das Modul der histogrammbasierten Segmentierung erscheint in JAMIP nach Auswahl des Menüpunktes *Bildverarbeitungsmethoden -> Methoden der Bildanalyse -> Segmentierungsverfahren -> Histogrammbasierte Segmentierung* ein neues Fenster, in dem das Histogramm des gesamten aktuellen Bildes oder einer benutzerdefinierten ROI hieraus mit den gefundenen eingefärbten Segmenten angezeigt wird. Optional kann eine Liste der Segmente und ihrer jeweiligen absoluten und relativen Größe sowie ihrer Grauwertbereiche aufgerufen werden. Das Segmentierungsergebnis selbst wird direkt im aktuellen Bild angezeigt, d. h. die Pixel des Bildes werden farblich entsprechend ihrer Zugehörigkeit zu einem der im Histogramm gefundenen Cluster eingefärbt (siehe Abbildung 7.23). Für die Farbverteilung können die Segmente wahlweise nach ihrer Größe oder ihrem Grauwertbereich sortiert werden.
Um in der Praxis ein möglichst optimales Ergebnis zu erzielen, können obere und untere

Grenze des zu betrachtenden Histogramms angepasst werden, ebenso kann der Nutzer die Anzahl der Segmente begrenzen und das Histogramm mittels Eingabe eines Wertes für den gleitenden Mittelwert glätten, um eine Übersegmentierung zu vermeiden.

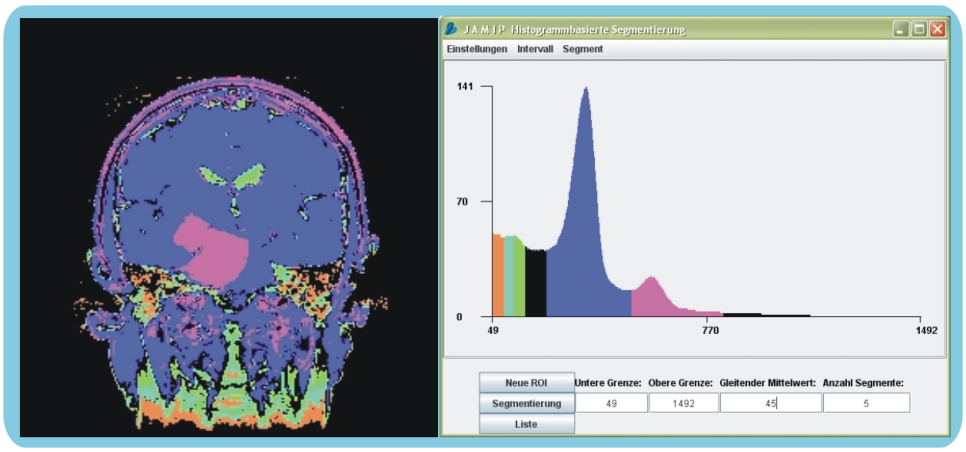


Abbildung 7.23: Die histogrammbasierte Segmentierung; links: segmentiertes Ergebnisbild, rechts: Segmente im Histogramm

7.3.8 Aktive Konturmodelle - Snakes

Snakes, auch *Aktive Konturen* genannt, sind ein Konzept, das zu den Segmentierungsalgorithmen gezählt wird. In der medizinischen Bildverarbeitung werden Snake-Algorithmen verwendet, um Objektkonturen zu bestimmen, so zum Beispiel in der Diagnostik bei Ultraschallaufnahmen.

7.3.8.1 Theoretische Grundlagen zu Snakes

Bei Snakes handelt es sich um deformierbare Kurven, die ihre Form unter Einbeziehung der vorliegenden Bilddaten verändern. Sie benutzen dafür die sogenannte *interne Energie* der Kontur selbst, die während der Deformation für Glattheit und Stetigkeit sorgt, und die *externe Energie*, die aus den Bilddaten berechnet wird und sich auf Objekte im Bild zubewegt. Aktive Konturen besitzen eine gewisse Unempfindlichkeit gegenüber Rauschen und Lücken in Objektgrenzen. Erstmal beschrieben wurden sie von [69], ein Beispiel für eine mit JAMIP gefundene Objektkontur ist in Abbildung 7.26 zu sehen.

7.3.8.2 Umsetzung der Snakes in JAMIP

Abbildung 7.24 zeigt das nach Aufruf des Menüpunktes *Bildverarbeitungsmethoden -> Methoden der Bildanalyse -> Segmentierungsverfahren -> Aktive Konturmodelle - Snakes*

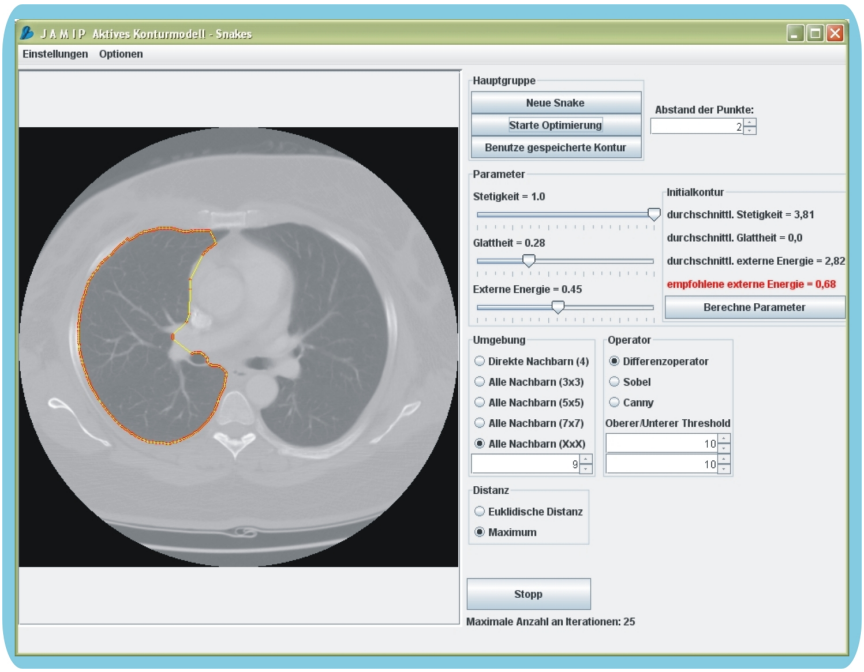


Abbildung 7.24: Die Snakes in JAMIP

erscheinende Fenster. Links ist eine Kopie des aktuellen Bildes des Hauptfensters zu sehen, während sich auf der rechten Seite die Steuerungsknöpfe und Schieberegler für das Modul befinden. Für die Segmentierung mit aktiven Konturen muss zunächst manuell eine Kontur in dem Bild initialisiert werden, die sich möglichst nahe an der zu segmentierenden Struktur befindet. Ausgehend davon erfolgt dann die automatische Definition der Kontur. Das Ergebnis hängt stark von der Güte der Initialisierung ab; generell gilt dabei, je näher sich die Initialkontur an der Objetktgrenze befindet, um so besser ist das Ergebnis. Für die Optimierung der Snake stehen dem Nutzer dabei in JAMIP eine Reihe von Parametern zur Verfügung, neben den Energiewerten - Stetigkeit, Glattheit und externe Energie - können noch der Abstand der Punkte, die zu betrachtende Umgebungsgröße, der Operator zur Bestimmung der externen Energie und die Anzahl der Iterationen eingestellt werden. Die Veränderung der Snake wird standardmäßig animiert dargestellt, d. h. der Anwender kann zusehen wie sich die Initialkontur aufgrund der eingestellten Parameter dem gesuchten Objekt anpasst. Grundsätzlich wird ein Zusammenziehen der Snake verhindert, dies kann aber unter dem Menüpunkt *Einstellungen -> Schrumpfen der Snake verhindern* geändert werden.

7.3.9 K-Means-Clusteranalyse

Auch Clusteranalyseverfahren zählen zur Gruppe der Segmentierungsalgorithmen. Sie werden in der medizinischen Bildverarbeitung vor allem zur Analyse und Segmentierung multispektraler 2D- und 3D-Bilddaten eingesetzt.

7.3.9.1 Theoretische Grundlagen der K-Means-Clusteranalyse

Unter Clusteranalyse werden Verfahren verstanden, die Objekte in homogene, disjunkte Gruppen einteilen. Dabei sollen sich die Objekte einer Gruppe unter bestimmten Gesichtspunkten möglichst "ähnlich" sein, d. h. ähnliche Merkmalsvektoren haben. Die Gruppen werden Cluster genannt. Einzelne Bildpunkte eines jeden Clusters müssen dabei nicht unbedingt auch im Bildraum zusammenhängend liegen.
Bei iterativen, partitionierenden Clusterverfahren liegt jeder Bildpunkt in einem Cluster, es gibt somit keine leeren Cluster. Die Pixel werden anhand ihrer Eigenschaften iterativ einer Partition zugeordnet. Dabei werden für die eindeutige Zuweisung Distanzfunktionen benutzt, die sich auf qualitative Eigenschaften des Bildes beziehen. Eine solche Distanzfunktion kann z. B. die Euklidische Distanz sein.

Das *K-Means-Verfahren* als Beispiel für ein in JAMIP implementiertes Clusteranalyseverfahren geht von einer Initialpartition aus, welche in jedem Iterationsschritt des Verfahrens verbessert wird. Die Anzahl der Partitionen ist von vornherein festgelegt, ist sie zu klein, werden möglicherweise nicht alle Bildstrukturen gut genug ausgearbeitet, ist sie zu groß, so kann es zu einer Übersegmentierung kommen. Bei medizinischen Bilddaten haben sich die Clusterzahlen 5 und 6 bewährt. In Abbildung 7.26 findet sich für die Segmentierung mit der K-Means-Clusteranalyse in JAMIP ein Beispiel.

7.3.9.2 Umsetzung der K-Means-Clusteranalyse in JAMIP

Gestartet wird die K-Means-Clusteranalyse in JAMIP unter *Bildverarbeitungsmethoden -> Methoden der Bildanalyse -> Segmentierungsverfahren -> K-Means-Clusteranalyse*. Das in Abbildung 7.25 zu sehende Modulfenster zeigt zu Beginn in beiden Bildbereichen eine Kopie des aktuellen Bildes des Hauptfensters an, nach Auswahl der Clusteranzahl - voreingestellt sind drei Cluster - und Drücken des Start-Knopfes erscheint dann auf der linken Seite das aktuelle Ergebnis, während rechts die Änderungen des gegenwärtigen Berechnungsschrittes durch rot eingefärbte Pixel angezeigt werden. Zusätzlich lässt sich der Verlauf der Clusteranalyse verfolgen, die Varianz darstellen und ein Fenster mit den ermittelten Clustern anzeigen. Soll der Ablauf des K-Means-Verfahrens nicht

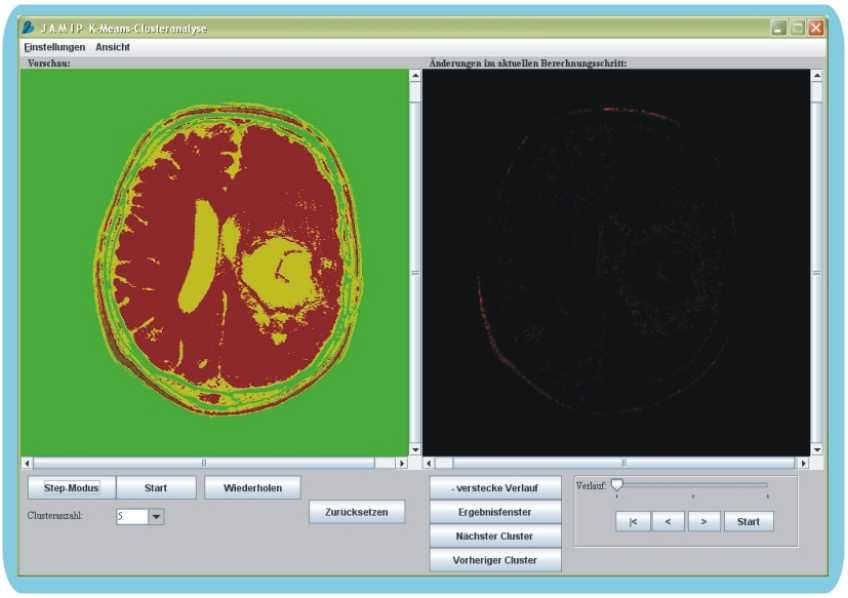


Abbildung 7.25: Die K-Means-Clusteranalyse in JAMIP

automatisch sondern schrittweise erfolgen, so muss der Anwender statt des `Start`- den `Step-Modus`-Knopf drücken.

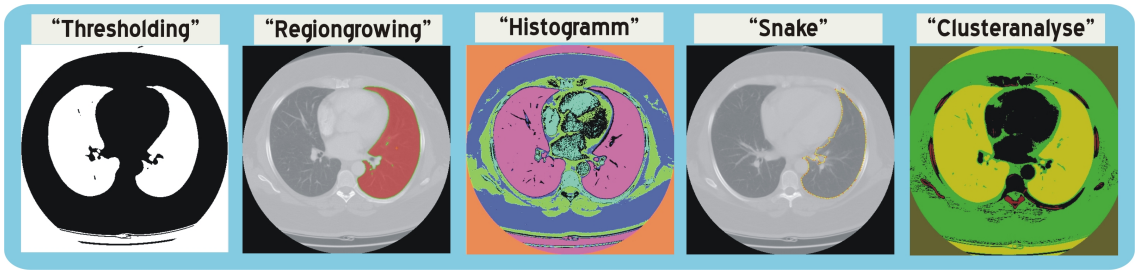


Abbildung 7.26: Segmentierungsergebnisse unterschiedlicher Verfahren in JAMIP

7.3.10 Statistische Texturanalyse mit Cooccurrence-Matrizen

Mit der Statistischen Texturanalyse mit Cooccurrence-Matrizen wurde in JAMIP eine weitere wichtige Methode der Bildanalyse im Rahmen der Medizinischen Bildverarbeitung implementiert.

7.3.10.1 Theoretische Grundlagen zur Statistischen Texturanalyse mit Cooccurrence-Matrizen

Unter dem Begriff Texturanalyse wird die Untersuchung von Texturen, also von Struktureigenschaften eines Bildbereichs, verstanden mit dem Ziel der Texturklassifikation, Textursegmentierung oder Textursynthese. Typisches Werkzeug der Texturanalyse ist die *Cooccurrence-Matrix*, die die auftretenden Häufigkeiten von Grauwertübergängen erfasst. In einem Bild werden dafür die Grauwertkombinationen aller Pixel, die eine vordefinierte geometrische Lage zueinander haben, gezählt und in die Cooccurrence-Matrix eingetragen. Ein unter anderem von Haralick vorgestellter Ansatz basiert auf der statistischen Analyse der Häufigkeit dieser Grauwertübergänge. Insgesamt gibt es 14 der sogenannten *Haralick'schen Texturmerkmale*, mit denen für jede Bildregion, aus der eine Cooccurrence-Matrix berechnet wurde, ein 14-dimensionaler Merkmalsvektor aufgestellt werden kann. Die vier wichtigsten dieser Merkmale sind der Kontrast, die Entropie, das Inverse Difference Moment und das Second Angular Moment. Abbildung 7.27 zeigt als Originalbild einen Grauwertverlauf, den die Cooccurrence-Matrix als Grauwertbild dargestellt als helle Diagonale widerspiegelt.

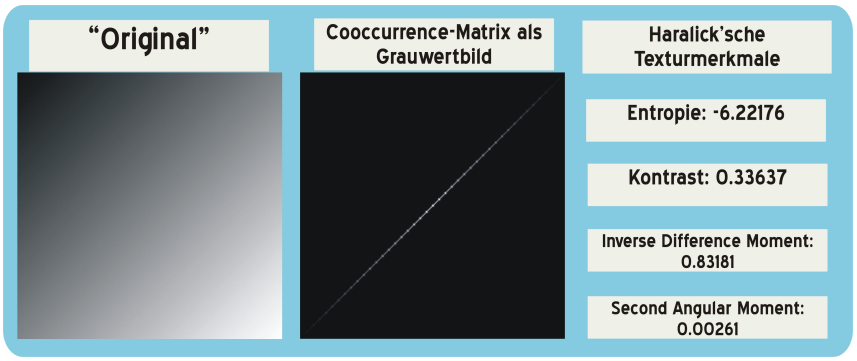


Abbildung 7.27: Beispiel für eine Statistische Texturanalyse mit Cooccurrence-Matrix bei einem Grauwertverlauf

7.3.10.2 Umsetzung der Statistischen Texturanalyse mit Cooccurrence-Matrizen in JAMIP

Unter *Bildverarbeitungsmethoden -> Methoden der Bildanalyse -> Statistische Texturanalyse mit Cooccurrence-Matrizen* wird das Modul für die Berechnung der Statistischen Texturanalyse gestartet, in Abbildung 7.28 ist das entsprechende Fenster zu sehen. Links wird eine Kopie des aktuellen Bildes des Hauptfensters angezeigt, in dem der Benutzer eine ROI für die Berechnung der Cooccurrence-Matrix durch Aufziehen eines Rechtecks

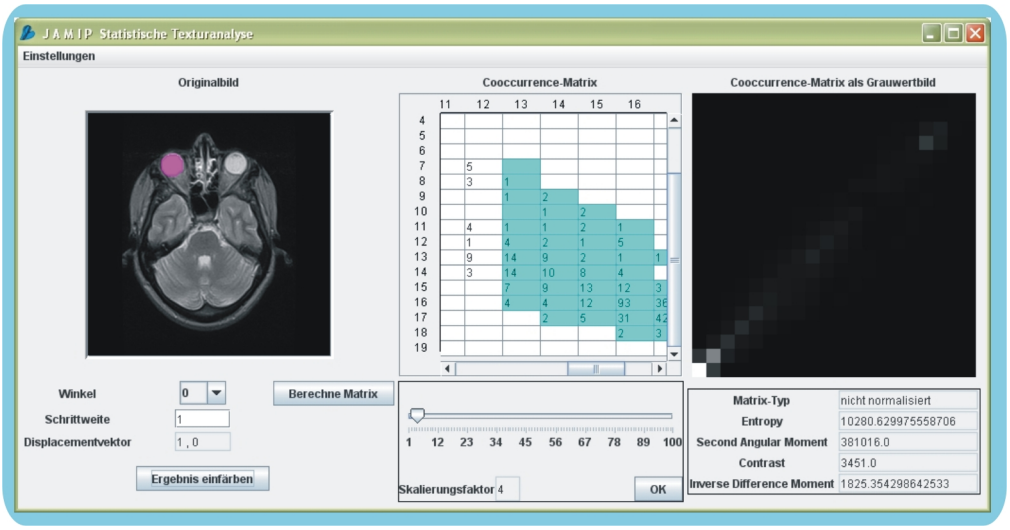


Abbildung 7.28: Das Modul Statistische Texturanalyse mit Cooccurrence-Matrizen in JAMIP

auswählen muss. Nach Auswahl eines Winkels und Einstellen einer Schrittweite erscheint dann in der Mitte die berechnete Cooccurrence-Matrix und rechts daneben die Matrix als Grauwertbild. Da die Matrix im Allgemeinen recht groß ist, kann mit einem Schieberegler die Anzahl der zu untersuchenden Grauwertkombinationen herunterskaliert werden. Unter dem Menüpunkt *Einstellungen* besteht die Auswahl zwischen 14 verschiedenen Harralick'schen Texturmerkmalen, deren Werte dann in dem Modulfenster angezeigt werden. Standardmäßig werden mit Entropy, Second Angular Moment, Contrast und Inverse Different Moment die vier wichtigsten Merkmale berechnet. Weiterhin kann die Cooccurrence-Matrix entweder nicht normalisiert, normalisiert, symmetrisch oder isotrop sein. Es besteht außerdem die Möglichkeit des Einfärbens ausgewählter Grauwertkombinationen. Dazu müssen die gewünschten Tabellenzellen der Cooccurrence-Matrix zunächst angeklickt werden - sie wechseln dabei ihre Farbe von weiß nach türkis, bevor dann der Button `Ergebnis einfärben` gedrückt wird. Alle Pixel des Originalbildes, die zu einer der ausgewählten Grauwertkombinationen gehören, wechseln jetzt ihre Farbe. Durch geschickte Auswahl der Grauwertkombinationen ist sogar eine Segmentierung möglich.

7.3.10.3 Die Statistische Texturanalyse im Animationsmodus

Der Animationsmodus für die Statistische Texturanalyse mit Cooccurrence-Matrizen startet unter *Animationsmodus -> Statistische Texturanalyse* und ist in Abbildung 7.29 zu sehen. Gezeigt wird hier die schrittweise Berechnung und Entstehung der Cooccurrence-

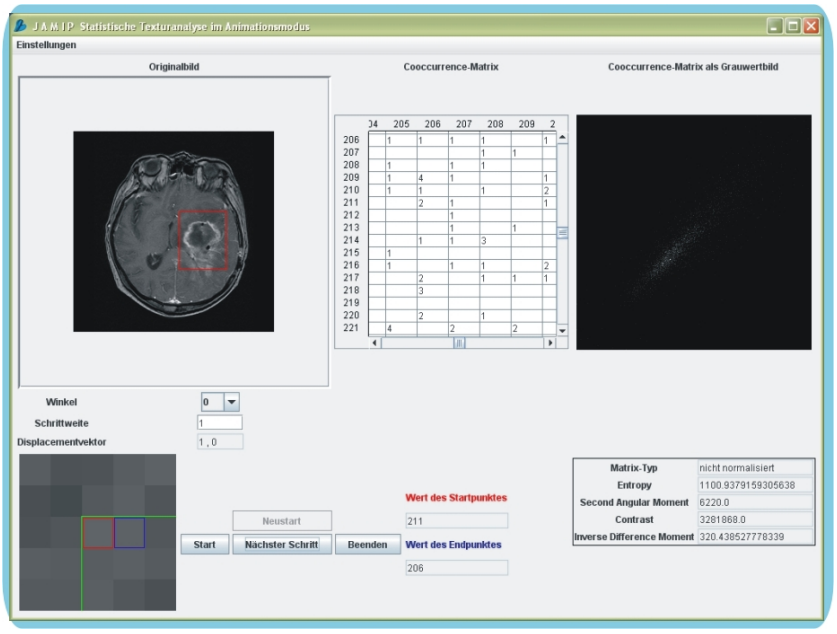


Abbildung 7.29: Die Texturanalyse im Animationsmodus

Matrix. Dazu muss der Nutzer auch hier zunächst eine ROI im Originalbild aufziehen und den Winkel sowie die Schrittweite eintragen, bevor mit dem Step-Modus begonnen werden kann. Im Unterschied zum "normalen" Verfahren wird hier zusätzlich in einer Extrakomponente ein vergrößerter Ausschnitt des Originalbildes angezeigt, in dem die Ermittlung der aktuellen Grauwertkombination farblich hervorgehoben wird. Parallel wird die Tabelle der Cooccurrence-Matrix an der entsprechenden Stelle um eins erhöht. So soll der Ablauf verdeutlicht werden.

7.4 Datenversand und Verschlüsselung

Neben den lokalen Verfahren zur Bildbearbeitung bietet JAMIP die Möglichkeit, Bilddaten über ein Netzwerk an ein anderes JAMIP-Programm zu schicken. Der Versand dieser Daten kann dabei verschlüsselt oder unverschlüsselt erfolgen. Da es sich bei medizinischen Bilddaten im Allgemeinen um nicht-öffentliche, personenbezogene Informationen handelt, ist bei JAMIP standardmäßig der verschlüsselte Versand voreingestellt. Näheres zu dem realisierten Verschlüsselungsverfahren ist Abschnitt 7.4.2 zu entnehmen. Zunächst wird jedoch im folgenden Abschnitt 7.4.1 der Datenversand in JAMIP vorgestellt.

7.4.1 Realisierung des Datenversandes in JAMIP

Sollen mit JAMIP Bilddaten an ein JAMIP-System auf einem anderen Rechner verschickt werden, beispielsweise um anschließend eine telekooperative Besprechung dieser Daten abzuhalten, so muss zunächst sichergestellt sein, dass beide Rechner über einen Internetzugang verfügen. Gleichzeitig sollte dem Sender die IP-Adresse des Empfängers bekannt sein, da die interne Kommunikation bei JAMIP mittels Java-RMI (siehe dazu Abschnitt 5.1.3) realisiert wurde, diese API wiederum TCP-IP-basiert ist und die Verbindung daher über die IP-Adressen zustande kommt.

Durch Auswahl des Buttons oder über den Menüpunkt *Datei -> Programm für den Datenversand anmelden* melden sich Server und Empfänger zunächst zum Datenversand- bzw. -empfang an. Ist das Programm sende- und empfangsbereit, so wechselt das Symbol seine Farbe von rot nach grün: . Sollen Daten versendet werden, so muss der Anwender nun entweder den Menüpunkt *Datei -> Versenden* oder den Button auswählen. Es erscheint dann ein Dateiauswahlfenster, in dem der Sender die zu verschickenden

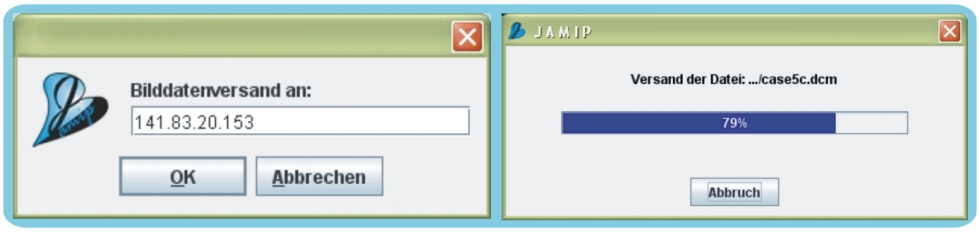


Abbildung 7.30: Die Eingabemaske für den Datenversand; links: Eingabe der IP-Adresse des Empfängers, rechts: Verlaufskontrolle

Dateien markiert, er kann dabei entweder ganze Ordner anwählen oder auch nur einzelne Bilddateien. Anschließend muss die schon beschriebene IP-Adresse des Empfängers eingegeben werden. Konnte die Verbindung hergestellt werden, so erfolgt jetzt je nach ausgewählter Verschlüsselungsmethodik - siehe dazu Abschnitt 7.4.2 - der Datenversand. Damit die versendeten Bilddaten auf beiden Seiten wiedergefunden werden können, erfolgt die Ablage bei JAMIP automatisch, d. h. Sender und Empfänger erhalten jeweils eine Kopie der Daten, die in exakt gleich lautenden Unterverzeichnissen des Programms auf den jeweiligen Systemen gespeichert werden. Sendet also Rechner A Daten an Rechner B, so werden diese auf beiden Seiten in folgendem Verzeichnis abgelegt: ... / *JAMIP / TELEKONFERENZ / < RechnernameA - RechnernameB > / < Datum + Uhrzeit>* /.... Die Reihenfolge der aneinandergehängten Rechnernamen ergibt sich hier nach dem Alphabet und das Datum aus dem genauen Absendedatum inklusive der Uhrzeit.

Sind alle Daten erfolgreich übermittelt, so erhalten Sender und Empfänger eine abschließende Statusmeldung über den erfolgten Versand. Die Systeme können jetzt durch erneutes Drücken des entsprechenden Buttons vom Datenversand wieder abgemeldet werden.

7.4.2 Realisierung der Verschlüsselung in JAMIP

In JAMIP wurde für den chiffrierten Versand medizinischer Bilddaten über ein Netzwerk das in Abschnitt 6.4.2 beschriebene hybride System mit symmetrischer Verschlüsselung des Datensatzes und asymmetrischer Verschlüsselung des Schlüssels installiert. Für den Programmbenutzer geschieht dieser Teil des Datenversandes weitestgehend unsichtbar. Er hat lediglich wie in Abbildung 7.31 zu sehen ist unter dem Menüpunkt *Bearbeiten -> Einstellungen -> Verschlüsselung* die Auswahl zwischen zwei verschiedenen symmetrischen Chiffrieralgorithmen - DES und IDEA, von denen der IDEA-Algorithmus aufgrund seiner höheren Sicherheit standardmäßig voreingestellt ist. Des Weiteren besteht die Möglichkeit, statt der gesamten Bilddatei nur den personenbezogenen Header für die Übertragung zu verschlüsseln oder gar nicht zu chiffrieren. Letzteres sollte in Anbetracht der sensiblen Daten nur in Ausnahmesituationen, wie z. B. innerhalb eines internen, nach außen gesicherten Netzwerkes, erfolgen.

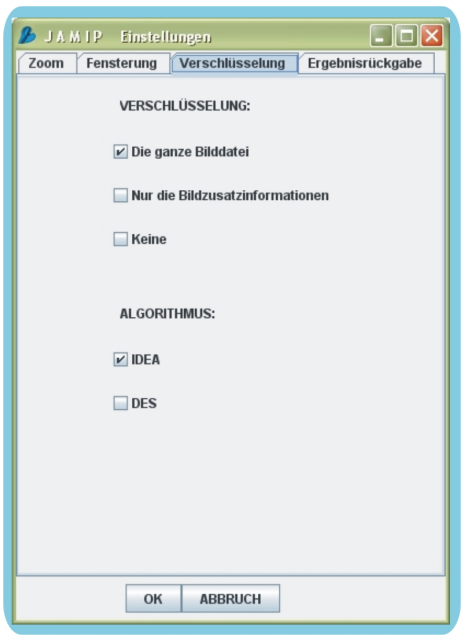


Abbildung 7.31: Einstellung der Verschlüsselung in JAMIP

7.5 Das Telekonferenz-Modul JAMIPTele

Wie zu Beginn dieses Kapitels erwähnt sind Telekonferenzen mit JAMIP über das integrierte und auch eigenständig lauffähigen Modul JAMIPTele möglich. JAMIPTele ist als telekonferenzfähige Variante von JAMIP realisiert. Der Anwendungsfokus liegt hier auf der telekooperativen Besprechung medizinischer Bilddaten über ein Netzwerk. Das Programm ist JAMIP in Aufbau und Funktion nachempfunden - allerdings mit eingeschränktem Funktionsumfang im Bereich der Bildverarbeitung, so steht beispielsweise der speziell für den Einsatz in der Lehre in JAMIP eingebundene Animationsmodus in JAMIPTele nicht zur Verfügung. Für die Anzeige und Bearbeitung medizinischer Bilddaten ist auch in JAMIPTele ein DICOM-Viewer integriert, daneben stehen ebenfalls die Standardoperationen einer radiologischen Workstation wie das Zoomen, die Invertierung, die Level-Window-Operation etc. zur Verfügung. Zusätzlich sind ausgewählte Bildverarbeitungsalgorithmen wie z. B. das Regiongrowing oder die Faltung auch für die quasi-parallele Ausführung im Rahmen einer Telekonferenz realisiert worden.
Zu einer telekooperativen Sitzung können sich bis zu 9 Personen gleichzeitig anmelden; jedem Teilnehmer wird dabei eine Farbe zugeordnet, in der sein Mauszeiger in den Programmen der anderen Teilnehmer erscheint. Zusätzlich dazu enthält das Programm eine Statuszeile mit einer Auflistung der miteinander verbundenen Teilnehmer, die jeden Teilnehmernamen in der ihm zugeordneten Farbe anzeigt. Des weiteren werden alle Eingaben, die ein Teilnehmer zur parallel stattfindenden Kommunikation während einer Konferenz in dem Chatfenster des Programms macht, farblich entsprechend bei den anderen Teilnehmern dargestellt. In der jeweils eigenen Anwendung sieht der Mauszeiger hingegen “normal” aus, der eigene Name und die eigenen Chateingaben werden nicht farbkodiert. Die Steuerung des Programms läuft wie im nicht konferenzfähigen Modus, alle Eingaben die per Mausklick oder über die Tastatur gemacht werden, werden ausgeführt, zusätzlich jedoch auf die Programme der übrigen Teilnehmer übertragen und dort ebenfalls ausgeführt. Auf Funktionen zur Regelung des Aktionsrechtes wurde bewusst verzichtet, da sich dies im praktischen Einsatz als zu umständlich erwies und mit dem Chatfenster neben der ansonsten üblichen Kommunikation über Telefon eine weitere Möglichkeit zum Informationsaustausch zur Verfügung steht.

Die folgenden Abschnitte geben Auskunft über die Besonderheiten beim Programmstart von JAMIPTele (Abschnitt 7.5.1) sowie über die Unterschiede im Aufbau der Benutzeroberfläche im Vergleich zu der von JAMIP (Abschnitt 7.5.2). Anschließend wird in Abschnitt 7.5.3 kurz auf die Umsetzung der Bildverarbeitung in JAMIPTele eingegangen.

7.5.1 Start einer Telekonferenz mit JAMIPTele

Vor Beginn einer Telekonferenz sollte auf allen beteiligten Rechnern dasselbe zur gemeinsamen Betrachtung vorgesehene Datenmaterial vorliegen. Alternativ lässt sich JAMIPTele in seiner Ausführung als Applet so konfigurieren, dass das zu betrachtende Datenmaterial auch von einem gemeinsam genutzten Server geladen werden kann. In beiden Fällen werden die zu betrachtenden Bilder vor dem eigentlichen Beginn einer Konferenz ausgewählt.

Gestartet werden kann das Programm aus dem Hauptprogramm JAMIP durch Anklicken des Telekonferenz-Icons in der Buttonleiste oder durch Auswahl der Einstellung *Telekonferenz* aus dem Datei-Menü des Systems. Wahlweise ist JAMIPTele auch eigenständig lauffähig, auf der Webseite des Projekts unter `http://www.imi2.uni-luebeck.de/jamip/jamiptele` finden sich sowohl eine Application- als auch eine Applet-Variante.

Anders als in JAMIP erscheint bei JAMIPTele zunächst das in Abbildung 7.32 zu sehende Anmeldefenster, in dem der Benutzer seinen Namen eintragen und die Auswahl über die

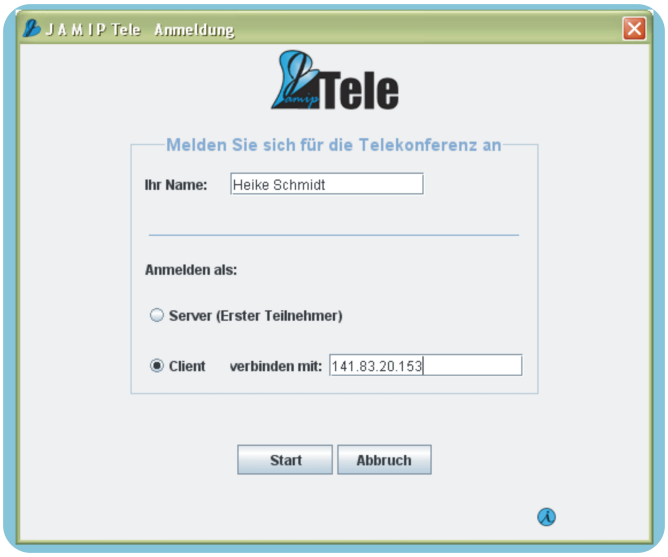


Abbildung 7.32: Anmeldung zur Telekonferenz

Art der Sitzung treffen muss. Gewählt werden kann dabei zwischen *Client* und *Server*, wobei sich bei jeder Telekonferenz der erste der Teilnehmer immer als Server anmelden muss und die übrigen sich dann als Client über die Angabe der IP-Adresse mit diesem verbinden. Der Server übernimmt dabei die Kontrollfunktion über die Teilnehmer einer

Konferenz, für jede Neuanmeldung ist dabei zunächst das 'OK' des Servers erforderlich. Konnte die Verbindung zwischen den Systemen hergestellt werden, so erscheint bei jedem Teilnehmer eine ferngesteuerte bzw. fernsteuerbare Version des JAMIPTele-Programms mit dem zuvor ausgewählten Bildmaterial. Das Kontrollfenster (siehe Abbildung 7.33) gibt zusätzlich Auskunft über die Anzahl der Teilnehmer der Telekonferenz, die jeweiligen IP-Adressen und den Status der Verbindung.
Beendet der Server die Konferenz, so endet damit die Konferenz für alle Teilnehmer, meldet sich hingegen einer der Clients von der Konferenz ab, so können die übrigen Teilnehmer weiterhin telekooperativ arbeiten.

Abbildung 7.33: Das Kontrollfenster

7.5.2 Die Benutzeroberfläche von JAMIPTele

Nach erfolgreicher Anmeldung zu einer Telekonferenz erscheint bei allen Teilnehmern die in Abbildung 7.34 zu sehende Benutzeroberfläche von JAMIPTele, die sich in einigen Punkten von der von JAMIP unterscheidet und nun nachfolgend kurz beschrieben wird:

- ❶ In der Mitte befindet sich das *Hauptfenster* des Tools, das die eingeladenen Bilddaten in Originalgröße anzeigt; im Unterschied zu JAMIP kann dieses Hauptfenster hier allerdings nicht unterteilt werden.

- ❷ Die *Übersichtsanzeige* zeigt in einer vertikalen Übersichtsleiste alle in das Programm geladenen Bilder klein an. Mittels der Buttons zum Durchsteppen (siehe unter Punkt 4) wird ein Bild dieser Leiste rot umrandet markiert und im Hauptfenster in Originalgröße dargestellt.

- ❸ Über die *Menüleiste* des Systems wird das Bearbeiten der Bilder gesteuert. Eingeteilt ist sie in folgende Kategorien:

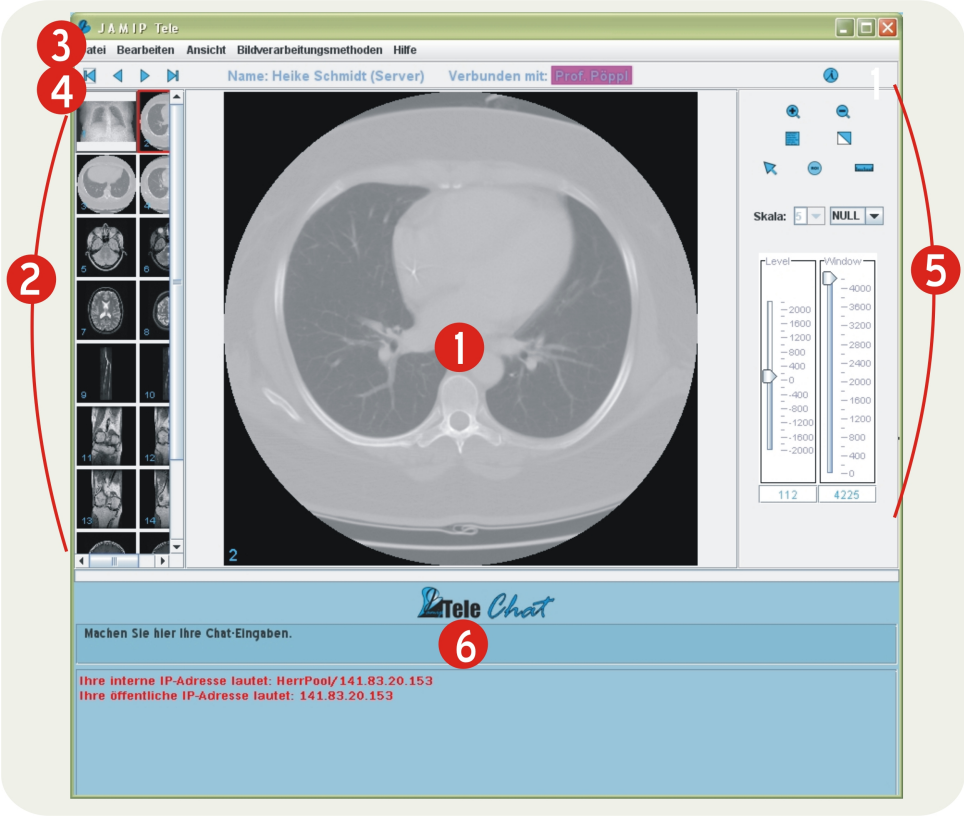


Abbildung 7.34: Die Benutzeroberfläche von JAMIPTele

- *Datei:* Da die Bilder in JAMIPTele vor dem Programmstart ausgewählt werden müssen und während einer Konferenz keine neuen Bilder eingeladen werden können, kann über diesen Menüpunkt das Programm lediglich geschlossen werden.
- *Bearbeiten:* Wie bei JAMIP verbergen sich auch hier unter diesem Punkt Optionen zum Ändern der Bilder mittels Zoom oder Invertierung. Der Unterpunkt *Einstellungen* ist in JAMIPTele jedoch nicht eingefügt.
- *Ansicht:* Hier lässt sich die Headerinformation eines DICOM-Bildes anzeigen sowie die Ein- oder Ausblendung des Kontrollfensters für die Telekonferenz vornehmen.
- *Bildverarbeitungsmethoden:* Aufgeteilt nach *Methoden der Bildvorverarbeitung* und *Methoden der Bildanalyse* finden sich unter diesem Menüpunkt die in JAMIPTele integrierten Bildverarbeitungsmethoden, die während einer Telekonferenz quasi-parallel ausführbar sind. Näheres hierzu kann Abschnitt 7.5.3 entnommen werden.

  - *Hilfe:* Hierunter finden sich Informationen zur Programmversion sowie die Bedienungsanleitung des Systems

- ❹ Die *Buttonleiste* enthält in JAMIPTele Funktionen zum Durchsteppen der Bilder der Übersichtsleiste und wird als Namensanzeige für die Telekonferenz genutzt:
  - ⏮, ◀, ▶, ⏭ Mit diesen Knöpfen wird wahlweise das letzte, das vorherige, das nächste oder das erste Bild der Übersichtsleiste groß im Hauptfenster angezeigt.
  - *Name:* Hier wird der eigene bei der Anmeldung zur Telekonferenz eingegebene Name angezeigt.
  - *Verbunden mit:* Hier werden die Namen der Telekonferenzteilnehmer aufgelistet, mit denen man sich aktuell in einer Telekonferenz befindet; die Namen dieser Teilnehmer sind in derselben Farbe eingefärbt, in der auch ihr Mauszeiger in der Anwendung erscheint.
  - λ Hinter diesem Button verbergen sich Informationen zur Programmversion.

- ❺ Auch für JAMIPTele konnten die *Standardfunktionen* eines Telemedizinsystems wie in JAMIP realisiert werden; sie sind in Abschnitt 7.2 ausführlich beschrieben, in JAMIPTele werden die Operationen dann in allen zur Telekonferenz angemeldeten Programmen durchgeführt.

- ❻ Anders als bei JAMIP wurde in JAMIPTele in diesen Teil der Benutzeroberfläche das sogenannte *Chatfenster* integriert, das eine parallele Kommunikation ermöglicht. Jeder Teilnehmer, der eine Chat-Nachricht an die anderen Teilnehmer versenden möchte, gibt diese dazu im oberen Bereich des Fensters ein. Durch Drücken des `Return`-Buttons erscheint diese Nachricht dann bei allen Konferenzteilnehmern im unteren Teil des Chat-Bereichs - und zwar immer in der dem Teilnehmer zugeordneten Farbe (siehe Abbildung 7.35).

7.5.3 Die Bildverarbeitung in JAMIPTele

Für die Bildverarbeitung in JAMIPTele wurden nur die Methoden ausgewählt, deren Anwendung im Rahmen einer Telekonferenz sinnvoll erschien, dazu gehören:

- die Histogrammerstellung,

Abbildung 7.35: Das Chatfenster von JAMIPTele mit Meldungen verschiedener Konferenzteilnehmer

- die Profilerzeugung,
- die Grauwerttransformation,
- die Filter,
- das Schwellwertverfahren,
- das Bereichswachstumsverfahren und
- die Histogrammbasierte Segmentierung.

Einzelheiten zu den Verfahren an sich werden in Abschnitt 7.3 erläutert.
Kompliziertere Methoden wie beispielsweise die in JAMIP implementierten Aktiven Konturen, die erheblich mehr Interaktivität vom Benutzer erfordern als z. B. das Schwellwertverfahren, erwiesen sich für die Ausführung in einer Telekonferenz als zu zeitaufwendig. Stattdessen können Ergebnisse, die mit diesen Verfahren gemacht wurden, vor der Konferenz ausgetauscht und dann gemeinsam besprochen werden.
Die Umsetzung der in JAMIPTele integrierten Verfahren entspricht größtenteils der aus JAMIP, es wurden jedoch an einigen Stellen Vereinfachungen gemacht, so steht beispielsweise beim Regiongrowing statt dreien nur ein Homogenitätskriterium zur Auswahl.

Ruft ein Konferenzteilnehmer eine Bildverarbeitungsmethode auf, so öffnen sich die zur Steuerung dieser Methode gehörigen Fenster bei allen Teilnehmern; alle Teilnehmer bewirken durch Änderungen in ihren Fenstern Änderungen in den Fenstern aller Teilnehmer. So wird die Verschiebung eines Reglers in denselben Fenstern bei allen Teilnehmern ausgeführt und sichtbar, im Unterschied zum Hauptfenster sieht man hier die verschiedenfarbigen Mauszeiger jedoch nicht. Abbildung 7.36 zeigt ein Beispiel für eine quasi-parallele Ausführung des Schwellwertverfahrens während einer Telekonferenz

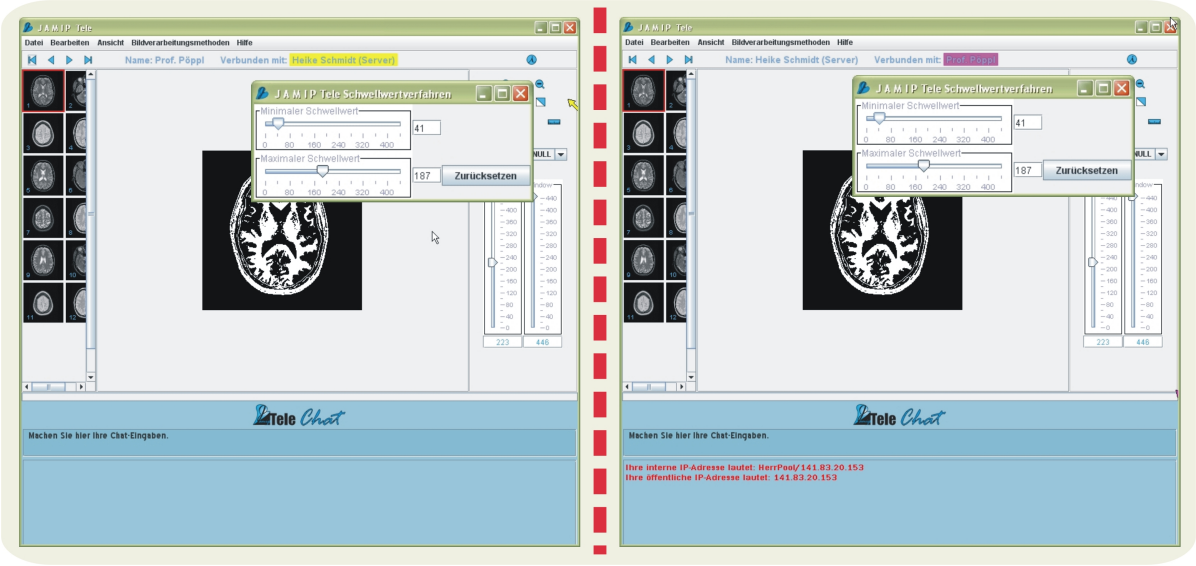


Abbildung 7.36: Das Schwellwertverfahren während einer Telekonferenz mit JAMIPTele

zwischen zwei Teilnehmern, der im Bild links zu sehende Teilnehmer steuert hierbei das Verfahren. Im Gegensatz zu JAMIP werden bei JAMIPTele alle Ergebnisse bei Abschluss eines Verfahrens wieder zurückgesetzt.

8 JAMIP - Einsatz und Vergleich

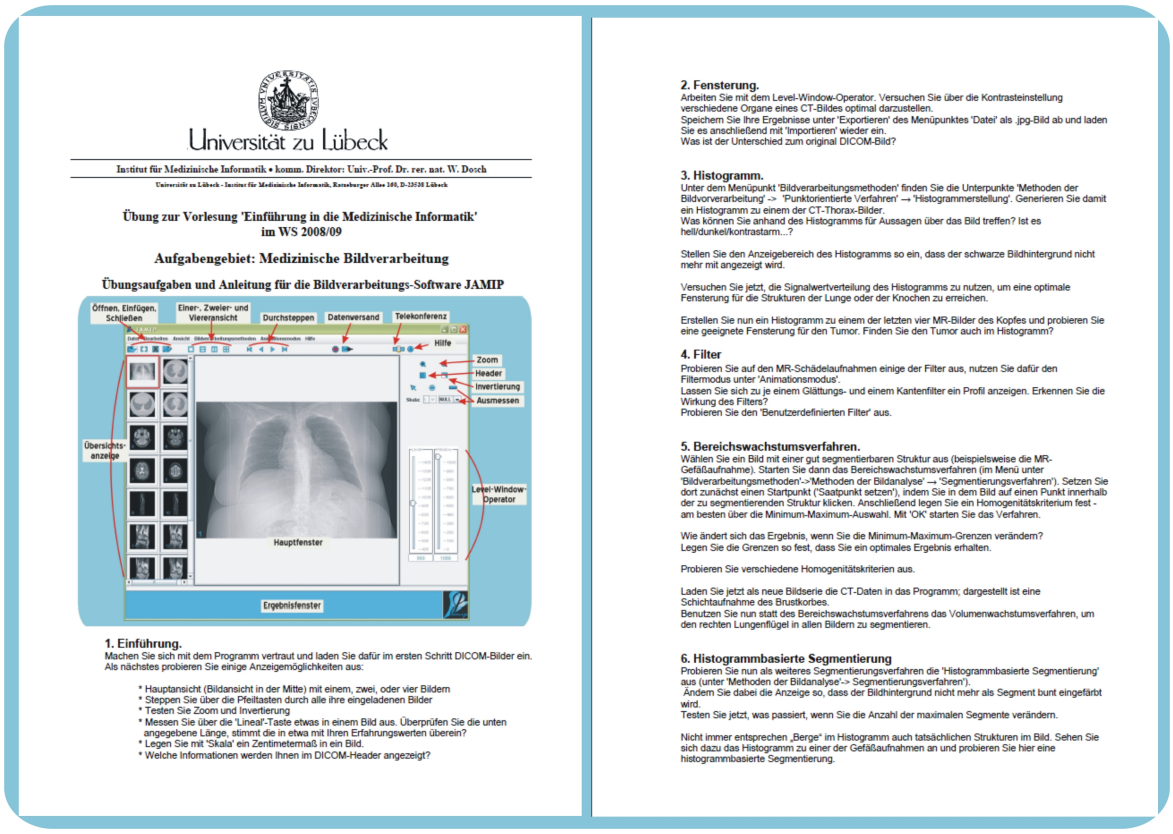
Das Projekt JAMIP läuft am Institut für Medizinische Informatik bereits seit einigen Jahren, das Programm unterliegt dabei einer fortwährenden Weiterentwicklung wie z. B. durch die Integration neuer Bildverarbeitungsmodule. Daneben wird das System durch den Einsatz in Vorlesungen und Übungen immer wieder evaluiert und optimiert.
Dieses Kapitel beschreibt in Abschnitt 8.1 die Veranstaltungen und Projekte, in deren Rahmen JAMIP und JAMIPTele an der Universität zu Lübeck und anderen Institutionen bereits praktisch genutzt wurden und werden. Danach liefert Abschnitt 8.2 einen Vergleich des in dieser Arbeit entstandenen Programms mit in Kapitel 4 beschriebenen Softwaresystemen für den Einsatz in Telemedizin und Lehre.

8.1 Einsatz

Entwickelt für den Einsatz in der Präsenzlehre an der Universität zu Lübeck und für das Online-Angebot der FernUniversität Hagen im Bereich der Grundlagen der Medizinischen Bildverarbeitung konnte JAMIP mittlerweile bereits in verschiedenen Veranstaltungen eingesetzt werden. Abschnitt 8.1.1 zeigt diese im Überblick.
Daneben konnte auch das Telekonferenz-Modul JAMIPTele in mehrere praktische Anwendungskontexte integriert werden, die in Abschnitt 8.1.2 näher vorgestellt werden.
Als besonders vorteilhaft für den Einsatz von JAMIP und JAMIPTele hat sich im Zusammenhang mit den an einer Universität vorherrschenden heterogenen Betriebssystemzuständen die Plattformunabhängigkeit des Programms erwiesen, die Installation dieses Java-basierten Tools erfolgte auf allen Plattformen der Universität nahezu problemlos.

8.1.1 Einsatz JAMIP

JAMIP wird als Tool sowohl in Vorlesungen zur Unterstützung des Lehrenden bei der Demonstration des Ablaufs eines Bildverarbeitungsalgorithmus genutzt als auch in Übungen und Praktika zum Erlernen elementarer Fertigkeiten im Umgang mit medizinischen Bilddaten und Programmen zur Bildbearbeitung sowie zum Ausprobieren der vorgestellten Algorithmen [6, 7, 8]. Dabei findet das Programm neben der Universität zu Lübeck



Universität zu Lübeck

Institut für Medizinische Informatik • komm. Direktor: Univ.-Prof. Dr. rer. nat. W. Dosch

Universität zu Lübeck - Institut für Medizinische Informatik, Ratzeburger Allee 160, D-23538 Lübeck

Übung zur Vorlesung 'Einführung in die Medizinische Informatik' im WS 2008/09

Aufgabengebiet: Medizinische Bildverarbeitung

Übungsaufgaben und Anleitung für die Bildverarbeitungs-Software JAMIP

1. Einführung.
Machen Sie sich mit dem Programm vertraut und laden Sie dafür im ersten Schritt DICOM-Bilder ein. Als nächstes probieren Sie einige Anzeigemöglichkeiten aus:

* Hauptansicht (Bildansicht in der Mitte) mit einem, zwei, oder vier Bildern
* Steppen Sie über die Pfeiltasten durch alle ihre eingeladenen Bilder
* Testen Sie Zoom und Invertierung
* Messen Sie über die 'Lineal'-Taste etwas in einem Bild aus. Überprüfen Sie die unten angegebene Länge, stimmt die in etwa mit Ihren Erfahrungswerten überein?
* Legen Sie mit 'Skala' ein Zentimetermaß in ein Bild.
* Welche Informationen werden Ihnen im DICOM-Header angezeigt?

2. Fensterung.
Arbeiten Sie mit dem Level-Window-Operator. Versuchen Sie über die Kontrasteinstellung verschiedene Organe eines CT-Bildes optimal darzustellen.
Speichern Sie Ihre Ergebnisse unter 'Exportieren' des Menüpunktes 'Datei' als .jpg-Bild ab und laden Sie es anschließend mit 'Importieren' wieder ein.
Was ist der Unterschied zum original DICOM-Bild?

3. Histogramm.
Unter dem Menüpunkt 'Bildverarbeitungsmethoden' finden Sie die Unterpunkte 'Methoden der Bildvorverarbeitung' -> 'Punktorientierte Verfahren' → 'Histogrammerstellung'. Generieren Sie damit ein Histogramm zu einem der CT-Thorax-Bilder.
Was können Sie anhand des Histogramms für Aussagen über das Bild treffen? Ist es hell/dunkel/kontrastarm...?

Stellen Sie den Anzeigebereich des Histogramms so ein, dass der schwarze Bildhintergrund nicht mehr mit angezeigt wird.

Versuchen Sie jetzt, die Signalwertverteilung des Histogramms zu nutzen, um eine optimale Fensterung für die Strukturen der Lunge oder der Knochen zu erreichen.

Erstellen Sie nun ein Histogramm zu einem der letzten vier MR-Bilder des Kopfes und probieren Sie eine geeignete Fensterung für den Tumor. Finden Sie den Tumor auch im Histogramm?

4. Filter
Probieren Sie auf den MR-Schädelaufnahmen einige der Filter aus, nutzen Sie dafür den Filtermodus unter 'Animationsmodus'.
Lassen Sie sich zu je einem Glättungs- und einem Kantenfilter ein Profil anzeigen. Erkennen Sie die Wirkung des Filters?
Probieren Sie den 'Benutzerdefinierten Filter' aus.

5. Bereichswachstumsverfahren.
Wählen Sie ein Bild mit einer gut segmentierbaren Struktur aus (beispielsweise die MR-Gefäßaufnahme). Starten Sie dann das Bereichswachstumsverfahren (im Menü unter 'Bildverarbeitungsmethoden'->'Methoden der Bildanalyse' → 'Segmentierungsverfahren'). Setzen Sie dort zunächst einen Startpunkt ('Saatpunkt setzen'), indem Sie in dem Bild auf einen Punkt innerhalb der zu segmentierenden Struktur klicken. Anschließend legen Sie ein Homogenitätskriterium fest - am besten über die Minimum-Maximum-Auswahl. Mit 'OK' starten Sie das Verfahren.

Wie ändert sich das Ergebnis, wenn Sie die Minimum-Maximum-Grenzen verändern?
Legen Sie die Grenzen so fest, dass Sie ein optimales Ergebnis erhalten.

Probieren Sie verschiedene Homogenitätskriterien aus.

Laden Sie jetzt als neue Bildserie die CT-Daten in das Programm; dargestellt ist eine Schichtaufnahme des Brustkorbes.
Benutzen Sie nun statt des Bereichswachstumsverfahrens das Volumenwachstumsverfahren, um den rechten Lungenflügel in allen Bildern zu segmentieren.

6. Histogrammbasierte Segmentierung
Probieren Sie nun als weiteres Segmentierungsverfahren die 'Histogrammbasierte Segmentierung' aus (unter 'Methoden der Bildanalyse'-> Segmentierungsverfahren').
Ändern Sie dabei die Anzeige so, dass der Bildhintergrund nicht mehr als Segment bunt eingefärbt wird.
Testen Sie jetzt, was passiert, wenn Sie die Anzahl der maximalen Segmente verändern.

Nicht immer entsprechen „Berge" im Histogramm auch tatsächlichen Strukturen im Bild. Sehen Sie sich dazu das Histogramm zu einer der Gefäßaufnahmen an und probieren Sie hier eine histogrammbasierte Segmentierung.

Abbildung 8.1: Einsatz von JAMIP im Rahmen des Nebenfachstudiums 'Medizinische Informatik' im Informatikstudium an der Universität zu Lübeck

auch an der FernUniversität Hagen und weiteren Instituten Anwendung. Die folgenden Abschnitte geben einen Überblick.

8.1.1.1 JAMIP an der Universität zu Lübeck

Im Bereich des Nebenfachstudiums Medizinische Informatik an der Universität zu Lübeck wurde JAMIP in der Hauptstudiumsvorlesung *Grundlagen Medizinischer Bild- und Signalanalyse* eingesetzt. Zu Evaluierungszwecken wurde das Programm hier auf CD-ROM an die Studenten der Vorlesung ausgeteilt. Die Erfahrungen damit waren durchweg positiv, Probleme bei der Installation und Anwendung gab es keine. Weiterhin wird JAMIP in Lübeck für die Vorlesung und Übung *Einführung in die Medizinische Informatik* in den Bachelor-Studiengängen Informatik mit Nebenfach Medizinische Informatik und Medizinische Ingenieurwissenschaft sowie für das Praktikum *Software-Schnittstellen im Gesundheitswesen* genutzt. Ein Beispiel für eine Übung mit JAMIP ist in Abbildung 8.1 zu sehen, die Studenten müssen hier unter Anleitung verschiedene Verfahren des

Bildverarbeitungslabors auf unterschiedlichem Bildmaterial ausprobieren.
In ähnlicher Art und Weise wird JAMIP mittlerweile in Lübeck auch für die Ausbildung im Rahmen des Studiengangs Humanmedizin eingesetzt. In der Vorlesung und dem zugehörigen Praktikum *Bildverarbeitung in der Medizin* des Querschnittsbereichs "Epidemiologie, medizinische Biometrie und medizinische Informatik" wird das Tool benutzt, um Medizinstudenten die Methoden der Bildverarbeitung zu veranschaulichen. Gleichzeitig soll ihnen ein Einblick in das grundsätzliche Arbeiten mit Bildverarbeitungsprogrammen und Bildern in der Medizin vermittelt werden. Durch die Umstrukturierung der Querschnittsveranstaltung von einer reinen Vorlesung hin zu einer Mischung aus theoretischen und praktischen Anteilen wie dem Arbeiten mit dem Bildverarbeitungslabor JAMIP konnten die Evaluierungsergebnisse für das Studienfach Medizinische Informatik erheblich verbessert werden (siehe dazu [8]).
Für die Praktika und Übungen wurde JAMIP sowohl im Bildlabor des Instituts für Medizinische Informatik als auch auf den Workstations des Rechenzentrums der Universität installiert. Außerdem steht das Programm zusammen mit den zugehörigen Vorlesungen und Übungen den Studenten als Teil der E-Learning-Plattform Moodle des Instituts für Medizinische Informatik (siehe hierzu `http://moodle.imi.uni-luebeck.de/moodle/`) zur Verfügung.

8.1.1.2 JAMIP an der FernUniversität Hagen

Studierende der FernUniversität Hagen, die Informatik mit Nebenfach Medizinische Informatik belegen, können mit Hilfe der im Rahmen des medin-Projekts entwickelten Webstruktur online studieren (siehe Abschnitt 2.2.2). JAMIP ist als mediales Element in diese Webseiten integriert. Das Programm steht in seiner Funktion als virtuelles Bildverarbeitungslabor zusammen mit ausgewählten, anonymisierten Testdaten zum kostenlosen Download zur Verfügung. Die im Online-Material vorgestellten Bildverarbeitungsalgorithmen können so ausprobiert sowie spezielle Selbsttest- und Übungsaufgaben gelöst werden. Außerdem sind einzelne Verfahren des Systems als Applets in die entsprechenden Vorlesungen auf den Webseiten des Studiengangs eingebunden. So findet sich beispielsweise anschließend an die Erklärung des Algorithmus' zum Bereichswachstumsverfahren das in Abbildung 8.2 zu sehende Applet des Animationsmodus zu diesem Verfahren.

8.1.1.3 JAMIP an anderen Universitäten und Instituten

Auch an der Universität Hamburg werden Medizinstudenten mit JAMIP in Pflichtseminaren im Fach Medizinische Informatik im Bereich der Medizinischen Bildverarbeitung

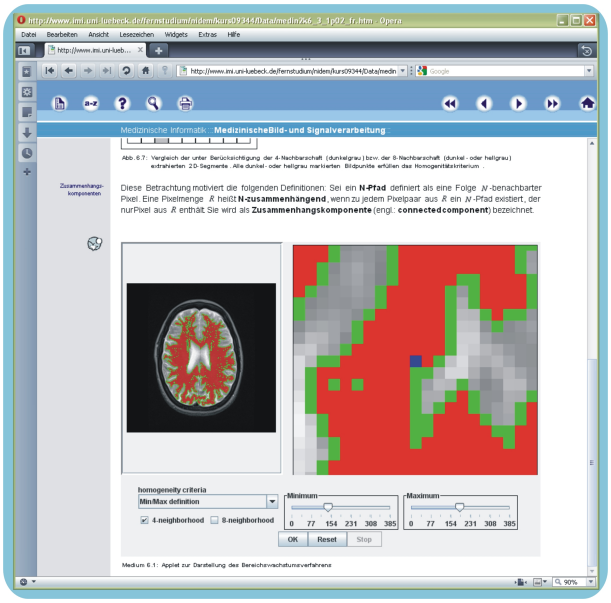


Abbildung 8.2: Das Bereichswachstumsverfahren im Animationsmodus in den Online-Seiten des Fernstudiengangs Medizinische Informatik

ähnlich wie in Lübeck in interaktiven Praktika und Vorlesungen geschult (siehe [9]).
Des Weiteren wird JAMIP von dem *Johner Institut für IT im Gesundheitswesen* in dem gleichnamigen Masterstudiengang für Übungsaufgaben im Bereich der Medizinischen Bildverarbeitung genutzt (siehe Abbildung 8.3).

8.1.2 Einsatz JAMIPTele

Die Telekonferenzfunktionen des Programms konnten im Rahmen eines Modellprojekts mit einer Vorgängerversion des Systems namens Cypris getestet werden. In zwei Abteilungen des Universitätsklinikums Lübeck, der Klinik für Neurologie und der Klinik für Neurochirurgie, wurde das Programm dabei hinsichtlich seiner Anforderungen an die Bedienbarkeit und Funktionalität, die sich im praktischen Einsatz ergeben, optimiert [1, 2, 3]. Eine Weiterentwicklung wurde daraufhin dann vor allem im Hinblick auf eine Erweiterung der Bildbearbeitungsfunktionalität und eine optimierte Anpassung an die jeweiligen Einsatzbereiche betrieben, die letztendlich zu den beiden Programmversionen JAMIP und JAMIPTele führte.
Die aktuelle Version von JAMIPTele findet sich zu Testzwecken in die *mycare2x-Plattform* des Instituts für Medizinische Informatik integriert. Unter der bisher nur mit Passwort zugänglichen Webseite `http://141.83.20.116:9900/mycare2x` sind auf dieser Open-Source-Plattform verschiedene Lösungen für das Gesundheitswesen aus dem

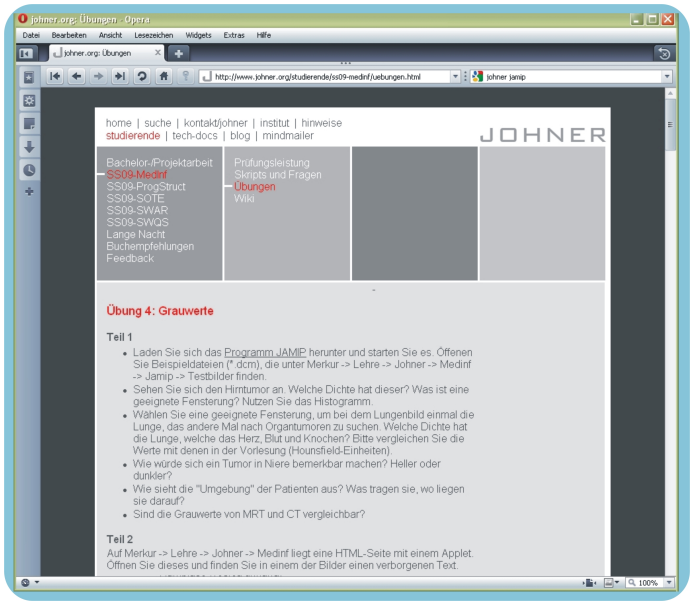


Abbildung 8.3: Übungsaufgabe des Johner Instituts mit JAMIP

Bereich KIS, RIS, PACS zusammengefasst.

JAMIPTele ist hier als Applet an das eingebundene Conquest-PACS-System angeschlossen; ausgewählte Bilder werden direkt in JAMIPTele zu einer Telekonferenz geöffnet. Hier wurde die in Abschnitt 6.2 beschriebene Einsatzmöglichkeit der Ansteuerung des Applets durch die HTML-Seite realisiert. Durch Anklicken eines Bildes wird dabei dem Applet der Dateiname des Bildes übermittelt und JAMIPTele kann mit diesem Bild geöffnet werden. Zwei oder mehr Benutzer der mycare2x-Plattform können so eine telekooperative Besprechung von Bilddaten abhalten, die sich auf diesem gemeinsam genutzten Server befinden. Es ist dann kein vorheriger Bilddatenaustausch mehr vonnöten. Daneben werden die Telekonferenzfunktionen des Programms auch im Rahmen der Vorlesung und Übung *Einführung in die Medizinische Informatik* an der Universität zu Lübeck zu Lehr- und Testzwecken eingesetzt.

Im Zusammenhang mit dem Aufbau einer Telematikinfrastruktur für das deutsche Gesundheitswesen (siehe dazu Abschnitt 2.1.2) bietet sich die Integration von JAMIPTele zur telekooperativen Befundung medizinischer Bilddaten, die z. B. auf einem zentralen Server in einer elektronische Patientenakte abgelegt wurden, an.

8.2 Vergleich

Ein direkt mit JAMIP vergleichbares System lässt sich in der Literatur nicht finden. Das ergibt sich aus den verschiedenen Anwendungsmöglichkeiten des Programms. Lediglich Teile und damit verbundene Einsatzgebiete lassen sich vergleichen.
In Kapitel 4 wurden Softwaresysteme vorgestellt, die Teilaspekte der in dieser Arbeit gestellten Aufgabenstellung erfüllen. So beinhalten die in Abschnitt 4.1 beschriebenen Teleradiologiesysteme oft ein Telekonferenztool. JAMIP ist im Gegensatz dazu jedoch kein Teleradiologiesystem, es entspricht weder vom Anforderungsprofil noch vom Funktionsumfang einem solchen Programm und soll daher hier auch nicht mit einem verglichen werden. JAMIP - insbesondere die Telekonferenzanwendung JAMIPTele - ist eher als Erweiterung eines solchen Systems zu sehen.
Des Weiteren ist JAMIP kein "High-End"-Bildverarbeitungsprogramm für die Medizin und daher auch nicht mit den in Abschnitt 4.4 beschriebenen Bildverarbeitungsprogrammen vergleichbar. Auch wenn der Aufbau der Benutzeroberflächen ähnlich gestaltet ist, so sind die realisierten Funktionen der beschriebenen Programme sehr viel mächtiger als die in JAMIP integrierten Grundlagenmethoden. Für die Lehre oder eine Anwendung als DICOM-Viewer zu Übungszwecken erweisen sich diese Tools als ungeeignet, da sie aufgrund ihrer Komplexität nicht einfach zu bedienen sind. Für eine telekooperative Besprechung medizinischer Bilddaten ist keines der Programme ausgestattet.

JAMIP wird im folgenden Unterabschnitt 8.2.1 zunächst mit frei verfügbaren DICOM-Viewern verglichen, bevor anschließend in Abschnitt 8.2.2 eine Gegenüberstellung von JAMIPTele mit dem kommerziellen Telekonferenztool TeleConsult (siehe dazu 4.2) erfolgt. Zum Schluss wird dann in Abschnitt 8.2.3 der Teilaspekt Lehrsystem betrachtet; JAMIP wird hier anderen Lernplattformen und -programmen für die Medizinische Bildverarbeitung gegenübergestellt.

8.2.1 Vergleich mit DICOM-Viewern

Wie in Abschnitt 4.3 bereits beschrieben gibt es sehr viele - auch kostenfrei zur Verfügung stehende - DICOM-Viewer, die sich in ihrem Funktionsumfang und ihrer Ausrichtung zum Teil erheblich voneinander unterscheiden. So existieren unter diesem Begriff Realisierungen für spezielle medizinische Fragestellungen ebenso wie PACS-Lösungen, ganz einfache Programme ohne weitere Bearbeitungsmöglichkeiten neben aufwendig gestalteten Viewern mit diversen Bildverarbeitungsfunktionen. Es gibt jedoch keinen frei verfügbaren DICOM-Viewer, der für die Lehre konzipiert oder als Telekonferenztool

einsetzbar ist.
JAMIP hingegen fällt genau genommen gar nicht unter den Begriff des DICOM-Viewers - auch wenn mit der Anwendung DICOM-Bilddaten betrachtet werden können, so konzentriert sich die Ausrichtung des Programms auf Lehranwendungen im Bereich der Medizinischen Bildverarbeitung bzw. auf die Ausrichtung von Telekonferenzen. Der implementierte DICOM-Viewer stellt eher einen Prototypen dar, es werden nicht alle Arten von DICOM-Bildformaten wie beispielsweise Filmsequenzen unterstützt, auch sind weitere DICOM-Funktionen wie z. B. das Drucken oder Speichern von Daten auf einer DICOM-CD nicht in das Programm integriert. Dennoch ist JAMIP für seine Ausrichtung als Lehr- oder Übungsprogramm was Aufbau und Umfang der Standardfunktionen angeht durchschnittlichen DICOM-Viewern nachempfunden (siehe dazu 6.1) und kann auch als solcher genutzt werden.
Im Gegensatz zu den meisten kommerziell vertriebenen Viewern und zu einigen frei verfügbaren wie z. B. dem Sante DICOM Viewer Free, die oft mit einer verwirrenden Vielfalt von Funktionen und Darstellungsmöglichkeiten aufwarten, ist JAMIP bewusst so aufgebaut, dass eine Bedienung auch dem ungeübten Nutzer leicht möglich ist. Die Benutzeroberfläche ist in klar voneinander getrennte Bereiche untergliedert und die Funktionen sind übersichtlich in die Menüführung integriert.
Einen weiteren Vorteil gegenüber vielen DICOM-Viewern bietet JAMIP zudem in seiner Plattformunabhängigkeit - nur für wenige Tools gibt es eine Portierung für mehrere Betriebssysteme. JAMIP steht außerdem in zwei verschiedenen Varianten - als Applet oder Anwendung - zur Verfügung, so dass für die Ausführung eine Installation nicht zwingend erforderlich ist. Von den in Tabelle 4.1 aufgelisteten Tools gilt das sonst nur für die eViewBox und das zugehörige eViewBoxApplet - ein sehr schlichtes Programm, was als reines Visualisierungstool genutzt werden kann.

8.2.2 Vergleich mit dem Telekonferenztool TeleConsult

Die in Abschnitt 4.2 beschriebene Software TeleConsult der Firma MedCom ist speziell für die telemedizinische Gesundheitsversorgung entlegener Gebiete wie es sie z. B. in Lateinamerika gibt konzipiert und geht daher einher mit dem Aufbau eines telemedizinischen Netzwerks. Bestandteil des Systems ist eine angeschlossene, zentrale Datenbank, in der Bild- und Befunddaten gespeichert werden und auf die von allen angebundenen Client-Rechnern aus zugegriffen werden kann. JAMIP ist mit seinem Telekonferenzmodul JAMIPTele dagegen für den Einsatz in der deutschen Gesundheitslandschaft gedacht und bewusst ohne solche zusätzlichen Strukturen entwickelt worden, um eine Eingliederung

in bereits bestehende Lösungen wie z. B. Webportale für die integrierte Versorgung (siehe hierzu 4.1) oder zukünftige Einrichtungen wie die elektronische Patientenakte möglichst einfach zu gestalten. Die Plattformunabhängigkeit und die Applet-Version sollen diesen Prozess noch unterstützen. TeleConsult steht im Gegensatz dazu nur für Windows-Rechner zur Verfügung.
Auch weitere Unterschiede zwischen den beiden Programmen ergeben sich aus den unterschiedlichen Einsatzbereichen, so berücksichtigt JAMIP für den Datenversand beispielsweise keine anderen Transportkanäle als das Internet, wohingegen TeleConsult aufgrund mangelnder Verfügbarkeit schneller Netze in Entwicklungsländern auch andere Leitungen wie z. B. analoge Telefonverbindungen als Transportmedium anbietet. Genauso stellen die Integration eines medizinischen Kommentierungswerkzeugs und einer Bilderzeugungssoftware für Ultraschalldaten in das TeleConsult-Tool Anwendungen dar, die speziell auf die Anforderungen in den eingesetzten Projekten in Lateinamerika zugeschnitten sind.
Trotz der Unterschiede, die sich aus den generell verschiedenen Ausrichtungen der Programme ergeben, sind die Systeme was den Aufbau ihrer Benutzeroberflächen und den Telekonferenzmodus angeht ähnlich gestaltet. Beide Programme (siehe dazu Abbildung 4.2) sind übersichtlich und benutzerfreundlich angelegt, haben links eine Übersichtsleiste der eingeladenen Bilder, in der Mitte das Hauptfenster zur Bilddarstellung und rechts Standardfunktionen zur Bildbearbeitung, weitere Funktionen finden sich in den Menüleisten. Im Telekonferenzmodus verfügen beide Applikation über ein Chatfenster, was parallel zur gemeinsam steuerbaren Anwendung zur Kommunikation genutzt werden kann. Während einer Konferenz werden bei beiden Systemen Mausbewegungen und Ereignisse abgefangen und auf die andere Seite übertragen. Es kann entweder auf das zuvor verschickte auf beiden Seiten lokal vorliegende Datenmaterial oder auf das auf einem gemeinsam genutzten Server vorliegende Material zurückgegriffen werden.
Im Gegensatz zu JAMIP handelt es sich bei TeleConsult jedoch um ein kommerzielles Tool.

8.2.3 Vergleich mit Bildverarbeitungstools für die Lehre

Zum Vergleich mit JAMIP werden hier die in Abschnitt 4.5 beschriebenen Tools der Online-Angebote der FH Regensburg, der TU München und der FH Jena herangezogen. Wie JAMIP stehen diese Programme frei oder zumindest als kostenlose Demoversion zur Verfügung und sind für die Lehre in der Bildverarbeitung konzipiert. Es können in unterschiedlichem Umfang die in den zugehörigen Studienmaterialien beschriebenen

Algorithmen ausprobiert werden. Sowohl bei JAMIP wie auch bei den auf den ODITEB-Seiten vorgestellten Verfahren handelt es sich hierbei um Methoden, die speziell in der medizinischen Bildverarbeitung eingesetzt werden, also einen medizinischen Anwendungshintergrund haben. Im Gegensatz dazu ist die Bildverarbeitung in dem Tool IMAGINE der FH Regensburg und dem DBV-E-Learning-Modul der FH Jena eher allgemein ausgerichtet. Nur JAMIP und ODITEB sind auch in Aufbau und Funktion tatsächlichen Systemen der Medizin nachempfunden und in der Lage, DICOM-Daten zu visualisieren. Der Case Viewer von ODITEB konzentriert sich in seinem Funktionsumfang jedoch sehr auf die Diagnostik in der Medizin, wohingegen JAMIP auf die Vermittlung und Anwendung medizinischer Bildverarbeitungsalgorithmen ausgerichtet ist. Wie das DBV-E-Learning-Modul ist der Case Viewer in die zugehörigen Webseiten eingebettet. JAMIP und IMAGINE sind hingegen eigenständig lauffähige Programme, die auch unabhängig von einem Online-Lehrbuch gestartet werden können. Eigene Bilder von der Festplatte oder von einem Server können außer mit JAMIP mit keinem der beschriebenen Programme eingeladen werden, auch Ergebnisse der durchgeführten Operationen lassen sich nur mit JAMIP speichern. Demonstrationen zur Visualisierung des Ablaufs einzelner Verfahren finden sich für ausgewählte Module auf den Online-Seiten der FH Jena und bei JAMIP. Hier ist mit dem bereits beschriebenen Animationsmodus ein extra Menüpunkt enthalten, unter dem die integrierten Algorithmen in ihrem Ablauf betrachtet werden können. Daneben existieren verschiedene Sichten auf die Daten, die die Vermittlung und das Verständnis der Funktionsweise der einzelnen Verfahren erleichtern sollen.

JAMIP ist somit das einzige der hier vorgestellten Programme, was unabhängig von bestehenden Online-Angeboten als eigenständig lauffähiges Tool zur Vermittlung der Grundlagen Medizinischer Bildverarbeitung genutzt werden kann.

Tabelle 8.1 zeigt den Vergleich von JAMIP mit den anderen Tools im Überblick.

	JAMIP	**IMAGINE**	**ODITEB Case Viewer**	**DBV-E-Learning-Modul**
Verfügbarkeit	frei	eingeschränkt	frei	frei
eigenständig lauffähiges Programm	ja	ja	nein	nein
Bildverarbeitung	ja	ja	eingeschränkt	ja
medizinische Ausrichtung	ja	nein	ja	nein
DICOM-Viewer	ja	nein	ja	nein
Einladen eigener Bilder	ja	nein	nein	nein
Ergebnisse speichern	ja	nein	nein	nein
erklärender Modus	ja	nein	nein	teilweise
Hilfe/Anleitung	ja	ja	?	nein

Tabelle 8.1: JAMIP im Vergleich mit anderen Tools für die Lehre der Grundlagen der Bildverarbeitung

9 Fazit und Ausblick

Ziel dieser Arbeit war die Erstellung eines kostenlosen, plattformunabhängigen Bildverarbeitungstools mit telekooperativen Funktionen, das die quasi-parallele Betrachtung und Bearbeitung medizinischer Bilddaten möglichst unabhängig von bestehenden Strukturen und bereits vorhandener Software ermöglicht. Daneben sollte das Programm um Grundlagenalgorithmen aus dem Bereich der Medizinischen Bildverarbeitung ergänzt werden, um als Lehr- und Lernsoftware in Medizin und Informatik eingesetzt werden zu können. Insbesondere sollten die Algorithmen dabei lehr-/lernunterstützend aufbereitet werden.

Dieses Kapitel resümiert im folgenden Abschnitt 9.1 die Ergebnisse dieser Arbeit in Bezug zur Aufgabenstellung, bevor in Abschnitt 9.2 abschließend kurz auf mögliche Erweiterungen und zukünftige Entwicklungen des Programms eingegangen wird.

9.1 Fazit

Mit JAMIP ist im Rahmen dieser Arbeit ein frei verfügbares Werkzeug für die Betrachtung und Bearbeitung medizinischer Bilddaten im DICOM-Format sowie für deren synchrone telekooperative Besprechung zum Einsatz in Forschung, Lehre und Praxis entstanden, das insbesondere dem Einfluss der neuen Technologien im Gesundheitswesen und der Lehre Rechnung trägt.
Einsatz- und Anwendungsszenarien finden sich für das Tool bislang hauptsächlich in der Lehre, JAMIP konnte bereits in verschiedenen Veranstaltungen für die Fern- und Präsenzlehre der Medizinischen Bildverarbeitung erfolgreich eingesetzt werden. Daneben konnte JAMIPTele, die Telekonferenzanwendung des Systems, als Applet in die Gesundheitsplattform des Instituts für Medizinische Informatik integriert werden, weitere Einbindungen im Rahmen klinik- oder abteilungsübergreifender Serverlösungen sind denkbar.

Um die Anforderungen des ersten Teils der Zielsetzung dieser Arbeit zu erfüllen, wurde als Basisprogramm ein Bildverarbeitungstool implementiert, was medizinische Bilddaten

im DICOM-Format anzeigen und mit Standardfunktionen wie dem Zoomen, der Invertierung oder der Fensterung bearbeiten kann. Vorlage für die Gestaltung und die Auswahl der Funktionen waren hier übliche in der Medizin zur Bildvisualisierung eingesetzte Programme wie z. B. DICOM-Viewer wie sie in Abschnitt 4.3 beschrieben werden. Dabei wurden für die Gewährleistung der Plattformunabhängigkeit zur Implementierung des Systems die Programmiersprache Java sowie ausschließlich Java-basierte Zusatzbibliotheken benutzt. JAMIP ist damit vollständig Java-basiert und so auf jedem handelsüblichen PC ausführbar. Es steht in zwei Anwendungsarten - als Applet und als Application - zur Verfügung und kann daher entweder lokal von einer Festplatte oder auf einem zentralen Server gestartet werden. Das Tool wurde unabhängig von anderer Software als flexibel einzusetzendes Bildverarbeitungsprogramm realisiert, um so eine Eingliederung in bereits bestehende Lösungen wie z. B. Webportale für die integrierte Versorgung (siehe hierzu Abschnitt 4.1) oder in zukünftige Einrichtungen der Telematikinfrastruktur (siehe dazu Abschnitt 2.1.2) wie die elektronische Patientenakte leicht zu ermöglichen.
Die Realisierung der geforderten telekooperativen Funktionen des Programms erfolgte in dem eigenständigen Modul JAMIPTele, was eine telekonferenzfähige Variante von JAMIP darstellt und die quasi-parallele Betrachtung und Bearbeitung medizinischer Bilddaten sowie die Kommunikation der Teilnehmer per Texteingabe über ein Chatfenster ermöglicht. Da die Durchführung einer Telekonferenz das Vorhandensein der zu besprechenden Bilddaten auf einem gemeinsam zugänglichen Server oder lokal bei allen Teilnehmern erfordert, wurde zusätzlich ein Modul für den verschlüsselten Datenversand in das System integriert. So kann der Einsatz von JAMIP auch unabhängig von bestehenden Strukturen und Anwendungen wie Webportalen oder DICOM-E-Mail gestaltet werden.
Das so realisierte Programm ist das einzige frei verfügbare Tool, mit dem medizinische Bilddaten außerhalb komplexer Teleradiologielösungen in telekooperativen Sitzungen betrachtet und bearbeitet werden können.

Um JAMIP als Lehr- und Lernsoftware im Bereich der Medizinischen Bildverarbeitung einsetzten zu können und damit die Anforderungen des zweiten Teils der Zielsetzung zu realisieren, wurden vorlesungsrelevante Bildverarbeitungsmethoden aus dem Grundlagenbereich in das System integriert. Neben ihrer “normalen” Ausführung stehen die meisten Algorithmen dabei auch im sogenannten Animationsmodus zur Verfügung. Zur Unterstützung des Lehrenden bei der Präsentation eines Verfahrens sowie des Lernenden beim Verständnis wird hier ein Algorithmus in seine Einzelschritte unterteilt und so in seinem Ablauf visualisiert. Außerdem besteht in den meisten Fällen die Möglichkeit durch unterschiedliche Sichten auf die zugrundeliegenden Daten oder mittels diverser

Optionen zur eingehenden Auseinandersetzung mit dem Verfahren. Dabei ist anzumerken, dass sich die Umsetzung dieses Modus nicht für alle Verfahren gleich gut bewerkstelligen ließ. Aufgrund der Unterschiedlichkeit der Verfahren war hier ein einheitliches Vorgehens für die Visualisierung des Ablaufs leider nicht möglich.
Für die Integration in den Fernstudiengang Medizinische Informatik der FernUniversität Hagen, wurden einzelne Bildverarbeitungsalgorithmen von JAMIP als separate Applets realisiert und in die Webseiten der FernUniversität miteingebunden. Daneben steht das Tool hier genau wie an der Universität zu Lübeck für die Studenten zum Download zur Verfügung. JAMIP stellt so eine gute Ergänzung zu Büchern und Vorlesungsunterlagen dar und ist trotz der großen Anzahl mittlerweile verfügbarer E-Learning-Materialien das einzige Programm, was auch außerhalb bestehender Online-Angebote zur Vermittlung der Grundlagen Medizinischer Bildverarbeitung genutzt werden kann (siehe dazu Abschnitt 8.2.3).
Im praktischen Einsatz hat sich das Tool bereits als gut geeignete Lehr- und Lernsoftware sowohl im Informatik- als auch im Medizinstudium erwiesen (siehe dazu Abschnitt 8.1.1). Hier konnten einerseits Anwendungskenntnisse für den Umgang mit Bildverarbeitungs- und Bildkommunikationssoftware sowie andererseits besagte Bildverarbeitungsmethoden und deren Ablauf vermittelt werden.

Insgesamt konnten damit die in der Zielsetzung definierten Anforderungen an das im Rahmen dieser Arbeit zu entwickelnde System erfüllt werden.

9.2 Ausblick

Wie im vorangegangenen Abschnitt bereits beschrieben befindet sich JAMIP vor allen Dingen als Lehr- und Lerntool im praktischen Einsatz. In diesem Zusammenhang wäre eine Weiterentwicklung des Systems durch den Einbau neuer Module für die Bildverarbeitung denkbar. Für die Ausbildung von Informatikstudenten könnten hier neben komplexeren Verfahren beispielsweise 3D-Tools integriert werden oder für Medizinstudenten die Möglichkeit, DICOM-Videos zu betrachten.

Für den Einsatz als Telekonferenztool ist mit JAMIPTele in dieser Arbeit ein Prototyp entstanden, der im klinischen Alltag erst noch getestet werden muss. Hier ergeben sich gerade im Zusammenhang mit dem Aufbau einer Telematikinfrastruktur für das deutsche Gesundheitswesen (siehe dazu Abschnitt 2.1.2) verschiedene Möglichkeiten zur Integration des Tools, so könnte JAMIPTele z. B. zur telekooperativen Befundung medizinischer

Bilddaten, die in einer elektronischen Patientenakte oder auf einem Webserver abgelegt wurden, benutzt werden.

Als Erweiterung dieses Programmteils wäre der Einbau einer Funktion zur Regelung des Aktionsrechtes denkbar. Des Weiteren könnten die Bearbeitungsfunktionen von JAMIPTele ausgebaut und dem Umfang von denen in JAMIP angepasst werden. Außerdem könnte das Telekonferenztool noch intensiver in Vorlesungen und Übungen integriert werden, z. B. für ein telemedizinisches Training für Mediziner.
Erweiterungsmöglichkeiten ergeben sich auch in Bezug auf den integrierten DICOM-Viewer. JAMIP und JAMIPTele bieten hier bislang nur eine eingeschränkte DICOM-Funktionalität, die im Wesentlichen auf die Vermittlung der Lehrinhalte ausgerichtet ist. Es können bisher nicht alle Arten von DICOM-Bildern angezeigt werden, auch bietet das System keine PACS-Unterstützung oder die Möglichkeit für DICOM-E-Mail oder DICOM-Druck. Da es mittlerweile verschiedene Java-Bibliotheken für den Umgang mit DICOM-Daten gibt, könnte der hier entwickelte Einleseprozess durch diese APIs ergänzt werden.

Literaturverzeichnis

[1] H. Schmidt, H. Handels, T. Schößler, U. Knopp, G. Seidel, S. J. Pöppl; “Kryptographischer Versand und kooperative Befundung medizinischer Bilddaten in Java”; G. Steyer, M. Engelhorn, W. Fabricius, K. P. Löhr, T. Tolxdorff (Hrsg.); Tagungsband zur Telemed 2001; FU Berlin; 111-116

[2] H. Schmidt, H. Handels, U. Knopp, G. Seidel, S. J. Pöppl; “Bildgestützte Telediagnostik und 3D-Teleimaging in Java”; M. Meiler, D. Saupe, F. Kruggel, H. Handels, T. Lehmann (Hrsg.): Bildverarbeitung für die Medizin 2002; Informatik aktuell; Springer Verlag; 225-228

[3] H. Schmidt, H. Handels, T. Schößler, U. Knopp, G. Seidel, S. J. Pöppl; “Cypris - Java-basierte Telediagnostik und Teleimaging”; A. Jäckel (Hrsg.): Telemedizinführer 2003; Medizin Forum Bad Nauheim; 180-183

[4] H. Schmidt, H. Handels, K. Schwidrowski, O. Schmidt, T. Hahn, S. J. Pöppl; “Java Based Medical Image Processing for Distance Learning and Teleteaching”; Steyer G, Löhr K.P., Tolxdorff T. (Hrsg.), Tagungsband zur Telemed 2002, FU Berlin, 198-202

[5] H. Schmidt, H. Handels, K. Schwidrowski, O. Schmidt, T. Hahn, O. Burmeister, M. Busse, A. Maciak, S. J. Pöppl; “Das virtuelle Bildverarbeitungslabor JAMIP”; Informatik, Biometrie und Epidemiologie in Medizin und Biologie, GMDS-Tagungsband, September 2003, Abstraktnr.44/3, 412-414

[6] H. Schmidt, H. Handels, O. Burmeister, M. Busse, T. Hahn, O. Schmidt und S. J. Pöppl; “JAMIP - ein virtuelles Bildverarbeitungslabor für das Fernstudium der Medizinischen Informatik”; Jäckel (Hrsg.): Telemedizinführer 2004, ISBN 3-9808837-1-X, 185-190

[7] H. Schmidt, H. Handels, T. Hahn, A. Maciak, O. Schmidt, K. Schwidrowski und S. J. Pöppl; “JAMIP - Entwicklung eines virtuellen Bildverarbeitungslabors für die Lehre”; BVM-Tagungsband 2004, ISBN 3-540-21059-8, 400-404

[8] H. Schmidt, T. Bußhaus, J. Ingenerf, G. Katalinic, R. Linder, S. J. Pöppl und R. Stöckelhuber; "Querschnittsbereich Medizinische Informatik: Praxisbezogene Medizinerausbildung am Beispiel der Medizinischen Bildverarbeitung"; Wichmann et al. (Hrsg.), Kongress Medizin und Gesellschaft, Abstractband, ISBN 3-938975-12-1, 305

[9] M. Riemer, W. Hampe, M. Wollatz, C. Peimann, H. Handels; "Erste Erfahrungen mit der eLearning-Plattform Moodle im Universitätsklinikum Hamburg-Eppendorf - Evaluationsergebnisse im Querschnittsfach Medizinische Informatik und Biochemie"; Kundt G., Bernauer J., Fischer M., Haag M., Klar R., Leven J., Matthies H., Puppe F. (Hrsg.), eLearning in der Medizin, Proceedings, CBT 2007, Rostock, 235-246

[10] M. Herczeg; "Einführung in die Medieninformatik"; Oldenbourg Wissenschaftsverlag GmbH 2007, ISBN 3-486-58103-1

[11] Projektgruppe ARD/ZDF-Multimedia; "Internet zwischen Hype, Ernüchterung und Aufbruch - 10 Jahre ARD/ZDF-Onlinestudie"; http://www.ard-zed-onlinestudie.de, 2007

[12] K. Jähn, E. Nagel; "e-Health"; Springer-Verlag Berlin Heidelberg 2004; ISBN 3-540-43937-4

[13] P. Haas; "Gesundheitstelematik: Grundlagen, Anwendungen, Potenziale"; Springer-Verlag Berlin Heidelbergi 2006; ISBN 978-354020740-5

[14] C. Link; "Telemedizinische Anwendungen in Deutschland und Frankreich"; Herbert Utz Verlag 2007; ISBN 978-3-8316-0731-0

[15] U. Engelmann, H. Münch, A. Schröter, T. Schweitzer, K. Christoph, R. Eilers, H. Olesen, J. Møller Jensen, H.P. Meinzer; "A teleradiology concept for entire greenland"; Int. J CARS (2006); 1:121-123

[16] G. Weisser, M. Walz; "Grundlagen der Teleradiologie"; Der Radiologe; Springer Medizin Verlag; Februrar 2007

[17] A. Delvos; "Die elektronische Gesundheitskarte - Einführung in das deutsche Gesundheitswesen"; GRIN Verlag 2008; ISBN 978-3-640-29480-0

[18] Bundesministerium für Gesundheit; "Die elektronische Gesundheitskarte"; http://www.die-gesundheitskarte.de

[19] Konferenz der Datenschutzbeauftragten des Bundes und der Länder; “Datenschutz und Telemedizin - Anforderungen an Medizinnetze”; Der Bundesbeauftragte für den Datenschutz und die Informationsfreiheit, Arbeitsgruppe Telemedizin, Stand 10/2002; http://www.bfdi.bund.de

[20] C. Eckert; “IT-Sicherheit: Konzepte - Verfahren - Protokolle”; Oldenbourg Wissenschaftsverlage GmbH 2008; ISBN 978-3-486-58270-3

[21] S. Garfunkel; “PGP: Pretty Good Privacy”; O’Reilly 1995; ISBN 1-56592-098-8

[22] U. Engelmann, B. Schütze, A. Schröter, G. Weisser, M. Walz, M. Kämmerer, P. Mildenberger; “Teleradiologie per DICOM-E-Mail: Der empfohlene Minimalstandard der Deutschen Röntgengesellschaft”; G. Steyer, T. Tolkxdorf (Hrsg.): Proceedings Telemed 2005. bit for bit - Halbzeit auf dem Weg zur Telematikinfrastruktur. Akademischer Verlagsanstalt Berlin 2005; ISBN 3-89838-064-5; S. 102-107

[23] J. Domingo-Ferrer, J. Posegga, D. Schreckling (Hrsg.); “Smart card research and advanced applications”; 7th IFIP WG 8.8/11.2 International Conference, CARDIS 2006, Tarragona, Spain, April 19-21, 2006; Proceedings Lecture Notes in Computer Science; Springer-Verlag Berlin Heidelberg; ISBN 978-3-540-33311-1

[24] Bundesamt für Sicherheit in der Informationstechnik;“BSI TR-03116: Technische Richtlinie für eard-Projekte der Bundesregierung”; April 2009; http://www.bsi.bund.de

[25] M. Kerres, B. Voß (Hrsg.); “Digitaler Campus: Vom Medienprojekt zum nachhaltigen Medieneinsatz in der Hochschule”; Waxmann Verlag GmbH, Münster 2003, ISBN 3-8309-1288-9

[26] D. Euler, S. Seufert (Hrsg.); “E-Learning in Hochschulen und Bildungszentren”; Oldenbourg Wissenschaftsverlag GmbH 2005, ISBN 3-486-20008-9

[27] M. Kerres; “Multimediale und telemediale Lernumgebungen: Konzeption und Entwicklung”; Oldenbourg Wissenschaftsverlag GmbH 2001, ISBN 3-486-25055-8

[28] M. Kerres; “Didaktische Konzeption multimedialer und telemedialer Lernumgebungen”; HMD - Praxis der Wirtschaftsinformatik (1999)

[29] A. Gussenstätter; “eLearning Glosar”; http://www.foraus.de/download/elearn_tipps/glossar030801.pdf

[30] B. Gross Davis; “Tools for Teaching - 2nd edition”; John Wiley & Sons 2009; ISBN 978-0-7879-6567-9

[31] J. Cole, H. Foster; “Using Moodle - 2nd edition”; O’Reilly Media Inc. 2008; ISBN 978-0-596-52918-5

[32] M. Gevantmakher, C. Meinel; “Medizinische Bildverarbeitung - eine Übersicht”; Forschungsgruppe Institut für Telematik, Universität Trier; http://www.hpi.uni-potsdam.de/fileadmin/hpi/FG_ITS/papers/TR_Med_Bildverarbeitung.pdf

[33] T. Lehmann: “Bildverarbeitung für die Medizin : Grundlagen, Modelle, Methoden, Anwendungen”; Springer-Verlag Berlin Heidelberg; 1997; ISBN 3-540-61458-3

[34] H. Handels; “Medizinische Bildverarbeitung”; Teubner Verlag; 2000; ISBN 3-519-02947-2

[35] P. A. Rinck, S. B. Petersen, R. N. Muller; “Magnetresonanz in der Medizin”; Georg Thieme Verlag Stuttgart, New York; ISBN 978-3-936072-13-6

[36] E. Oberhausen, R. Berberich, G. Brill, B. Glöbel, R. Kunkel; “Technik der Nuklearmedizin”; Deutscher Ärzte Verlag; ISBN 3-7691-1089-7

[37] P. Haberäcker; “Praxis der Digitalen Bildverarbeitung und Mustererkennung”; Hanser Verlag; 1995; ISBN 3-446-15517-1

[38] H. Oosterwijk, P. T. Gihring;“DICOM Basics, 3nd ed.”; OTech, Inc., Aubrey, TX; 2002; ISBN 0-9718867-4-1

[39] A. Metzner; “Konzepte des DICOM-Standards und DICOM Structured Reporting”; Seminararbeit Technische Universität Berlin 2001; http://uebb.cs.tu-berlin.de/~ame/files/uncompressed/dicom-seminar.pdf

[40] National Electrical Manufacturers Association: “Digital Imaging and Communication in Medicine (DICOM). Part 1-16.”; Rosslyn, Virginia, USA; http://medical.nema.org/dicom/2008

[41] L. Bruder; “Konvertierung von DICOM-Dateien in eine XML-Transfersyntax”; Diplomarbeit; Fachhochschule Heilbronn/Universität Heidelberg, Sommersemester 2005; http://vsm.mi.hs-heilbronn.de/pub/doc/theses/diploma/leif-bruder.pdf

[42] S. Fischer, A. El Saddik; “Open Java - Von den Grundlagen zu den Anwendungen”; Springer; ISBN 3-540-65446-1

[43] S. Oaks; “Java Security”; O’Reilly; ISBN 1-56592-403-7

[44] W. Diffie, M.E. Hellman; “New directions in cryptography”; IEEE Transactions on Information Theory 22 (1976), 644-654

[45] R. Rivest, A. Shamir, L. Adleman;“A method for obtaining Digital Signatures and Public Key Cryptosystems.” Communications of the ACM. Feb., 1978 21(2) pages 120-126; http://people.csail.mit.edu/rivest/Rsapaper.pdf

[46] J. Knudsen; “Java Cryptography”; O’Reilly; ISBN 1-56592-402-9

[47] H. Handels, C. Busch, J. Encarnacao, C. Hahn, V. Kühn, J. Miehe, S. J. Pöppl, E. Rinast, C. Roßmanith, F. Seibert, A. Will; “KAMEDIN: A Telemedicine system for Computer Supported Cooperative Work and Remote Image Analysis in Radiology”; Computer Methods and Programs in Biomedicine 1997; 52: 175-183

[48] C. Hahn, H. Handels, E. Rinast, C. Busch, M. Groß, J. Miehe, S. Nowacki, A. Will, K. Rösler, H. Putzar;“Telekonferenzen über ISDN mit dem Softwaresystem KAMEDIN”; Digitale Bildverarbeitung in der Medizin, Universität Freiburg, 1994

[49] U. Engelmann, A. Schröter, U. Baur, M. Schwab, H. Müller, M. Bahner, H. P. Meinzer; “Second Generation Teleradiology”; Lemke HU, Vannier MW, Inamura K (eds): Computer Assisted Radiology and Surgery; Amsterdam: Elsevier (1997) 632-637

[50] K. Ludwig, U. Bick; “Kommerzielle Teleradiologiesysteme in Deutschland”; Der Radiologe, Springer-Verlag 1997; 37:336-339

[51] U. Engelmann, A. Schröter, M. Schwab, U. Eisenmann, M. Vetter, J. Quiles, M. L. Bahner, H. P. Meinzer; “Radiologieworkstation CHILI”; Jäckel (Hrsg.)Telemedizinführer Deutschland, Bad Nauheim, Ausgabe 2000; 271-276

[52] U. Engelmann, H. Münch, A. Schröter, H. P. Meinzer; “Teleradiologie-Konzepte der letzten 10 Jahre am Beispiel von Chili”; Jäckel (Hrsg.) Telemedizinführer Deutschland, Bad Nauheim, Ausgabe 2008; 242-248

[53] U. Engelmann, H. Münch, C. Bohn, A. Schröter, H. P. Meinzer; “Eine DICOM-basierte Telemedizinakte”; Jäckel (hrsg.) Telemedizinführer Deutschland, Bad Nauheim, Ausgabe 2000; 222-225

[54] F. Unglauben, W. Hillen, Th. Kondring; “Java DICOM Viewer für die Teleradiologie”; Bildverarbeitung für die Medizin 2001, Springer Verlag 2001

[55] W. Menary, I. Sachpazidis, D. Rizou, J. Cardenas, C. Cabrera; “MEDical NETwork MedNET helps rural Peru access health care services with Telemedicine via satellite.”; E-Health Magazine August 2008; http://www.ehealthonline.org

[56] W. S. Rasband; “ImageJ”; U. S. National Institutes of Heakth, Bethesda, Maryland, USA http://rsb.info.nih.gov/ij/; 1997-2009

[57] M.D. Abramoff, P. J. Magelhaes, S. J. Ram; “Image Processing with ImageJ”; Biophotonics International, Volume 11, Issue 7, 36-42, 2004

[58] A. Horsch, T. Balbach, S. Melnitzki, J. Knauth; “Learning tumor diagnostics and medical image processing via the WWW–the case-based radiological textbook ODITEB”; International journal of medical informatics; September 2000; 58-59:39-50

[59] J. P.Lewis, U. Neumann; “Performance of Java versus C++”; Computer Graphics and Immersive Technology Lab, University of Southern California; http://scribblethink.org/Computer/javaCbenchmark.html

[60] C. Ullenboom; “Java ist auch eine Insel - Programmieren mit der Java Standard Edition Version 6”; Galileo Computing; 8. aktualisierte Auflage 2009; ISBN 978-3-8362-1371-4; http://openbook.galileocomputing.de/javainsel8/

[61] J. E. White; “RFC 707: A High-Level Framework for Network-Based Resource Sharing”; Proceedings of the 1976 National Computer Conference

[62] T. J. Mowbray, W. A. Ruh;“Inside CORBA. Distributed Object Standards and Applications”; Addison-Wesley Longman, Amsterdam; September 1997; ISBN 978-0201895407

[63] G. Knopp, T. Stark; “Handbuch der JAVA-Programmierung: Standard Edition Version 6”; Addison-Wesley-Verlag 2009; ISBN 978-3-8273-2874-8

[64] N. Walsh; “DocBook 5.0: The Definite Guide”; O’Reilly; ISBN 13: 9781565925809; http://www.docbook.org/tdg5/en/html/docbook.html

[65] B. Stayton; “DocBook XSL: The Complete Guide”; Fourth Edition, 2007; Sagehill Enterprises; ISBN-13: 978-0974152134; http://www.sagehill.net/docbookxsl/index.html

[66] H. Handels, T. Schößler, J. Ehrhardt, H. C. Klaiber, S. J. Pöppl; “Ein CSCW-Tool für die kooperative Bildbesprechung in Telekonferenzen unter Java”; Jäckel, A. (Hrsg.), Telemedizinführer 2001; Medizin Forum, Bad Nauheim, 220-223

[67] T.Schößler;“Java-basierte Telekooperation mit 3D-Visualisierungen medizinischer Bilddaten”; Diplomarbeit; Institut für Medizinische Informatik, Universität zu Lübeck; März 2000

[68] M. Herczeg; “Softwareergonomie”; Oldenbourg Wissenschaftsverlag GmbH 2005, ISBN 3-486-25052-3

[69] M. Kass, A. Witken, D. Terzopolous; “Snakes: Active contour models”; International Journal of Computer Vision; 1988; S 321-331

Abbildungsverzeichnis

2.1 Die digitale Revolution im zeitlichen Überblick 13
2.2 Das Wachstum des Internets 14
2.3 Stufenweise Einführung der elektronischen Gesundheitskarte 18
2.4 Aufbau der Telematikinfrastruktur 19
2.5 moodle am Institut für Medizinische Informatik 25

3.1 Bereiche der medizinischen Bildverarbeitung 28
3.2 Zeitliche Entwicklung der wichtigsten bildgebenden Verfahren 29
3.3 Beispiele für Röntgen-, CT-, MRT- und US-Bilder 32
3.4 Übersicht über weitere bildgebende Verfahren 33
3.5 Orts- und Intensitätsauflösung bei Bildern 34
3.6 Phasen eines Bildanalyse-Systems 36
3.7 Bildverwaltung im Krankenhaus 37
3.8 Der DICOM-Standard im Überblick 38
3.9 Der Aufbau eines DICOM-Bildes 40
3.10 Der Aufbau eines DICOM-Files 40
3.11 Der Aufbau eines DICOM Data Elements 41
3.12 Das DICOM Data Element in Abhängigkeit von der VR 42
3.13 Die Pixelzelle in DICOM 43
3.14 Symmetrische Verschlüsselung 45
3.15 Asymmetrische Verschlüsselung 45
3.16 Hybride Verschlüsselung 46

4.1 Webportal zur Bereitstellung von Bildern 51
4.2 Die Softwareapplikation TeleConsult 52
4.3 Beispiele für verschiedene DICOM-Viewer 53
4.4 Beispiele für komplexe Bildverarbeitungsprogramme 56
4.5 Das IMAGINE-Tool der FH Regensburg 58
4.6 Die ODITEB-Seiten für Informatiker 59

4.7 Das E-Learning-Modul der FH Jena 60
4.8 Der Demonstrationsmodus für lokale Operatoren bei der FH Jena 60

5.1 Die Ausführung eines Java-Programms 62
5.2 Das Modell des BufferedImage 63
5.3 Der Ablauf bei RMI 64
5.4 Die Einbindung von Jermes 69
5.5 Die Kommunikationsarchitektur bei Jermes 69
5.6 Der Ablauf einer Telekonferenz 70

6.1 Das Klassendiagramm des Basissytems von JAMIP 73
6.2 Der Einleseprozess für DICOM-Dateien in JAMIP 76
6.3 Das Klassendiagramm der Hauptklassen von JAMIPTele 79
6.4 Paketdiagramm von JAMIP 85

7.1 Die Webseite des Projekts 88
7.2 Zustimmung zur Ausführung des Applets erforderlich 89
7.3 Die Benutzeroberfläche von JAMIP 90
7.4 Die Bedienungsanleitung von JAMIP 92
7.5 Ansicht eines Headers in JAMIP 94
7.6 Beispiele für Abstand, Mittelwert und Skala in JAMIP 95
7.7 Die Fensterung/Level-Window-Operation 96
7.8 Das Histogramm in JAMIP 99
7.9 Die Histogrammerstellung im Animationsmodus 100
7.10 Die Erzeugung eines Profils in JAMIP 101
7.11 Die Grauwerttransformation in JAMIP 102
7.12 Aufbau einer Maske 103
7.13 3x3-Masken des Mittelwert- und des Gaußfilters 103
7.14 Das Filter-Modul in JAMIP 104
7.15 Der Animationsmodus bei der Filterung 105
7.16 Profilansicht und Numerische Ansicht bei der Filterung 105
7.17 Weitere Ansichten bei der Filterung 106
7.18 Das Steuerungsfenster für das Schwellwertverfahren 107
7.19 Das Steuerungsfenster für das Bereichswachstumsverfahren 109
7.20 Der Animationsmodus beim Bereichswachstumsverfahren 110
7.21 Segmentierungsergebnis eines Volumenwachstumsverfahrens auf einer CT-Serie 111

7.22 Ablauf der histogrammbasierten Segmentierung 112
7.23 Die histogrammbasierte Segmentierung 113
7.24 Die Snakes in JAMIP 114
7.25 Die K-Means-Clusteranalyse in JAMIP 116
7.26 Segmentierungsergebnisse unterschiedlicher Verfahren in JAMIP 116
7.27 Beispiel für eine Statistische Texturanalyse mit Cooccurrence-Matrix . 117
7.28 Das Modul Statistische Texturanalyse mit Cooccurrence-Matrizen in JAMIP 118
7.29 Die Texturanalyse im Animationsmodus 119
7.30 Die Eingabemaske für den Datenversand 120
7.31 Einstellung der Verschlüsselung in JAMIP 121
7.32 Anmeldung zur Telekonferenz 123
7.33 Das Kontrollfenster 124
7.34 Die Benutzeroberfläche von JAMIPTele 125
7.35 Das Chatfenster von JAMIPTele 127
7.36 Das Schwellwertverfahren während einer Telekonferenz 128

8.1 Einsatz von JAMIP an der Uni Lübeck 130
8.2 Das Bereichswachstumsverfahren an der FernUni Hagen 132
8.3 Übungsaufgabe des Johner Instituts mit JAMIP 133

Tabellenverzeichnis

3.1 Grauwertquantisierungsstufen 34
3.2 Beispiele aus dem Data Dictionary 41
3.3 Beispiele für die Transfersyntax in DICOM 42

4.1 Eine Übersicht frei verfügbarer DICOM-Viewer 54

6.1 Die verwendete Software 86

8.1 JAMIP im Vergleich mit anderen Tools für die Lehre 138

Abkürzungsverzeichnis

API Application Programming Interface

AWT Abstract Window Toolkit

Bmbf Bundesministerium für Bildung und Forschung

BSI Bundesministerium für Sicherheit in der Informationstechnik

CORBA Common Object Request Broker Architecture

DCOM Distributed Component Object Model

DES Data Encryption Standard

DICOM Digital Imaging and Communications in Medicine

gematik Gesellschaft für Telematikanwendungen der Gesundheitskarte mbH

GUI Graphical User Interface

IDEA International Data Encryption Algorithm

IIOP Internet Inter-ORB Protocol

IMI Institut für Medizinische Informatik

IuK-Technologien Informations- und Kommunikationstechnologien

JAMIP Java based Medical Imaging

JCA Java Cryptography Architecture

JCE Java Cryptography Extension

JVM Java Virtual Machine

KIS Krankenhausinformationssystem

medin Multimediales Fernstudium Medizinische Informatik

PACS Picture Archiving and Communication System

PGP Pretty Good Privacy

RIS Radiologieinformationssystem

RMI Remote Method Invocation

RPC Remote Procedure Call

WWW World Wide Web

Links

Aeskulap: http://aeskulap.nongnu.org

Aktionsprogramm Telemedizin in NRW: http://www.egesundheit.nrw.de

Bounty Castle Crypto APIs: http://www.bouncycastle.org/

Bundesamt für Sicherheit in der Informationstechnik : http://www.bsi.bund.de

CarDiCon: http://fhs-consulting.com/cardicon

CHILI: http://www.chili-radiology.com/de/

Clear Canvas Workstation: http://www.clearcanvas.ca

Cryptix: http://www.cryptix.org/

ConQuest: http://www.xs4all.nl/~ingenium/dicom.html

CORBA: http://www.omg.org/cgi-bin/doc?formal/04-03-12.pdf

DICOM3-Standard: http://medical.nema.org/dicom/2008

DicomLite: http://www.3d-head.com

DICOMScope: http://dicom.offis.de/dscope

DicomWorks: http://www.dicomworks.com

DobuDish: http://www.agynamix.de/products/dobudish/

DocBook-Standard: http://www.docbook.org/whatis

Eclipse: http://www.eclipse.org/

eViewBox und eViewBoxApplet: http://eviewbox.sourceforge.net

ezDICOM: http://www.cabiatl.com/micro/ezdicom

Fat Jar Plugin: http://sourceforge.net/projects/fjep/

Firefox: http://www.firefox.com

FP Image: http://www.fpimage.com

gematik: http://www.gematik.de

I-do-Imaging-Portal: http://www.idoimaging.com

ImageJ: http://rsbweb.nih.gov/ij/

Imread: http://www.uchsc.edu/sm/neuroimaging/download/imread/imread1.htm

iRAD: http://irad.sourceforge.net

JAMIP: http:www.imi2.uni-luebeck.de/jamip

JAMIPTele: http:www.imi2.uni-luebeck.de/jamip/jamiptele

Java: http://java.sun.com

Java Advanced Imaging: http://java.sun.com/products/java-media/jai/iio.html

JavaHelp: https://javahelp.dev.java.net/

JiveX: http://www.jivex.de/

JiveX DICOM Viewer: http://www.visus.com

Johner Institut für IT im Gesundheitswesen: http://www.johner-institut.de

Kradview: http://www.orcero.org/irbis/kradview

Madena: http://www.eyephysics.com/Madena

MEDISP DICOM Viewer: http://www.teiath.gr/stef/tio/medisp

MedNET: http://www.e-mednet.org

MeVisLab: http://www.mevis.de

MicroDicom Viewer: http://www.microdicom.com

MIPAV: http://mipav.cit.nih.gov

MIView: http://gbooksoft.com

Moodle-Plattform des IMI: http://moodle.imi.uni-luebeck.de/moodle/

mycare2x: http://www.hccgmbh.com/index.html

mycare2x-Plattform des IMI: http://141.83.20.116:9900/mycare2x/

NetBeans: http://netbeans.org/

Neue Medien in der Bildung: http://www.bmbf.bund.de/pub/neue_medien_bildung.pdf

Osiris: http://www.sim.hcuge.ch/osiris

OsiriXLite: http://www.osirix-viewer.com

PaxeraViewer: http://www.paxeramed.com

PixelMed Java DICOM Toolkit: http://www.pixelmed.com

Radscaper: http://www.divinev.com/radscaper

Rubo Medical Imaging Dicom Viewer: http://www.rubomed.com

Sante DICOM Viewer: http://www.santesoft.com

Sante DICOM Viewer Free: http://www.santesoft.com/dicom_viewer_free.html

SimpleDicom Viewer: http://www.upmc.com

Sobox Image Viewer: http://www.sobox.it

Synedra View Personal: http://download.synedra.com

TeleConsult: http://www.medcom-online.de

Telekolleg des BR: http://www.br-online.de/wissen-bildung/telekolleg

Thawte: http://www.thawte.com/

TomoVision: http://www.tomovision.com

Tudor DICOM Viewer: http://santec.tudor.lu/project/optimage/dicom

UML2.0-Notation: http://www.uml.org

VerigSign: http://www.verisign.com

VIMED: http://www.vimed.de

XMLSpy: http://www.altova.com/xml-editor

Danksagung

An dieser Stelle möchte ich mich bei allen Personen bedanken, die mich bei der Erstellung dieser Arbeit unterstützt haben.
Insbesondere gilt mein Dank meinem Doktorvater Herrn Prof. Dr. Dr. Pöppl für die Betreuung meiner Arbeit und den Kampf um Drittmittelprojekte sowie für das entgegengebrachte Vertrauen und den für die Fertigstellung nötigen Freiraum.
Daneben möchte ich mich bei Herrn Prof. Dr. Dosch für die tatkräftige Unterstützung, die vielen Ratschläge und ebenso wie bei Herrn Prof. Dr. Mertins für die Gewährung eines Werkvertrags und die Bereitstellung des Arbeitsplatzes bedanken.
Herrn Prof. Dr. Handels gilt mein Dank für die Initiierung des Themas und meinem Zweitgutachter Herrn Prof. Dr. Herczeg für seine Bereitschaft, meine Arbeit zu begutachten.
Außerdem möchte ich mich ganz herzlich bei Herrn PD Dr. Ingenerf für seine stetige Unterstützung und Zuverlässigkeit, seine fundierte Hilfe und seine sehr wertvollen Anregungen für das Abfassen meiner Arbeit bedanken.
Frau Dr. Seik danke ich für die nette Unterstützung in Fragen der mycare2x-Plattform sowie für Ratschläge aller Art bezüglich Promotionsarbeiten. Meiner Bürogenossin Frau Kulbe und meinem Bruder Michael gebührt mein besonderer Dank für den aufmunternden Zuspruch und ihr kritisches Korrekturlesen.
Nicht vergessen möchte ich an dieser Stelle meine Familie, insbesondere meine Eltern, die aus irgendeinem Grund immer an dieses Projekt geglaubt haben, meine Freunde, die trotz dieses Projektes immer noch meine Freunde sind, und natürlich meinen Freund Christof, der mir mit seiner Zuversicht mehr Hilfe und Ansporn war, als er glaubt, und mir mit seinem Humor so manchen Tag gerettet hat. Vielen Dank.