

i

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA
COMPUTAÇÃO**

Herculano Haymussi De Biasi

**DESENVOLVIMENTO DE UMA METODOLOGIA
DE VISÃO COMPUTACIONAL PARA O AUXÍLIO
NO PLANEJAMENTO CIRÚRGICO NO IMPLANTE
DE PRÓTESES ENDOLUMINAIS**

Dissertação submetida à Universidade Federal de Santa Catarina como parte dos requisitos para a obtenção do grau de Mestre em Ciência da Computação

Professor Dr.rer.nat Aldo von Wangenheim

Florianópolis, fevereiro de 2001

**DESENVOLVIMENTO DE UMA METODOLOGIA DE
VISÃO COMPUTACIONAL PARA O AUXÍLIO
NO PLANEJAMENTO CIRÚRGICO NO IMPLANTE
DE PRÓTESES ENDOLUMINAIS**

Herculano Haymussi De Biasi

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação Área de Concentração Sistemas de Computação e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Prof.Dr. Fernando A. O. Gauthier
- Coordenador -

Banca Examinadora

Prof. Dr.rer.nat. Aldo von Wangenheim
(orientador)

Prof. Dr. Pierre Galvagni Silveira
(co-orientador)

Prof. Dr. Sérgio Peters

Prof.Dr. José Eduardo De Lucca

*"Du musst herrschen und gewinnen,
Oder dienen und verlieren,
Leiden oder triumphieren
Amboss oder Hammer sein"*

*"Você deve ser o mestre e vencer
ou servir e perder
lamentar ou triunfar
ser a bigorna ou o martelo"*

J.W.von Goethe, *Der Gross-Cophta* (1791) Act ii

AGRADECIMENTO

Aos colegas e amigos, virtuais ou não,

*aos colegas do Projeto Cyclops, que de uma maneira ou outra
contribuíram para a realização deste, pelo apoio técnico,
mas principalmente, pela amizade*

*à Universidade do Contestado - UnC, pelo apoio
aos meus ex-alunos, pelo que me ensinaram e pelo incentivo dado*

*Aos professores e servidores da Universidade Federal de Santa Catarina,
em especial à Verinha e à Valdete, da secretaria do CPGCC*

*ao meu orientador, prof.Aldo von Wangenheim,
pelo incentivo e por possibilitar que este
trabalho fosse realizado no contexto do Projeto Cyclops*

e ao meu co-orientador prof.Pierre Galvagni Silveira

A CAPES/ACAFE/PIQDT pelo auxílio financeiro

À Deus

*À minha família,
principalmente minha irmã, Fernanda,
pelo sorriso que veio sempre na hora certa.*

Sumário

LISTA DE FIGURAS	IX
LISTA DE QUADROS	XIII
RESUMO	XIV
ABSTRACT	XV
1 - INTRODUÇÃO	16
1.1 - O Projeto Cyclops	16
1.2 - Motivação	16
1.3 - Definição do problema	17
1.4 - Objetivos	18
1.4.1 - Objetivo geral	18
1.4.2 - Objetivos específicos	18
1.5 - Análise de requisitos	19
1.6 - Estrutura do trabalho	20
2 - FUNDAMENTAÇÃO TEÓRICA	22
2.1 - Descrição do problema	22
2.2 - Aneurisma de aorta abdominal	22
2.2.1 - Variantes	24
2.2.2 - Incidência e complicações	24
2.2.3 - Diagnóstico	25
2.2.4 - Tratamento	27
2.2.4.1 - Tratamento Convencional	27
2.2.4.2 - Tratamento endoluminal / cirurgia endovascular minimamente invasiva	28
2.2.4.3 - Endopróteses	32
2.3 - Aquisição das imagens	36
2.3.1 - Tomografia computadorizada (TC)	36
2.3.2 - Ressonância magnética (RM)	38
2.4 - Armazenamento das imagens	39
2.4.1 - Padrão DICOM	39
2.4.2 - Conteúdo de uma imagem DICOM	41
2.5 - Processamento e análise de imagens	44
2.5.1 - Segmentação	44
2.5.2 - Técnicas de segmentação	45
2.5.3 - Crescimento de região (region growing)	46
2.5.3.1 - Divisor de águas (watershed)	47
2.5.3.2 - Segmentação combinada entre crescimento de regiões e watershed	50
2.5.3.3 - Mumford&Shah	50

2.5.4 - Detecção de bordas	55
2.5.4.1 - Operador Canny	56
2.5.4.2 - Filtro VDRF 60	
2.5.5 - Ferramenta de programação visual: Khoros 60	
2.6 - Redes neurais artificiais 62	
2.6.1 - Aprendizado de uma rede neural artificial	63
2.6.2 - Topologias	64
2.7 - Visualização tridimensional de imagens	66
2.7.1 - Reconstrução de imagens 2D em 3D na medicina	67
2.7.2 - Triangularização	67
2.8 - Disponibilização de reconstruções 3D	71
2.8.1 - VRML	72
2.8.2 - Modelagem de objetos em 3D	74
2.8.3 - Utilização do computador em visualização 3D	75
2.8.4 - Animação	75
2.8.5 - Raytracing e raycasting	76
2.9 - Verificação de consistência em imagens e correção de falhas de segmentação baseada em conhecimento	78
2.9.1 - Avaliação de expectativas generalizada	80
2.9.2 - Expectativas	81
2.9.3 - Resultados do processamento de imagens	83
2.9.4 - Avaliação de expectativas	84
2.9.5 - Método de rotulação consistente	85
2.9.5.1 - A árvore de busca em backtracking	86
2.9.5.2 - Backtracking com forward-checking	88
2.9.6 - Complexidade	93
2.9.6.1 - O problema de rotulação consistente inexato	93
2.9.7 - Extensões ao modelo de Haralick&Shapiro	97
2.9.7.1 - Estimativa do erro	99
2.9.7.2 - Levar em conta mais de uma unidade	100
3 - DESENVOLVIMENTO DO TRABALHO	101
3.1 - Obtenção e visualização das imagens	101
3.1.1 - Visualização	102
3.2 - Segmentação das imagens	105
3.3 - Identificação dos segmentos	109
3.4 - Detecção do tecido aneurismático	113
3.5 - Correção da segmentação através do módulo extCLP	114
3.6 - Detecção de bordas	115
3.7 - Geração do polígono	116
3.8 - Correção manual do polígono	118
3.9 - Mensuração da artéria	119
3.9.1 - Localização do centro do segmento	120
3.9.2 - Linha baricêntrica	122
3.9.3 - Geração dos planos perpendiculares	124
3.9.4 - Geração do "piloto artificial"	125

3.9.5 - Informações necessárias para o projeto da prótese	126
3.9.6 - Mensuração através do piloto	128
3.10 - Reconstrução tridimensional da artéria	130
3.10.1 - Processo de triangularização	131
3.10.2 - Visualização da reconstrução	132
3.10.3 - Geração da reconstrução	133
4 - RESULTADOS E DISCUSSÃO	135
4.1 - Estudo comparativo dos resultados das mensurações	135
4.2 - Trabalhos futuros	138
5 - CONCLUSÃO	140
6 - GLOSSÁRIO	142
7 - REFERÊNCIAS BIBLIOGRÁFICAS	143
ANEXO 1 - DIAGRAMA DE ENTIDADES E RELACIONAMENTOS DO DICOM	141
ANEXO 2 - MODELAGEM DO SISTEMA EM UML.....	142

Lista de Ilustrações

Figura 1: Segmento abdominal da aorta com aneurisma	23
Figura 2: Exemplos de aneurismas	25
Figura 3: Stents	29
Figura 4: Balão utilizado na operação de implantação da endoprótese	30
Figura 5: Aorta aneurismática com endoprótese implantada	31
Figura 6: Operação de implantação da prótese	31
Figura 7: Exemplo da modularidade da endoprótese	33
Figura 8: Endopróteses retas	33
Figura 9: Endopróteses bifurcada	33
Figura 10: Mecanismo de disparo da prótese	34
Figura 11: Introduutor do stent	34
Figura 12: Posicionamento da prótese	35
Figura 13: Flexibilidade na navegação de artérias tortuosas	35
Figura 14: Implantação da prótese	36
Figura 15: Seção transversal de tomografia computadorizada do cérebro	38
Figura 16: Corte transversal de RM do cérebro	39
Figura 17: Diagrama do modelo de comunicação	41
Figura 18: Processo de segmentação	46
Figura 19: Relevo topográfico do watershed	47
Figura 20: Barragem e mínimos locais	48
Figura 21: Construção do watershed	48
Figura 22: Sequência de imagens segmentadas pelo algoritmo de Mumford&Shah	55

Figura 23: Detecção de bordas por Canny	57
Figura 24: Imagem original e o resultado do processamento com Canny	60
Figura 25: Duas imagens a serem reconstruídas em 3D.	68
Figura 26: Representação das conexões após a primeira fase.	69
Figura 27: Imagem das conexões após a primeira fase.	69
Figura 28: Representação das conexões após a segunda fase.	70
Figura 29: Imagem das conexões após a segunda fase.	70
Figura 30: Demonstração de uma reconstrução complexa conexões e faces	71
Figura 31: Sistema de coordenadas universal	74
Figura 32: Pirâmide de visão	78
Figura 33: Exemplo de ray tracing	78
Figura 34: Corte original de uma TC da aorta e duas diferentes segmentações com Mumford&Shah mostrando problemas de sub e super-segmentação.....	79
Figura 35: Problema digraph-matching	86
Figura 36: Árvore de busca em backtracking	87
Figura 37: Árvore de busca em backtracking com checagem para frente	87
Figura 38: Segmento da árvore de busca para resolução do problema das 6 damas	89
Figura 39: Aquisição dos cortes tomográficos	101
Figura 40: DicomEditor	102
Figura 41: Aneurism Analyser Editor	103
Figura 42: Corte original de TC resultando da segmentação por limiarização, operações de morfologia matemática e classificação da região por uma rede neural	105
Figura 43: Segmentação utilizando limiarização (thresholding)	106
Figura 44: Processamento da imagem limiarizada com abertura morfológica	106
Figura 45: Exemplo de segmentação utilizando watershed	108
Figura 46: Geração dos segmentos	109

Figura 47: Identificação de um segmento pela rede neural como a luz da artéria	112
Figura 48: Corte original de uma TC da aorta e duas diferentes segmentações com Mumford&Shah mostrando problemas de sub e super-segmentação.....	113
Figura 49: Imagem original e segmentos divididos em três grupos	115
Figura 50: Os segmentos identificados como tecido aneurismático, os quais são fundidos num único segmento.	115
Figura 51: Detecção de bordas utilizando o filtro VDRF	116
Figura 52: Geração do polígono ao redor da borda	117
Figura 53: Exemplo mostrando uma segmentação defeituosa e posterior correção ..	118
Figura 54: Editor para correção do polígono	119
Figura 55: Determinação do centro do segmento	122
Figura 56: Junção dos centros em cada um dos cortes	123
Figura 57: Representação do cálculo do centro de massa	123
Figura 58: Geração da linha baricêntrica	124
Figura 59: Planos perpendiculares	124
Figura 60: Geração dos planos perpendiculares à orientação da figura	125
Figura 61: piloto sagital do volume tomográfico	126
Figura 62: piloto coronal do volume tomográfico	126
Figura 63: Dimensões do aneurisma para endopróteses torácicas	127
Figura 64: Dimensões do aneurisma para endopróteses aorto bi-iliaca	128
Figura 65: Modo de mensuração	129
Figura 66: Detalhe do módulo de visualização mostrando o diâmetro do segmento arterial	129
Figura 67: Mensuração através do piloto artificial	130
Figura 68: Definição dos itens a serem mensurados	130
Figura 69: Triangularização entre os cortes	131
Figura 70: Reconstrução com texturas e com triângulos	132

Figura 71: Reconstrução de um segmento do corpo humano	133
Figura 72: Visualização através de um browser	134
Figura 73: Mensurações 3D realizadas com o software Advantage Workstation	136
Figura 74: Mensuração 3D efetuada com o software Advantage Workstation	136
Figura 75: Mensuração 3D feita através do software StentPlanner	137
Figura 76: Mensuração 3D feita através do software StentPlanner	138
Figura 77: Reconstrução 3D feita através do software StentPlanner	139

Lista de Quadros

QUADRO 1: IODs implementados	42
QUADRO 2: Resultado da primeira fase da Triangularização	69
QUADRO 3: Resultado da segunda fase da Triangularização	70
QUADRO 4: Parâmetros usados para representar "views" de anatomias em imagens radiológicas	82
QUADRO 5: Parâmetros gerados para cada segmento de imagem resultante de uma segmentação super sensitiva que difere dos protótipos de base de dados	84
QUADRO 6: Estado da tabela de erros futuros durante a busca em árvore backtracking com forward-checking	90
QUADRO 7: Parte da tabela de erros futuros	97
QUADRO 8: Parâmetros para o watershed	107
QUADRO 9: Parâmetros de entrada para a rede neural	111

Resumo

O tratamento endoluminal tem evoluído de forma significativa, reduzindo enormemente os riscos do tratamento aneurismático. Essas técnicas de tratamento endoluminais poderiam ser ainda melhores se utilizassem um suporte computacional para uma clara e objetiva quantificação da patologia aneurismática. Em função das variações anatômicas entre pacientes existe a necessidade clara de um software específico para a mensuração das patologias vasculares tratáveis endoluminalmente a partir de volumes tomográficos no sentido de: a) prover dados para a escolha de uma prótese endovascular adequada dentre os modelos comercialmente disponíveis ou, caso não exista um modelo comercial adequado, b) prover dados quantitativos detalhados que permitam a um fabricante de endopróteses a confecção de uma prótese com diâmetro e curvatura adequados às especificidades anatômicas do paciente. Em função das deficiências da tecnologia atualmente empregada, faz-se necessário o desenvolvimento de um novo enfoque para o processamento de volumes tomográficos objetivando o planejamento cirúrgico endovascular. O trabalho aqui descrito objetivou o desenvolvimento de uma metodologia de visão computacional e de uma ferramenta de software para o auxílio ao planejamento do tratamento endoluminal. Para o desenvolvimento inicial da tecnologia escolheu-se o aneurisma de aorta abdominal como área de aplicação. Deve-se atender às seguintes exigências principais. a) identificação de forma automática com exatidão e confiabilidade a luz e o tecido aneurismático da aorta abdominal e a luz das artérias femurais e ilíacas com base em um volume tomográfico; b) execução uma reconstrução tridimensional e exportação desses dados de reconstrução e mensuração em um formato que permita a visualização através de técnicas de realidade virtual e compartilhamento através da Internet e que possam ser utilizados para a confecção das próteses endovasculares; c) realização de forma automática e confiável a mensuração de diâmetro, volume e extensão de segmentos arteriais, independentemente de sua curvatura; d) conformidade com o padrão DICOM 3.0 e independência de plataforma.

Abstract

Recently advances in endoluminal treatment have highly reduced risks in aneurysm treatment. An obstacle to a endoluminal techniques larger dissemination however, is the computational support absence to produce clean and objective aneurysmatic pathology quantification. Due to patients anatomical variations exists an evident necessity of specific software to make the automatic measurement of vascular pathologies endoluminally treatable, from tomographical volumes in order to: a) produce data to choose an adequate endovascular prosthesis among available commercially models, or, in the lack of one, b) produce qualitative detailed data that permit to an endoprosthesis manufacturer make a prosthesis with appropriate diameter and curvature to the specific patient anatomical. Due to the currently technological deficiencies used today, it makes necessary the development of a new approach to tomographical volume processing in order to do a better endovascular surgical planning. Our approach intended develop a software tool to aid the endovascular surgical planning. To the initial technology development we have chosen the abdominal aortic aneurysm as application area. This tool would be attend the following requirements: a) exact and reliable automatic identification of the abdominal aortic aneurysmatic light and tissue and the femoral and iliac arterial light with base in a tomographical volume; b) produce and export a three-dimensional reconstruction, allowing the visualization through virtual reality techniques and sharing by the Internet, and also allowing the design of the endoluminal prosthesis; c) reliable automatic diameter, volume and extension measurement of arterial segments, free of its curvature; d) platform independent, running as workstations as also PCs Windows and also radiological patterns conformity, in compliance with DICOM 3.0.

1 - INTRODUÇÃO

1.1 - O Projeto Cyclops

O projeto Cyclops é um projeto binacional de pesquisa de longo prazo iniciado pelos Professores Dr.rer.nat.Aldo von Wangenheim e Dr.Michael M.Richter na Universidade de Kaiserslautern - Alemanha, em 1992. Tem como objetivo o desenvolvimento e a transferência de novos métodos, técnicas e ferramentas no campo da Análise de Imagens Médicas através da utilização de técnicas de Inteligência Artificial e Visão Computacional.

Neste contexto, a cooperação com parceiros médicos e industriais foi iniciada em 1993. Hoje o Projeto se encontra em sua Fase II, estando focado na cooperação para o desenvolvimento de aplicações que possam ser de utilidade clínica prática, dentro de um consórcio de nível internacional.

Este consórcio pretende alcançar as metas do Projeto através da cooperação entre os parceiros do Brasil e da Alemanha, cujas competências específicas complementam-se em áreas relevantes ao mesmo. O referido consórcio internacional de pesquisa e desenvolvimento é composto por Universidades, parceiros industriais da área de “software”, parceiros médicos e empresas produtoras de equipamentos médicos radiológicos de ambos os países.

1.2 - Motivação

Com o crescente grau de complexidade de sistemas de análise e representação de imagens, o uso de conhecimentos proporcionalmente complexos torna-se cada vez mais exigido. Já na área médica, onde as imagens possuem qualidade e conteúdos bastante diferentes, essas exigências dos sistemas de análise e representação de imagens tornam-se bem mais necessários. Desta forma, é preciso desenvolver ferramentas computacionais capazes de suprir estas necessidades.

Desenvolver sistemas computacionais capazes de auxiliar no diagnóstico médico não é uma abordagem simples, estes sistemas exigem antes de tudo conhecimento médico ade-

quado. Para que se consiga desenvolver estas "soluções" é indispensável uma assessoria médica, este suporte irá fornecer os subsídios necessários para a resolução dos problemas. É neste âmbito que se encontra o projeto Cyclops, um projeto que procura reunir profissionais da informática e profissionais da área médica em busca do mesmo objetivo.

1.3 - Definição do problema

No campo médico, o tratamento endoluminal tem evoluído de forma significativa, reduzindo enormemente os riscos do tratamento aneurismático. Essas técnicas de tratamento endoluminais poderiam ser ainda melhores se utilizassem um suporte computacional para uma clara e objetiva quantificação da patologia aneurismática. Em função das variações anatômicas entre pacientes existe a necessidade clara de um software específico para a mensuração das patologias vasculares tratáveis endoluminalmente a partir de volumes tomográficos no sentido de: a) prover dados para a escolha de uma prótese endovascular adequada dentre os modelos comercialmente disponíveis ou, caso não exista um modelo comercial adequado, b) prover dados quantitativos detalhados que permitam a um fabricante de endopróteses a confecção de uma prótese com diâmetro e curvatura adequados às especificidades anatômicas do paciente.

Os pacotes de software que atualmente acompanham equipamentos radiológicos como workstations de análise de imagens são, mesmo possuindo "módulos vasculares", bastante primitivos e possuem uma série de deficiências: a) as técnicas de processamento de volumes tomográficos são extremamente simples e mesmo a identificação automática de estruturas "fáceis" como a luz de artérias de grande calibre é pouco confiável; tarefas mais complexas como a identificação de tecido aneurismático são executadas de forma ainda menos satisfatória; b) mesmo em workstations que provêm reconstrução tridimensional, não existem módulos que permitam a análise adequada desses dados, provendo informações como calibre de artérias, extensão de segmentos de artéria levando-se em consideração a sua curvatura etc; c) quando alguma forma de análise desses dados existe, os softwares não provêm nenhuma forma de se exportar esses dados para que possam ser utilizados por um sistema de projeto auxiliado por computador (CAD) para a confecção de uma prótese e d) os softwares existentes, mesmo aqueles em conformidade

com DICOM 3.0, não geram reconstruções tridimensionais que possam ser exportadas em um formato que permita que sejam compartilhadas pelo usuário com outros especialistas por exemplo através da Internet.

Em função das deficiências tecnológicas atualmente empregadas, faz-se necessário o desenvolvimento de um novo enfoque para o processamento de volumes tomográficos objetivando o planejamento cirúrgico endovascular.

1.4 - Objetivos

Os objetivos deste trabalho foram divididos em objetivo geral e objetivos específicos, a fim de facilitar a análise do problema a ser resolvido e a implementação

1.4.1 - Objetivo geral

Desenvolvimento, implementação e testes de metodologias capaz encontrar, reconstruir e mensurar a aorta e o aneurisma abdominal a partir de tomografias computadorizadas, helicoidais ou não.

1.4.2 - Objetivos específicos

- Desenvolver uma ferramenta capaz encontrar, reconstruir e mensurar a aorta abdominal a partir de tomografias computadorizadas, helicoidais ou não;
- Reconstruir em 3D a luz da aorta;
- Encontrar, mostrar e medir a linha baricêntrica da aorta. Linearizar a linha baricêntrica;
- Desenvolver uma ferramenta capaz encontrar, reconstruir e mensurar estruturas adjacentes como o trombo mural e calcificações a partir de tomografias computadorizadas, helicoidais ou não;

- Reconstruir em 3D as estruturas adjacentes à aorta abdominal;
- Estruturar uma interface que ofereça ao usuário condições de operar com segurança e confiabilidade, proporcionando produtividade nas funções do sistema;
- Realizar, a partir dos dados encontrados, um projeto de prótese endovascular específica a um determinado paciente;
- Auxiliar interativamente o usuário na confecção do laudo;

1.5 - Análise de requisitos

Aqui são descritos os requisitos a serem satisfeitos para que se atinja os objetivos propostos para este trabalho. O relacionamento entre os requisitos e os objetivos é necessário, proporcionando desta forma, a satisfação dos objetivos anteriormente propostos.

- O Sistema *Cyclops Stent Planner* deverá segmentar de forma automática com exatidão e confiabilidade a luz da aorta abdominal e a luz das artérias femurais e renais com base em um volume tomográfico. (Obj.I)
- Após isso deverá ser realizada a correta classificação do segmento relativo à luz da artéria aorta (Obj.I).
- Após isso o segmento correspondente à luz da artéria deverá ser efetuada a correta mensuração da estrutura. (Obj.I)
- O segmento referente à luz da artéria deverá ser reconstruído em três dimensões. (Obj.II)
- A linha baricêntrica da luz da artéria deverá ser encontrada, linearizada e medida. (Obj.III)
- O Sistema *Cyclops Stent Planner* deverá segmentar de forma automática com exatidão e confiabilidade o tecido aneurismático abdominal com base em um volume tomográfico. (Obj.IV)
- Após isso deverá ser realizada a correta classificação do segmento relativo ao tecido aneurismático abdominal (Obj.IV).

- O segmento referente ao tecido aneurismático deverá ser reconstruído em três dimensões. (Obj.V)
- Desenvolver uma interface que ofereça ao usuário condições de operar com segurança e confiabilidade, proporcionando produtividade nas funções do sistema. (Obj.VI)
- Desenvolver uma ferramenta que a partir dos dados encontrados, possa realizar o projeto de uma prótese endovascular específica a um determinado paciente. (Obj.VII)
- Fornecer condições de se analisar os resultados obtidos com o sistema, demonstrando graficamente em três dimensões. Será utilizado um método de visualização desenvolvido na linguagem Smalltalk [6], este método fornecerá informações em código VRML. (Obj.VI)
- Exportar os dados em um formato passível de ser utilizado por um sistema de CAD (Obj.VII)
- Fornecer condições de auxiliar interativamente o usuário na confecção do laudo. (Obj.VIII)

Requisitos Operacionais:

- O projeto de desenvolvimento do sistema Cyclops Stent Planner, está localizado dentro do Projeto Cyclops, desta forma a definição das ferramentas segue o padrão utilizado no projeto. A linguagem de programação a ser utilizada para o desenvolvimento do Sistema Cyclops Stent Planner, devendo ser assim como na grande maioria das aplicações do Cyclops, Smalltalk Visualworks 3.0 (Obj.V):
- O sistema a ser desenvolvido apresentar compatibilidade com todos os outros sistemas do projeto, como também com o próprio Sistema Cyclops, proporcionando interoperabilidade desejada.

1.6 - Estrutura do trabalho

A estrutura deste trabalho ficou distribuída em capítulos da seguinte forma:

No capítulo dois é realizada uma contextualização teórica, apresentando uma descrição do problema, também é feita uma breve descrição do contexto médico, dos processos de análise de imagens e das técnicas de reconstrução tridimensional e visualização em realidade virtual.

No terceiro capítulo, são abordados com detalhes os métodos e materiais utilizados. No capítulo quatro são discutidos os resultados do presente trabalho e também são sugeridas algumas idéias para o desenvolvimento de trabalhos futuros e aperfeiçoamento deste. No capítulo cinco é desenvolvida a conclusão. No sexto capítulo é apresentado o glossário e, no sétimo capítulo, são apresentadas as referências bibliográficas utilizadas.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo tem o objetivo de descrever a base teórica relacionada às técnicas e ferramentas utilizadas na implementação computacional da metodologia proposta. Esta fundamentação estende-se também à parte médica, a qual torna-se imprescindível para aqueles que não tenham conhecimentos da área da medicina, para uma melhor compreensão do presente trabalho.

2.1 - Descrição do problema

O problema consiste em determinar, parametrizar e coordenar os diversos métodos adequados capazes de encontrar, reconstruir e mensurar a aorta e o aneurisma abdominal a partir de tomografias computadorizadas, helicoidais ou não.

2.2 - Aneurisma de aorta abdominal

As artérias são vasos que conduzem o sangue sob pressão do coração até os capilares. No homem o sistema arterial compreende dois troncos ramificados: a artéria pulmonar, que parte do ventrículo direito do coração e transporta sangue carregado de gás carbônico até os pulmões (pequena circulação); e a aorta, que nasce do ventrículo esquerdo e transporta sangue oxigenado para todos os órgãos do organismo (grande circulação). As paredes das artérias são constituídas por três túnicas: interna (íntima), média e externa (adventícia). As artérias são elásticas e contráteis, permitindo assim a manutenção de uma pressão arterial estável.

Se houver alguma alteração na estrutura da parede da aorta também haverá alteração no fluxo sanguíneo. São duas as principais doenças que causam depósito de material na parede da aorta, provocando o acúmulo de cálcio na corrente sanguínea: o arteriosclerose e o ateroma.

A arteriosclerose é uma doença involutiva do sistema arterial, causada principalmente pelo desenvolvimento de ateromas, com necrose e calcificação da túnica média da artéria e inflamação esclerosada da túnica interna. Provoca o acúmulo de cálcio na corrente sanguínea.

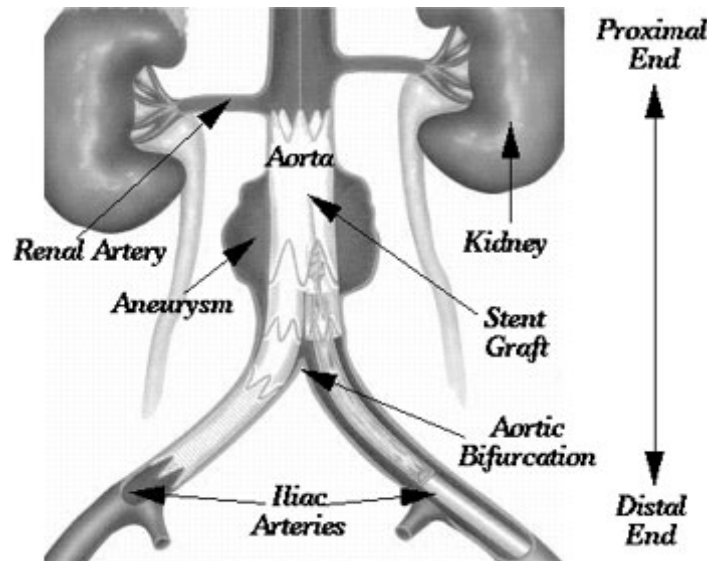


FIGURA 1: Segmento abdominal da aorta com aneurisma

O ateroma é a degeneração lipídica da túnica média das artérias, provocando a formação local de depósitos de colesterol e de sais de cálcio. A gordura vai aderindo ao cálcio formando uma camada na parede e pode ocasionar dois tipos de problema: o enfarto ou o aneurisma.

O enfarto ocorre quando a camada vai ocupando o interior da artéria, entupindo a mesma.

A palavra Aneurisma vem do grego e significa dilatação. Trata-se de um tumor causado pela dilatação de um segmento de uma artéria ou de um vaso sanguíneo. Devido ao forte fluxo da corrente, a camada expande a parede da aorta, causando a dilatação progressiva da mesma.

O diâmetro normal da aorta depende de fatores que incluem a idade, sexo e a pressão sanguínea. Em 1991, o *Subcomitê on Reporting Standard for Arterial Aneurysms* (SRSAA) definiu aneurisma como uma dilatação focal e permanente igual ou superior a 50% do diâmetro normal, sendo ectasia uma dilatação inferior a 50% e arteriomegalia uma dilatação arterial difusa. O aneurisma apresenta-se principalmente na forma fusiforme (forma de fuso) ou na forma sacular.

2.2.1 - Variantes

Os tipos de aneurisma mais comuns são: o abdominal, o torácico ou cardíaco e o arteriovenoso.

Aneurisma abdominal: o aneurisma abdominal é o tipo mais comum, ocorrendo geralmente na região infra-renal

Aneurisma torácico: é também chamado de aneurisma cardíaco e é uma complicação relativamente freqüente do infarto do miocárdio.

Aneurisma arteriovenoso: trata-se de uma comunicação direta entre uma artéria e uma veia, de origem traumática ou congênita. O aneurisma arteriovenoso cerebral é também chamado de angioma cerebral.

2.2.2 - Incidência e complicações

A doença é relativamente comum, ocorrendo em cerca de 5% dos homens a partir dos 60 anos, sendo que a mortalidade por ruptura pode variar entre 50 a 90%. Aproximadamente 15.000 mortes estão relacionadas ao aneurisma de aorta, constituindo-se na terceira causa de morte súbita em pacientes entre 65 e 80 anos. A incidência triplicou nas últimas três décadas. Os homens são mais afetados do que as mulheres com uma taxa de quatro homens para cada mulher.

O aneurisma da aorta apresenta evolução natural para a ruptura e morte. A única maneira de reduzir a taxa de mortalidade é a sua identificação e tratamento antes da ruptura, pois os pacientes submetidos ao reparo eletivo apresentam uma sobrevida a longo prazo semelhante a população normal. A técnica da correção cirúrgica iniciou pela ligadura do aneurisma em vários sítios anatômicos envolvendo principalmente artérias periféricas, evoluindo para a substituição do segmento aneurismático por um enxerto sintético e finalmente a terapêutica endovascular.

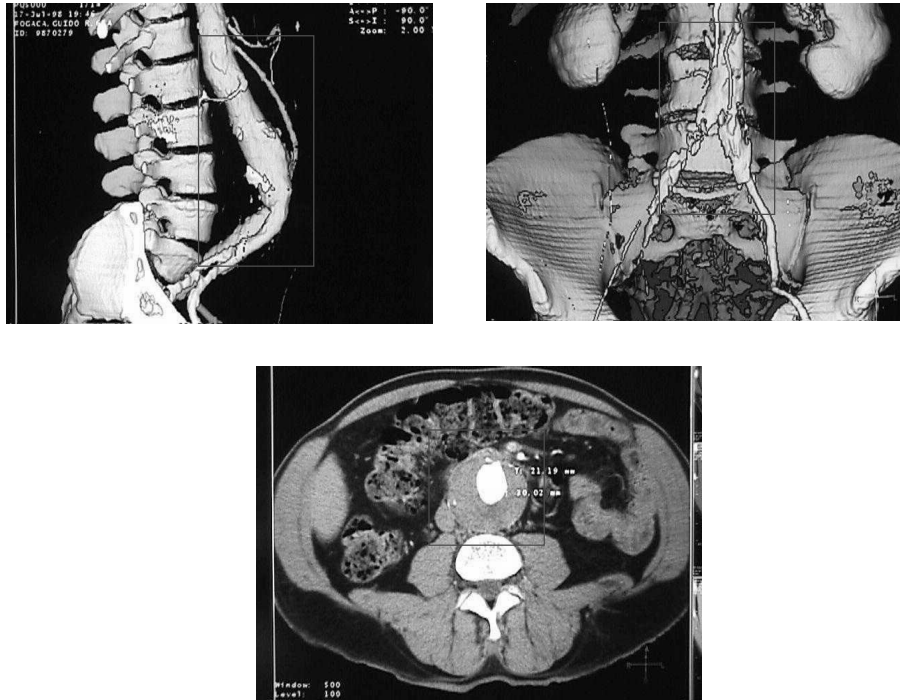


FIGURA 2: Exemplos de aneurismas

2.2.3 - Diagnóstico

Entre 70 a 75% dos aneurismas de aorta são assintomáticos, sendo diagnosticados durante investigações clínicas ou radiológicas. Também são diagnosticados durante explorações cirúrgicas abdominais provocadas por outras doenças. O aneurisma pode causar sintomas devidos à expansão, ruptura, compressão de estruturas adjacentes, embolização, dissecação ou trombose, causando dores súbitas e severas na região dorsal.

A radiografia simples pode revelar o contorno da aorta se houver calcificação. A ultra-sonografia abdominal é o método mais barato para diagnóstico e acompanhamento dos aneurismas, com sensibilidade de aproximadamente 100%. Vantagens: não utiliza meios de contraste iodado, nem radiação ionizante. Desvantagens: ineficiente em pacientes muito obesos ou com excesso de gases intestinais.

Outra desvantagem é no planejamento cirúrgico, pois não documenta adequadamente a extensão proximal e distal do aneurisma e não avalia o acometimento de ramos arteriais, como as artérias mesentéricas¹ e renais.

A tomografia computadorizada é altamente sensível e específica para o diagnóstico, avaliando melhor o tamanho do aneurisma. Necessita de contraste iodado e radiação ionizante, mas fornece melhores informações sobre a forma do aneurisma e suas relações com os vasos viscerais e renais. É mais cara que a ultra-sonografia e apresenta utilidade limitada na avaliação pré-operatória, pois não avalia a presença de doença oclusiva² nos ramos arteriais, sendo que, neste caso, pode-se utilizar a tomografia helicoidal.

A ressonância magnética fornece medidas acuradas e imagens vasculares com grande definição, podendo substituir a aortografia, com a vantagem de oferecer imagens em cortes coronais, sagitais e transversos, além de não utilizar contraste iodado e radiação ionizante. As desvantagens são o custo e dificuldades na avaliação de doenças oclusivas renais e mesentéricas, além de ser contra-indicada em pacientes com marcapasso e cliques metálicos.

A aortografia digital auxilia muito o planejamento cirúrgico pela sua ótima definição anatômica e melhor avaliação de doença oclusiva em ramos arteriais sendo de especial valor em casos de aneurismas extensos acima das artérias renais, na associação de hipertensão renovascular, suspeita de doença isquêmica³ mesentérica e associação com doença oclusiva ileofemoral. A determinação do tamanho do aneurisma fica prejudicada pela possibilidade de trombos murais não radiopacos e o efeito de paralaxe⁴, o que subestimaria o tamanho do mesmo. Há ainda controvérsia sobre a utilização rotineira do

-
1. Mesentérico: relativo ou pertencente ao mesentério, que é a estrutura anatômica, semelhante a leque, que prende o jejuno e o íleo à parede abdominal posterior, e se compõe de duas camadas, tendo extensão suficiente para dar a essas duas porções do intestino delgado uma considerável mobilidade
 2. Oclusiva: que produz obstrução de canal, de orifício ou de interior de órgão, parcial ou total, devida a causas diversas e promovida intencionalmente ou não.
 3. Isquêmica: que provoca suspensão ou baixa, localizada, de irrigação sangüínea, devida à má perfusão circulatória arterial.
 4. Paralaxe: expressão náutica, que significa a correção que é necessário introduzir na altura de um astro em relação ao horizonte aparente para ter a sua altura em relação ao horizonte artificial

método, sendo mais indicada para casos selecionados, pois tem caráter invasivo, utiliza contraste iodado, radiação ionizante e possui custo elevado.

A arteriografia é uma exame radiológico dos troncos arteriais e de seus ramos, feitos depois da injeção de um produto capaz de se misturar ao sangue e manter-se opaco aos raios.

2.2.4 - Tratamento

A recomendação da *Society for Vascular Surgery* e da *International Society for Cardiovascular Surgery* é que os aneurismas com diâmetro maior que 4 cm ou com diâmetro duas vezes superior ao da aorta infra-renal normal devem ser corrigidos, a menos que haja contra-indicação clínica. Se a correção eletiva for contra-indicada, deve-se fazer reavaliação com ultra-sonografia a cada seis meses para aneurismas com diâmetro entre 4 cm e 5 cm e a cada três meses nos maiores. Se a taxa de crescimento for maior ou igual a 0,5 cm, deve-se considerar o tratamento cirúrgico. Também deve-se levar em conta a expectativa de vida do paciente e o risco do procedimento cirúrgico.

2.2.4.1 - Tratamento Convencional

Um dos tratamentos é a correção cirúrgica, substituindo o segmento dilatado por uma prótese sintética. A operação, feita sob anestesia geral associada ou não à bloqueio epidural⁵, é realizada nos aneurismas abdominais por uma abordagem transpiretoneal⁶ através de uma laparotomia⁷ mediana xifopúbica ou por via retroperitoneal.

5. Epidural: posição superior a dura-máter, a mais externa e espessa camada que envolve o encéfalo e a medula espinhal.

6. Transperitoneal: através do peritônio, membrana serosa que reveste o interior da cavidade abdominal.

7. Laparotomia: processo destinado a participar por vezes do tratamento de graves peritonites e que consiste em, após intervenção cirúrgica abdominal, deixar aberta, intencionalmente, a parte abdominal, pelos seus aspectos anterior e laterais.

No aneurismas torácicos é utilizada a toracotomia⁸. É necessário o deslocamento de boa parte dos órgãos do aparelho digestivo para se conseguir a exposição cirúrgica do aneurisma. Após isso a aorta é pinçada acima e abaixo do aneurisma. A seguir realiza-se a abertura do saco aneurismático e esvaziamento de seu conteúdo. Em ato contínuo a prótese é implantada através de uma anastomose⁹ término-terminal com sutura junto ao colo proximal e distal do aneurisma. Quando a doença se manifesta também nas artérias ilíacas e ou femorais, utiliza-se um enxerto bifurcado, com as anastomoses distais ao nível das ilíacas ou mais frequentemente junto à bifurcação femoral bilateral. É necessária uma monitoração invasiva e rigorosa das constantes vitais do paciente no trans e pós-operatório imediato. Após o processo cirúrgico, o paciente necessitará de dois dias de internação em UTI (Unidade de Tratamento Intensivo) e mais algumas semanas de observação até obter alta. O período pós-operatório é delicado, sendo o infarto agudo do miocárdio a principal causa de morte tanto no trans quanto no pós-operatório.

Mesmo com a modernização das técnicas anestésicas e cirúrgicas a mortalidade operatória ainda é alta, entre 4% a 6% para procedimentos eletivos.

2.2.4.2 - Tratamento endoluminal¹⁰ / cirurgia endovascular minimamente invasiva

A outra técnica é o tratamento endoluminal, que consiste na utilização de dispositivos intravasculares (*stents* - endoprótese metálica expansível). O conceito de prótese endovascular (*stent-graft*) foi inicialmente discutido por Dotter em 1969 quando descreveu a implantação de um *stent* em uma artéria poplítea¹¹ canina. Seu experimento não descrevia especificamente a combinação do *stent* com um material protético vascular (Dacron ou PTFE-politetrafluoretileno expandido ultra-fino). A primeira exclusão de aneurisma aórtico intraluminal em um modelo animal foi realizado por Balko et al. em 1987. Desde então vários investigadores tem reproduzido experiências bem sucedidas

8. Toracotomia: incisão cirúrgica no tórax.

9. Anastomose: comunicação, material ou artificial, entre dois vasos sanguíneos ou outras formações tubulares.

10. Endoluminal: interno ao lume, espaço entre as paredes de um vaso.

11. Poplítea: relativo à região posterior do joelho.

com uma variedade de enxertos endovasculares e *stents* para exclusão de aneurisma em animais. Em 1990, Juan Carlos Parodi, cirurgião vascular argentino, descreveu em um modelo experimental com cães a aplicação com sucesso de enxertos intraluminais compostos de Dacron e *stent* de Palmaz. Logo após os primeiros estudos experimentais Parodi e seus colaboradores iniciaram as implantações de dispositivos semelhantes em pacientes portadores de aneurismas com alto risco para cirurgia convencional.

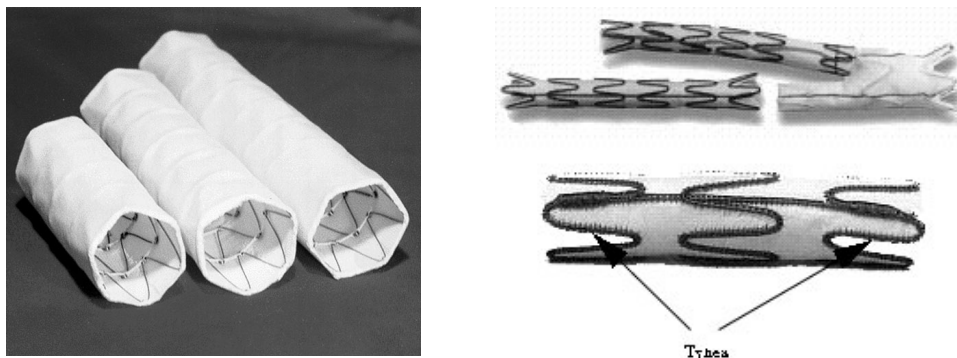


FIGURA 3: Stents

O material utilizado é protético vascular (Dacron ou PTFE). Baseia-se nos mesmos princípios da operação clássica. A diferença está no fato de que ao invés de se implantar a prótese através de uma laparotomia, o *stent* é introduzido na região interna do aneurisma através de um catéter a partir de um acesso remoto por uma das artérias femorais ou ilíacas orientados por um aparelho de angiorradiologia. Após isso o *stent* é inflado por um balão (Fig.4) até se alcançar a posição ideal. Existem também outras formas para a liberação que não através de balonamento. O *stent* é fixado devido à sua força radial, sem necessidade de sutura. Não é necessária anestesia geral e após poucos dias em observação, o paciente recebe alta. Por tratar-se de um procedimento que dispensa a laparotomia e a necessidade de anestesia geral, a vantagem da correção endoluminal não está apenas no menor risco, mas na diminuição do desconforto do paciente e do tempo de internação. O íleo paralítico¹² quando existente é breve, permitindo que o paciente se alimente no dia seguinte ao da intervenção. A deambulação também é mais precoce, e o tempo de observação na UTI é de poucas horas. A função respiratória é muito pouco afetada no pós operatório, prevenindo complicações pulmonares.

¹².Íleo: terceira e última porção do intestino delgado. Descreve em seu trajeto, inúmeras circunvoluções.



FIGURA 4: Balão utilizado na operação de implantação da endoprótese

Com a evolução das próteses criaram-se protocolos para sua implantação inicialmente aplicados em pacientes com contra-indicação maiores a cirurgia convencional. Ao mesmo tempo as indicações foram ampliadas tornando-se as mesmas dos pacientes candidatos a cirurgia convencional, apesar de que estudos clínicos com acompanhamento a longo prazo não haviam sido concluídos.

Atualmente várias publicações já existem relatando bons resultados com esta técnica, com casuísticas que se situam ao redor de 300 casos. A *International Society for Endovascular Surgery* dedicou a esse assunto sua reunião anual em 1996 (Phoenix-Arizona), concluindo que a experiência mundial somava pouco mais de 300 casos até aquela data, com seguimento de até 5 anos em algumas séries - como a de Parodi - e mortalidade de cerca de 10%. Cabe ressaltar que na experiência inicial da maioria dos centros os pacientes selecionados para realização desta nova técnica eram todos de altíssimo risco, geralmente nos quais a cirurgia convencional havia sido contra-indicada.

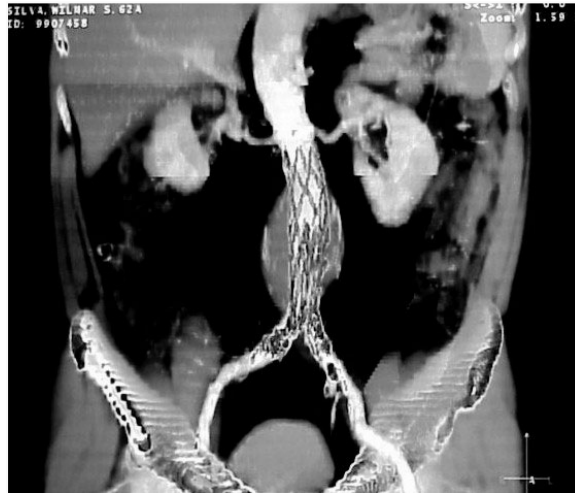


FIGURA 5: Aorta aneurismática com endoprótese implantada

Pelo fato de que os estudos clínicos com acompanhamento cirúrgico a longo prazo não terem sido concluídos, o tratamento endoluminal é aplicado a pacientes com contra-indicações maiores à cirurgia convencional. Durante o *51º Annual Meeting “The Society for Vascular Surgery”*, em 1997, James May et al. Apresentaram um estudo comparativo de 327 pacientes submetidos à cirurgia convencional e ao tratamento endovascular. Concluíram que o tratamento minimamente invasivo apresenta a mesma mortalidade do tratamento convencional apesar da taxa de 44% de mortalidade nos pacientes de alto risco com contra-indicação à cirurgia convencional.

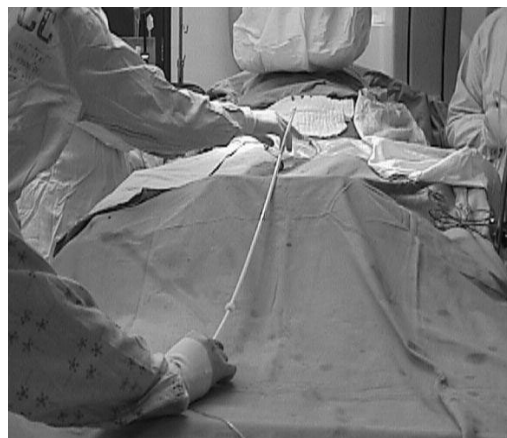


FIGURA 6: Operação de implantação da prótese

2.2.4.3 - Endopróteses

Atualmente existem vários dispositivos patenteados nos Estados Unidos, todos ainda em protocolos de estudo monitorados pelo *FDA (Food and Drug Administration)*. Dentre eles cabe salientar o *EVT* da *Endovascular Technologies Inc.* (composto de aço inoxidável e Dacron ultra-fino), *Vanguard* da *Boston Scientific Corporation* (níquel-titânio (NiTi) e Dacron Meadox), *Talent* da *World Medical Manufacturing Corporation* (NiTi e Polyester) e o *AneuRx* da *Meditronic Inc.* composto de NiTi e Polyester.

No Brasil atualmente são fabricados dois tipos de dispositivos. Uma prótese reta auto-expansível composta de "Z" *stents* de aço inoxidável e Dacron, comercializado pela *Brayle Biomédica*, e outra, desenvolvida por um consórcio entre a Universidade Federal de Santa Catarina e a Nano Endoluminal que são disponíveis nas formas reta e bifurcada modular (Fig.9). Recentemente a Nano Endoluminal se tornou a primeira empresa nacional a ter a produção de suas próteses homologada pelo Ministério da Saúde. Os dispositivos são elaborados sob medida, com diâmetro e comprimento que podem variar conforme as necessidades e características anatômicas de cada paciente. Existem vários modelos diferentes no diâmetro, comprimento da prótese e catéter introdutor, sendo escolhido o modelo que mais se aproxima das características anatômicas do paciente. A estrutura dos *stents* é feita de níquel-titânio flexível e é auto-expansível. O revestimento é de politetrafluoretileno expandido ultra-fino (e-PTFE). A extremidade contra-lateral é montada em catéter com calibre que permite sua introdução por punção. A prótese também conta com marcas radiopacas que orientam o posicionamento. O *stent* proximal é descoberto, permitindo a implantação sobre as artérias renais. As endopróteses são modulares, permitindo a realização de uma ampla gama de soluções minimamente invasivas para o tratamento de aneurismas e traumas de aorta abdominal e torácica, conferindo segurança e precisão na união de seus módulos.

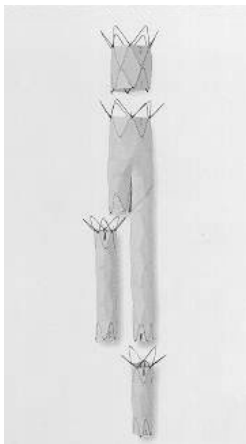


FIGURA 7: Exemplo da modularidade da endoprótese

As endopróteses retas (Fig.8), são projetadas para o tratamento de aneurismas de aorta abdominal, aneurismas e dissecções de aorta torácica e tratamento de traumas agudos (pseudo-aneurismas e fístulas artério venosas) de aorta e artérias ilíacas.



FIGURA 8: Endopróteses retas

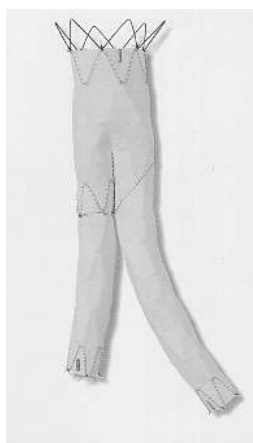


FIGURA 9: Endopróteses bifurcada

As próteses são fáceis de manusear e inserir. O sistema de disparo com tração axial e estrutura de níquel-titânio auto-expansível de suas endopróteses conferem ao sistema extrema facilidade no tratamento da patologia, dispensando balonamento, hipotensão arterial ou qualquer outro procedimento extra para a liberação da endoprótese.



FIGURA 10: Mecanismo de disparo da prótese

O sistema introdutor da endoprótese é constituída por um catéter com disparador, como mostrado na Fig.11. Em Fig.11a é mostrada a endoprótese montada dentro do sistema com trava de segurança posicionada. A Fig.11b mostra o curso de disparo livre para o início da liberação da endoprótese. A Fig.11c exibe a posição do catéter introdutor após a liberação total da endoprótese. Percebe-se também o atuador de disparo do sistema de tração axial da endoprótese.

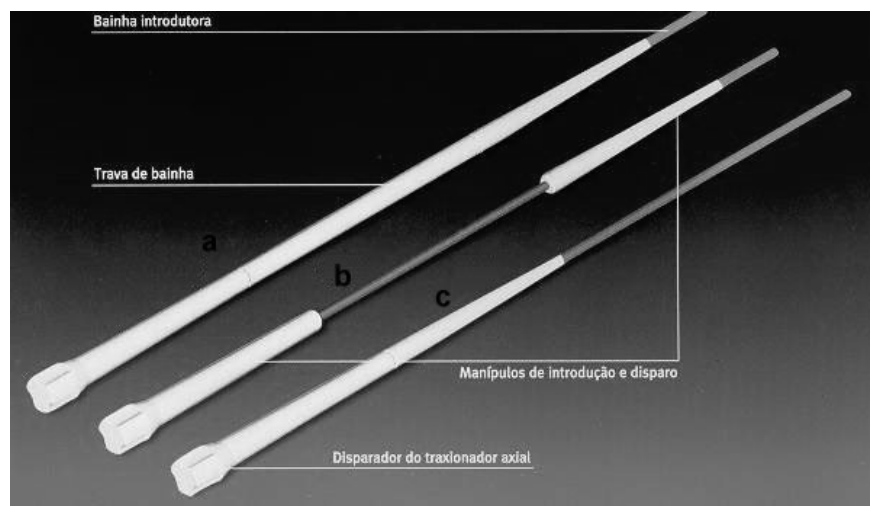


FIGURA 11: Introdutor do stent

O sistema de posicionamento e liberação é preciso. A tecnologia da tração axial aliada ao sistema de disparo em dois tempos proporciona maior precisão no posicionamento e fixação, evitando seu encurtamento ou deslocamento pela ação da pressão arterial (Fig.12). O reduzido perfil disponibiliza maior flexibilidade e facilidade de navegação em artérias tortuosas, proporcionando um procedimento cirúrgico menos agressivo e traumático, conforme mostrado na Fig.13.

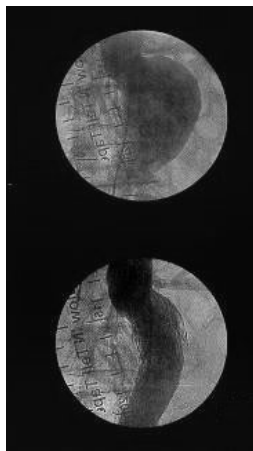


FIGURA 12: Posicionamento da prótese

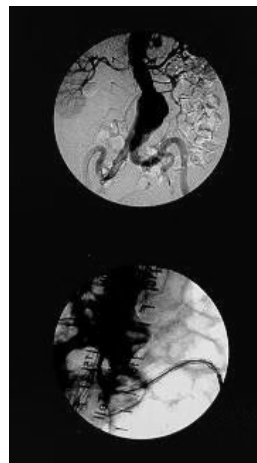


FIGURA 13: Flexibilidade na navegação de artérias tortuosas

A seguir é mostrado a seqüência para a introdução da prótese. Em Fig.14a é mostrada a progressão do sistema introdutor primário. Em Fig.14b ocorre o início da liberação da endoprótese. Na Fig.14c o corpo principal da endoprótese se liberta do catéter introdutor, o detalhe mostra a liberação do atuador de disparo do sistema de tração axial. Na Fig.14d ocorre o implante da extremidade contra-lateral.

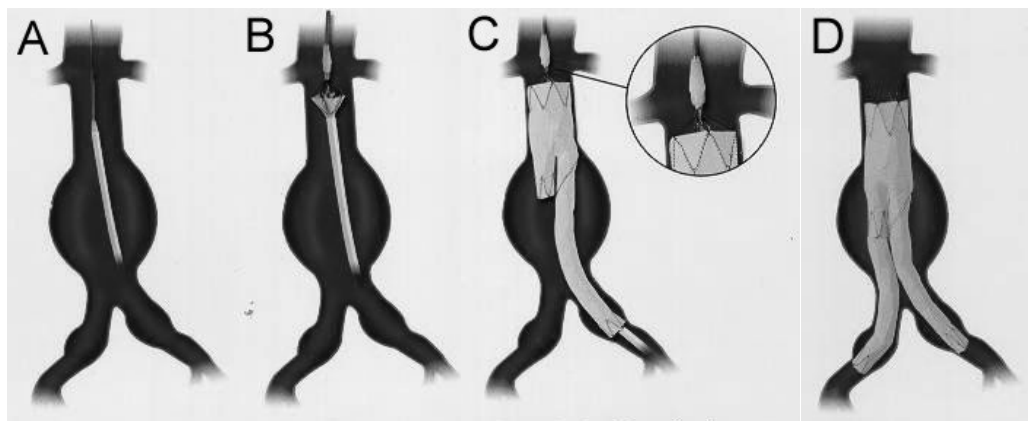


FIGURA 14: Implantação da prótese

Em 1998, cerca de 90.000 pacientes foram tratados de aneurisma de aorta. Aproximadamente 11.000 (12%) foram tratados por técnica endovascular. As previsões são de que estes números cresçam para 19.800 (18%) no ano 2000 e 41.400 (30%) em 2002. Vários estudos demonstram uma significativa redução da morbidade e da permanência hospitalar na utilização do tratamento endoluminal.

2.3 - Aquisição das imagens

2.3.1 - Tomografia computadorizada (TC)

O inventor dessa técnica foi o engenheiro inglês Geoffrey Housefield, em 1970. Por causa desse invento ele ganhou o prêmio Nobel de Medicina. A radiografia convencional é um método que utiliza o computador para aumentar a discriminação da densidade possível com raios-X. Permite distinguir apenas quatro densidades naturais do corpo humano. Isto se deve à obrigatória superposição de estruturas tridimensionais num filme bidimensional, à interação dos raios-X pouco colimados com o volume relativamente grande de tecido e à resposta fotográfica limitada do filme.

A tomografia computadorizada foi um grande avanço em relação aos raios-X, pois pode apresentar até 255 tons de cinza contra 20 ou 30 do raio-X. Outra vantagem é que a tomografia pode apresentar cortes em sentido transversal (axial) e eliminar a superposi-

ção de estruturas adjacentes, comuns em chapas de raios-X. Na tomografia computadorizada utiliza-se feixe de raios X altamente colimado e a leitura da radiação é feita por detectores especiais muito sensíveis. A radiação é emitida em torno do corpo com movimento continuado do tubo de raios X e o sinal luminoso periodicamente gerado nos detectores é transformado em impulsos elétricos, quantificados por computador. A imagem é reconstruída matematicamente, para integrar milhares de leituras de atenuação, e é numérica, mas pode ser apresentada analogicamente em um monitor. Esta imagem é na realidade tridimensional, constituída por elementos de volume de uma matriz com espessura típica entre 1.5 e 10 mm. Os elementos de superfície da matriz (pixels) definem a resolução espacial, que varia entre 0.3 e 1.5 mm² (milímetros quadrados). A discriminação espacial da tomografia computadorizada é inferior à da radiografia convencional.

No diagnóstico por TC toma-se a densidade da água como valor 1; as estruturas mais densas que a água, por exemplo a gordura, tem densidades negativas, sendo a do ar a mais negativa, representada por -500. O osso aparece em branco e o ar em preto, as demais estruturas intermediárias têm tons de cinza mais claro ou escuro, dependendo da densidade correspondente.

O uso do contraste iodado por via venosa é de grande importância em TC. Ele se acumula em alguns locais de maneira variável, mas apenas onde há quebra de barreira hematoencefalica. Nestes pontos, as neoplasias cerebrais por exemplo, a densidade do tecido patológico é aumentada e diz-se que houve realce anormal do contraste. Limitações que devem ser mencionadas incluem aneurismas, só visíveis de maneira confiável quando relativamente volumosos (maiores que 1 cm), e lesões isodensas ao liquor, quando nele inclusas, sem efeito de massa associado, como cistos parasitários ventriculares e cisternais.

A tomografia computadorizada contribui pouco na investigação de doenças comuns que não tem manifestação radiológica, como as meningites agudas não-complicadas, convulsões generalizadas na infância e doenças psiquiátricas não-orgânicas. Para uma vasta gama de enfermidades, entretanto (anomalias congênitas, doenças vasculares, infecciosas, neoplasias, doenças hereditárias, traumas), a tomografia computadorizada representa o passo mais importante e freqüentemente final no diagnóstico e planejamento terapêutico. A Fig.15 mostra uma seção transversal de tomografia computadorizada do cérebro:

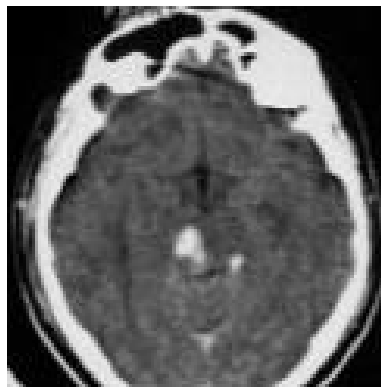


FIGURA 15: Seção transversal de tomografia computadorizada do cérebro

2.3.2 - Ressonância magnética (RM)

A ressonância magnética é um dos mais significativos avanços do século no que diz respeito a diagnósticos médicos por imagem. Permite que se tenha imagens com alta definição de qualquer parte do corpo. Sob efeito de um potente campo magnético, prótons do corpo humano são sensibilizados de maneira uniforme, principalmente os presentes nos átomos de Hidrogênio (a água perfaz 69% do volume corporal). Em seguida um campo magnético oscilatório (rádio frequência) é emitido, obedecendo o ritmo desses prótons (em ressonância com esses) que, uma vez cessado, devolve a energia absorvida nesse processo, permitindo a formação da imagem através da decodificação de sinais por computadores.

Além de não irradiar o paciente, a ressonância magnética na medicina contemporânea tornou-se um dos métodos mais estudados nos grandes centros médicos mundiais. Na Fig.16 temos um exemplo de seção transversal de ressonância magnética do cérebro:



FIGURA 16: Corte transversal de RM do cérebro

2.4 - Armazenamento das imagens

2.4.1 - Padrão DICOM

As imagens podem ser arquivadas como *bitmaps*, com compressão, e o arquivo pode ou não conter um cabeçalho que descreve a tabela de cores e os dados da imagem. O cabeçalho deve conter as informações sobre como os dados estão armazenados, os dados do paciente e demografia, e informação técnica sobre o exame e as imagens, como os detalhes geométricos, que permitam a integração com outras imagens.

Em se tratando de imagens médicas, um dos maiores problemas é seu acesso, já que a maioria dos aparelhos armazenam os dados em formatos proprietários, o que dificulta sua integração com imagens de outros aparelhos, ou sua reconstrução tridimensional. Percebendo esta situação, o *American College of Radiologists* (ACR) e o *National Electrical Manufacturers Association* (NEMA) reconheceram a necessidade de definir padrões que permitissem a conexão de aparelhos de diferentes fabricantes. Com isso surgiu o padrão DICOM (*Digital Imaging and Communications in Medicine* - Imagens e Comunicações Digitais na Medicina). Trata-se de uma especificação detalhada que descreve um método de modelagem e troca de imagens e informações associadas a ela. Ele trabalha com os padrões de comunicação e armazenamento da indústria destinados a

armazenagem de imagens digitais vindos dos mais diversos tipos de diagnósticos, tais como: tomografia computadorizada, ressonância magnética, ultra-som, raio-X, medicina nuclear, filmes digitalizados e captura de vídeo.

Os estudos sobre o DICOM iniciaram em 1984, organizados por um grupo formado por membros do meio médico pertencentes aos órgãos anteriormente citados e por membros da indústria pertencentes ao NEMA. Atualmente, até entidades fora da área médica o estão adotando, como o *Committee European of Normalization* e a *Japanese Industry Association for Radium Apparatus*. A primeira versão foi publicada em 1985 e uma edição revisada em 1988.

O DICOM surgiu principalmente para suprir a falta de um produto que facilitasse a troca de informações entre aparelhos de diversos fabricantes e que permitisse um armazenamento confiável e não dispendioso das informações. Assim, permite que os fabricantes de produtos trabalhem no desenvolvimento de melhores equipamentos sem se preocupar com a forma na qual eles irão armazenar as informações. Ele possui como foco a habilidade de enviar e receber informações padronizadas, facilitando assim a transmissão e permitindo a criação de bancos de dados de imagens.

A transmissão de dados via rede foi um dos desafios encontrados na elaboração do padrão. O padrão deveria poder se comunicar com os diversos tipos de rede existentes. Assim, foram desenvolvidos três espécies de comunicação: uma que define a comunicação ponto-a-ponto, outra que define a comunicação com o protocolo TCP/IP e uma terceira que define a comunicação com o padrão OSI. O diagrama do modelo de comunicação pode ser visto na figura seguinte:

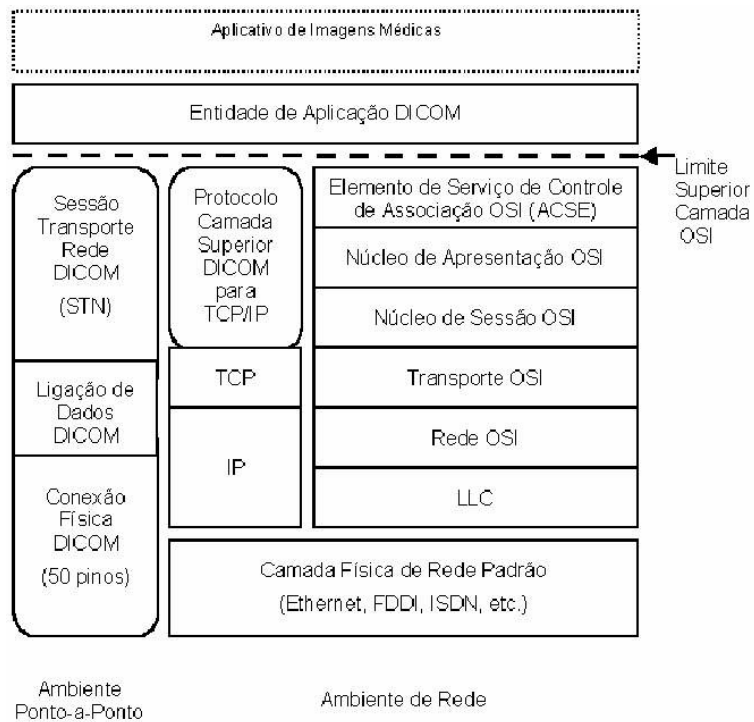


FIGURA 17: Diagrama do modelo de comunicação

O modelo de informação é baseado no diagrama de entidades e relacionamentos mostrado no Anexo I. Este processo de modelagem iniciou em um dos grupos de trabalho criados quando o DICOM estava sendo desenvolvido. Este grupo definiu os requisitos para a comunicação entre um sistema de arquivamento e comunicação de imagem e um sistema de informatização de um hospital. Esta definição permitia que fossem desenvolvidos sistemas de informatização independentes das informações do sistema de arquivamento.

2.4.2 - Conteúdo de uma imagem DICOM

O padrão DICOM descreve os objetos do mundo real como entidades neste modelo entidade-relacionamento. As entidades do modelo são objetos do mundo real como um paciente, um estudo (exame), uma imagem, as quais são descritas formalmente como IODs (*Information Objects Definition* - Definições de Objetos de Informação). A utilização do termo IOD é uma nomenclatura específica do DICOM e será utilizada neste documento a partir daqui.

Há no padrão DICOM dois tipos de IODs: IODs normalizados e IODs compostos. IODs normalizados representam objetos atômicos do mundo real e modelam apenas uma IE (*Information Entity* - Entidade de Informação). IEs são todos os objetos do mundo real do modelo que podem ser descritos através de um IOD normalizado.

Objetos do mundo real que são complexos são descritos por IODs compostos. IODs compostos modelam objetos que possuem mais de uma IE, como por exemplo, uma imagem de ressonância magnética ou uma imagem de tomografia computadorizada. IODs compostos foram modelados e implementados no *Dicom Editor* como Classes de Objetos.

Conjuntos de características de objetos compostos do mundo real que são interrelacionadas são descritas pelo Padrão DICOM como IOMs (*Information Object Modules* - Módulos de Objetos de Informação). IOMs são tipos especiais de IODs normalizados que sempre representam conjuntos dados do tipo numérico ou texto e que estão relacionados a um IOD complexo. Na modelagem do padrão DICOM 3.0 realizada na implementação do *Dicom Editor*, foram implementadas as seguintes classes de objetos na linguagem de programação Smalltalk, as quais modelam e implementam as IODs com as quais a Quadro 1 as relaciona. Essa modelagem e implantação foram feitas em trabalhos anteriores do projeto Cyclops.

Os nomes dos objetos não foram traduzidos para o português para evitar distorções na compreensão dos termos.

QUADRO 1: IODs implementados

Nome	IOD do Padrão DICOM Implementado	Tipo do IOD
CyclopsDBObject	Classe de Objeto Abstrata. Representa qualquer objeto DICOM acessado de uma base de dados DICOM. Super-classe de todos as classes aqui descrita.	-
CyclopsPatient	Patient IE	Normalizado
CyclopsPatientModule	Patient IOM	Normalizado
CyclopsPatientStudyModule	Patient Study IOM	Normalizado
CyclopsStudy	Study IE	Normalizado
CyclopsGeneralStudyModule	General Study IOM	Normalizado

QUADRO 1: IODs implementados

Nome	IOD do Padrão DICOM Implementado	Tipo do IOD
CyclopsSeries	Series IE	Normalizado
CyclopsGeneralSeriesModule	General Series IOM	Normalizado
CyclopsContrastBolusModule	Contrast Bolus IOM	Normalizado
CyclopsOverlayPlaneModule	Overlay Plane IOM	Normalizado
CyclopsSOPCommonModule	SOP Common IOM	Normalizado
CyclopsDicomImage	Image IE Utilizada no <i>Dicom Editor</i> como Classe de Objeto Abstrata. Representa qualquer objeto Imagem DICOM e implementa todas as características comuns a imagens no <i>Dicom Editor</i> . Superclasse de todos os tipos de imagens.	-
CyclopsDicomCTImage	CT Image IOD	Composto
CyclopsDicomMRImage	MR Image IOD	Composto
CyclopsDicomSCImage	SC Image IOD	Composto
CyclopsDicomUSImage	US Image IOD	Composto
CyclopsDicomXRAYImage	XRAY Image IOD	Composto
CyclopsGeneralImageModule	General Image IOM	Normalizado
CyclopsImagePixelModule	Image Pixel IOM	Normalizado
CyclopsImagePlaneModule	Image Plane IOM	Normalizado
CyclopsCTImageModule	CT Image IOM	Normalizado
CyclopsMRImageModule	MR Image IOM	Normalizado
CyclopsSCEquipmentModule	SC Equipment IOM	Normalizado
CyclopsUSImageModule	US Image IOM	Normalizado
CyclopsXRAYImageModule	XRAY Image IOM	Normalizado

O DICOM é o mais ambicioso projeto de padronização de imagens médicas empreendido pelas indústrias e pelas sociedades profissionais. É um padrão complexo, mas que pode ser implementado, e é muito útil. O padrão oferece o balanço correto entre o objetivo pragmático de ajudar na rápida implementação em produtos e a fundação

sólida e modular que assegura uma habilidade para evoluir e responder às necessidades futuras.

2.5 - Processamento e análise de imagens

Para que a luz da artéria e o tecido aneurismático possam ser detectados e reconhecidos é necessário um método para distinguí-los dentre as demais estruturas do corpo humano. Neste trabalho a artéria é localizada primeiramente para depois se localizar o tecido aneurismático.

A segmentação de imagens consiste na divisão de uma imagem específica em regiões (segmentos), regiões estas que devem corresponder as várias estruturas que compõem a imagem em questão. Dentre os algoritmos testados, destaca-se o algoritmo de Mumford&Shah utilizado neste trabalho por fornecer os resultados considerados mais confiáveis no que se refere ao isolamento (segmentação) de diversas estruturas adjacentes à aorta. Já o algoritmo Watershed é utilizado para isolar a luz arterial, segmentação essa necessária na reconstrução em 3D dos mesmos.

A segmentação de imagens é uma das tecnologias mais importantes dentro do escopo deste trabalho. Isso se explica pelo fato de que quando consegue-se isolar a região interessada, no caso a luz arterial, do restante da imagem, permite-nos fazer a classificação de forma segura. Se não se consegue isolar a região interessada pode-se considerar impossível uma análise correta da luz arterial. Isso salienta a importância de se conseguir uma boa segmentação. Esta tecnologia será melhor especificada a seguir.

2.5.1 - Segmentação

Na área de detecção e reconhecimento de imagens não basta simplesmente representar uma imagem com diferentes cores ou graduações de cinza. Também é necessário identificar regiões e estabelecer subdivisões na imagem em sua unidade básica (pixel) para que possa ser interpretada de acordo com uma finalidade específica. A identificação de regiões ou segmentos (não sobrepostos) na imagem é chamada de segmentação. A

segmentação de imagens faz a separação de uma imagem inteira em um conjunto de regiões que não são sobrepostas. Estas regiões separadas da imagem podem ser usadas por outras técnicas de interpretação de imagens para sua melhor interpretação e reconhecimento. Sendo assim, a segmentação decompõe a imagem em partes que são significativas com respeito a uma aplicação qualquer em particular. Utilizada na identificação de imagens médicas para separar fragmentos de imagens de um determinado corte tomográfico, a segmentação procura distinguir as áreas de imagem umas das outras, assim como do fundo da própria imagem. Esta distinção, permite ao sistema de computador interpretar pixels contínuos e agrupá-los em regiões.

Os métodos de segmentação supõem que os objetos possuem superfícies homogêneas que correspondem a regiões de constante ou de suaves variações de intensidade na imagem e que a intensidade muda abruptamente nas extremidades. Isto é quase sempre válido. Encontrar objetos com propriedades que atendam completamente a essas características é uma tarefa difícil pois regiões estritamente uniformes e homogêneas são tipicamente cheias de pequenos buracos e possuem bordas ásperas.

Insistir que regiões adjacentes possuem grandes diferenças de intensidade pode causar agrupamento de regiões e as bordas serão perdidas. Por esta razão, não existe nenhuma teoria de segmentação de imagens e o estado-da-arte neste campo consiste num conjunto de técnicas *ad hoc*. O que também contribui para este estado é a dificuldade de avaliação da qualidade da segmentação. Usualmente, esta avaliação é qualitativa e baseada na visualização das imagens segmentadas.

2.5.2 - Técnicas de segmentação

A segmentação pode ser feita de diversas formas. Mas existem duas abordagens principais: a segmentação baseada em detecção de bordas e a segmentação baseada em regiões. Na primeira, para formar as bordas, busca-se por discontinuidades na imagem, que não precisam ser, exclusivamente, da mesma intensidade, mas devem aparecer em indicadores estatísticos calculados pelos valores de intensidade. Já na última, conjuntos de pixels conectados que compartilham uma propriedade em comum (tal como a mesma intensidade ou cor) são encontrados e serão usados para formar as bordas.

Os dois métodos são complementares e um apresentará resultados melhores que o do outro, tudo dependerá da aplicação onde serão utilizados. Os dois métodos podem ser combinados para melhorar o resultado. A essas combinações dá-se o nome de técnicas híbridas. Na Fig.18 é exemplificado um processo de segmentação, que inicia com a imagem original até obter os segmentos desejados.

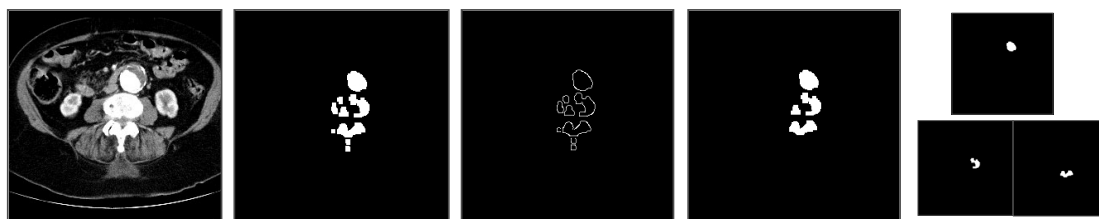


FIGURA 18: Processo de segmentação

2.5.3 - Crescimento de região (region growing)

Esta técnica procura obter regiões distintas de uma imagem baseada no procedimento descrito a seguir.

A técnica mais simples é a baseada na exploração linha por linha da imagem. Os segmentos da primeira linha são determinados pela aplicação de detecção do ponto inicial e, para cada linha subsequente, para cada segmento já existente, são adicionados aqueles pixels adjacentes, baseado em suas propriedades em comum. Da mesma forma, um novo segmento é iniciado com cada pixel que não é anexado ao segmento anterior. Os segmentos detectados podem fundir-se, dividir-se ou desaparecer na linha seguinte.

Os métodos de crescimento de região tipicamente resultam numa segmentação altamente fragmentada. Para obter regiões maiores, o processo de agregação deve ser guiado "semanticamente", ou seja, pelo uso de propriedades da imagem tomadas a priori, tal como a relação de adjacência entre regiões de propriedades variadas.

2.5.3.1 - Divisor de águas (*watershed*)

Watershed é uma poderosa ferramenta para a segmentação. É freqüentemente utilizada em conjunto com métodos de crescimento de regiões, mas pode ser aplicada isoladamente obtendo-se bons resultados.

Esta metodologia possui um certo número de variâncias. O método de segmentação watershed calcula o gradiente para todos os pixels da imagem. Imagine que os valores de gradiente formem uma superfície topográfica com vales e montanhas. As regiões mais baixas seriam correspondentes as de menor gradiente e as mais altas as de maior gradiente. O crescimento de regiões seria equivalente a uma inundação, realizada na mesma velocidade em cada mínimo regional, iniciando do mais baixo e então em cada um tão logo que o nível global da água alcance sua altitude. Os segmentos seriam formados por regiões que partindo de um mínimo local, formassem uma bacia hidrográfica. Daí o nome divisor de águas. Onde a água de diferentes mínimos irá misturar-se, constroem-se represas para separar as inundações. As represas que emergem à superfície da água constituem as linhas divisórias do *watershed*, que são formadas por contornos fechados que envolvem cada mínimo regional e correspondem às linhas das cristas do relevo, realizando uma boa segmentação por uma única linha. O agrupamento dos pixels é feito por mecanismos de busca de valores próximos a partir de cada mínimo local. Esta é a descrição do algoritmo desenvolvido por Vincent e Soille que propôs uma aproximação baseada em imersão para calcular as *watersheds*.

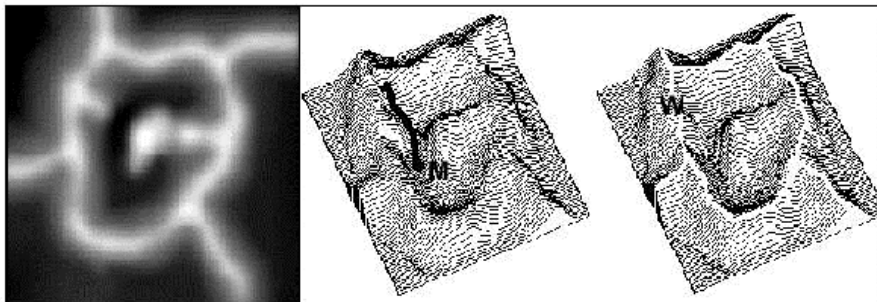


FIGURA 19: Relevo topográfico do watershed

Considere uma imagem f com uma superfície topográfica e defina os mínimos locais (represas) de f e as linhas do Watershed pelo processo de inundação. Imagine que

furando cada região mínima $Mi(f)$ desta superfície topográfica S , e que mergulhando esta superfície em uma lagoa com uma velocidade vertical constante, a entrada da água através de buracos inundaria a superfície S . Durante a inundação, duas ou mais enchentes atingiriam o nível dos mínimos locais fundindo-a em uma só. Para evitar este evento é necessário construir uma barragem sobre os pontos da superfície S , onde a inundação unisse. No fim do processo, somente as barragens iriam aparecer. Estas barragens definem o watershed da função f . Elas separam as várias represas $CBi(f)$, cada uma contendo somente um mínimo $Mi(f)$.

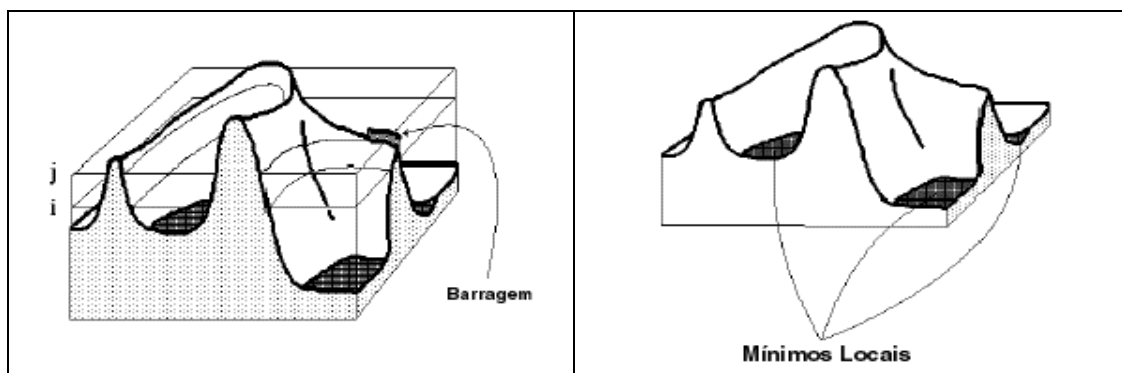


FIGURA 20: Barragem e mínimos locais

A definição da transformada de *watershed* pelo processo de inundação pode ser reorganizada pelo uso das seções da função f .

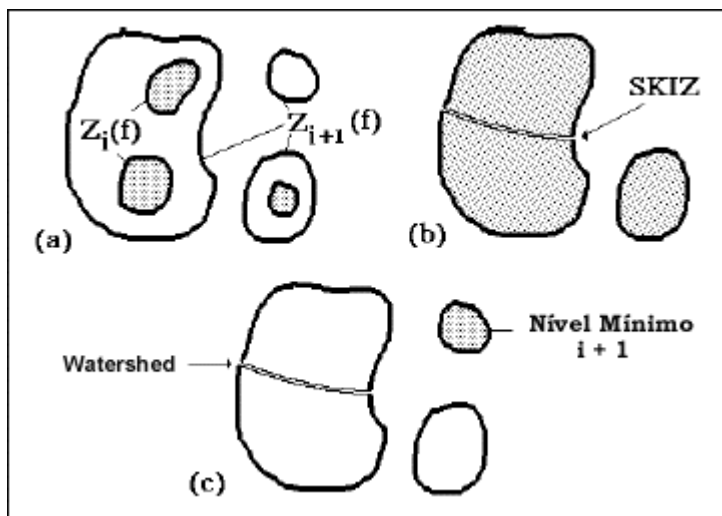


FIGURA 21: Construção do watershed

Considerando a figura anterior, uma seção $Z_i(f)$ de f no nível i , e supondo que as inundações alcance esta altura. Considere agora a seção $Z_{i+1}(f)$. Observe que a inundaçãõ de $Z_{i+1}(f)$ é realizada nas zonas de influência dos componentes conectados de $Z_i(f)$ em $Z_{i+1}(f)$. Alguns componentes conectados de $Z_{i+1}(f)$ dos quais não estão atingidos pela inundaçãõ, são por definição nível mínimo em $i + 1$. Estes mínimos devem então ser adicionados a área inundada.

Denotando por $W_i(f)$ a seção no nível i das represas de f , e por $M_{i+1}(f)$ o mínimo desta função na altura $i + 1$, temos:

$$W_{i+1}(f) = [Z_{i+1}(f) (X_i(f))] \cup M_{i+1}(f) \quad (1)$$

Os mínimos no nível $i + 1$ são dados por :

$$M_{i+1}(f) = Z_{i+1}(f) / R_{Z_{i+1}(f)}(X_i(f)) \quad (2)$$

Este algoritmo iterativo é iniciado com $W_{-1}(f) = \emptyset$. No fim deste processo, a linha do Watershed $DL(f)$ é igual a:

$$DL(f) = W_{cn}(f) \text{ (com } \max(f) = N) \quad (3)$$

Uma das melhores e mais bem testadas implementações do region-growing com *watershed* é denominada *viterRegGrow* e foi implementada por Gangolf Mittelhäusser no Departamento de Informática da Universidade de Kaiserslautern. Esta implementação foi utilizada extensivamente para as mais variadas finalidades, tanto pela própria Universidade de Kaiserslautern, como pelo Instituto Max Planck de Psicologia Cognitiva em Leipzig, e também pelo projeto Cyclops.

Atualmente existe uma versão deste algoritmo disponível no pacote *BRIAN (BRain Image ANalysis)*, que vem sendo desenvolvido pelo Instituto Max Planck. O *viterRegGrow* possui a característica de realizar segmentação de imagens 3D, caso se forneça um arquivo representando um volume de dados, como uma tomografia computadorizada, por exemplo. Para que *viterRegGrow* opere em 3D, é necessário que as imagens sejam fornecidas como série no formato VISTA.

Um aspecto importante desta implementação é o de que ela não calcula os gradientes necessários para a determinação dos divisores de águas. Este cálculo deve ser efetuado em separado, de preferência através da utilização do algoritmo de Canny.

2.5.3.2 - Segmentação combinada entre crescimento de regiões e watershed

Um segmento é um conjunto de pixels formado por uma região homogênea na imagem, cercado por regiões não homogêneas (bordas) ou por outros segmentos com diferentes níveis médios de cinza. A homogeneidade se refere também aos valores dos pixels contidos dentro do segmento em questão.

Se um segmento não possuir bordas e apresentar-se fragmentado, é por que deve haver um ou mais segmentos vizinhos com gradientes de transição muito suaves entre si. Estas transições suaves são dificilmente detectadas por algoritmos de segmentação baseados em gradientes de valores como o *watershed*. Os métodos que também utilizam gradiente apresentam alguns problemas, como o agrupamento de regiões que apresentam o mesmo gradiente mas com homogeneidade diferente. Além disto, estes algoritmos prevêm também, que a borda entre dois segmentos deve se localizar em um ponto que possui um gradiente mínimo entre eles.

A diferença entre o método combinado e o watershed, é que no primeiro os gradientes são classificados em ordem crescente, antes da inundação. Desta forma, obtêm-se um mecanismo de segmentação muito mais rápido do que o watershed, Além disto, ele determina em toda a imagem as regiões com a mesma homogeneidade, através de um algoritmo de fusão de segmentos.

2.5.3.3 - Mumford&Shah

Dos métodos de segmentação por crescimento de região, este é o mais preciso conhecido. Este método utiliza a idéia de que as regiões são grupos de pixels com uma “borracha” ao redor. As regiões crescem enquanto pudermos esticar a borracha. Quanto maiores as variações entre os pixels dentro de uma região, tanto menor a elasticidade da borracha. Seu maior problema é o tempo de processamento, que é bastante longo. O algoritmo de Mumford&Shah é indicado para situações onde uma segmentação confiável e estável com o *watershed* não é possível para o tipo de dados que se trabalha e onde é possível esperar pelos resultados, eventualmente através de processamento em lote.

Esse método é baseado na equação funcional da energia do algoritmo de Mumford&Shah (4).

Esta equação funcional da energia utiliza como base para seus cálculos a variância dos tons de cinza da imagem, ou seja, quanto maior a variância do tom de cinza maior mais difícil é a união de regiões e, quanto menor a variância do tom de cinza é maior é a união das regiões. Esta energia por sua vez, dirá o quanto uma região poderá agrupar com outras regiões, delimitando assim as extremidades das regiões.

A equação funcional de Mumford&Shah esta assim descrita:

$$E(u,K) = \int \int_{\Omega} \|u - g\|^2 dx dy + \int \int_{\frac{\Omega}{K}} \|\nabla u\|^2 d(1)x dy + \lambda \times l(K) \quad (4)$$

$E(u,K)$ - energia funcional de Mumford&Shah em função de m e da fronteira K

$g(x,y)$ - função intensidade de luz no ponto x,y ; é suave (valor de gradiente pequeno) dentro de cada região e descontínua no cruzamento com as bordas

Ω - domínio da imagem, geralmente um retângulo

Ω_i - decomposição da imagem em *piece-wise*

$$\Omega = \Omega_1 \cup \Omega_2 \cup \dots \cup \Omega_n \cup K \quad (5)$$

K - Fronteira entre regiões

$l(K)$ - comprimento dos arcos das bordas (fronteiras). Reduz-se à medida em que as regiões vão sendo fundidas.

λ - parâmetro de escala. A cada iteração incrementa-se λ . Isso pode ser realizado em passos lineares ou passos exponenciais.

$u(x, y)$ - função diferenciável dentro de cada região

$$\bigcup_{i=1}^n \Omega_i \quad (6)$$

Para o presente trabalho, a função $u(x,y)$ será uma função constante, esse valor constante será a média dos valores de intensidade $g(x,y)$ de cada ponto da região.

Abaixo tem-se a interpretação o significado de cada termo da equação funcional da energia de Mumford&Shah:

Primeiro termo: mede se $u(x,y)$ é uma boa aproximação de $g(x,y)$. Quanto melhor a função $u(x,y)$ se aproximar da função $g(x,y)$, menor será a contribuição desse termo para o valor da energia.

$$\iint_{\Omega} \|u - g\|^2 dx dy \quad (7)$$

Segundo termo: calcula a variação mínima de $u(x,y)$ dentro cada região sem a borda:. Como, no presente trabalho, a função $u(x,y)$ será o valor médio dos pontos de $g(x,y)$ dentro da região e portanto um valor constante, não existirá variação, assim este termo será sempre zero.

$$\iint_{\frac{\Omega}{K}} \|\nabla u\|^2 dx dy \quad (8)$$

Terceiro termo: calcula o comprimento das fronteiras multiplicado por um parâmetro de escala λ . Penaliza o comprimento das fronteiras, quanto maior o comprimento das fronteiras, maior será a contribuição desse termo para o valor da energia. Esse valor será multiplicado por um parâmetro que funciona como um peso.

$$\lambda \times l(K) \quad (9)$$

Equação de energia funcional simplificada de Mumford&Shah:

$$E(K) = \int \int_{\frac{\Omega}{K}} \|u - g\|^2 \, dx dy + \lambda \times l(K) \quad (10)$$

Assume $u(x,y)$ como função constante dentro de cada região, a qual é a média dos valores de $g(x,y)$ nesta região. A função da energia simplificada de Mumford&Shah fica em função apenas de K (fronteiras das regiões).

O critério de junção das regiões esta assim descrito:

Dadas duas regiões adjacentes O_i e O_j separadas por uma fronteira comum $\delta(O_i, O_j)$ e o valor de energia $E(u, K)$. Uma nova segmentação (m', K') será obtida, removendo-se a fronteira comum $\delta(O_i, O_j)$ dessas duas regiões adjacentes. Esta nova segmentação:

$(m', K') = (m, K) / \delta(O_i, O_j)$ é agora chamada uma subsegmentação.

Se a energia dessa nova subsegmentação for menor que a energia anterior (antes de juntarmos as duas regiões), ou seja:

$$E(m', K') < E(m, K)$$

então, é interessante a junção dessas duas regiões. Porém antes de juntarmos estas duas regiões devemos fazer o mesmo para todas as regiões vizinhas à estas duas regiões O_i e O_j e verificarmos qual é a junção que ocasionará o maior decréscimo de energia.

Caso não se encontre nenhuma região adjacente que ocasione nenhum decréscimo de energia, ou seja:

$$E(m', K') \geq E(m, K)$$

então, a segmentação (m, K) é chamada de 2-normal.

Quando não for mais possível juntar nenhuma região, deve-se aumentar o valor de λ . O incremento de λ poderá ser linear ou exponencial.

Abaixo é mostrada a implementação do modelo simplificado Mumford&Shah

O critério de juntar duas regiões O_i e O_j está na dependência do sinal de $E(K)/\delta(O_i, O_j) - E(K)$, no qual é apenas o decréscimo de energia. Assim a equação para o critério fica:

$$E\left(\frac{K}{\partial(O_i, O_j)}\right) = \frac{|O_i| \times |O_j|}{|O_i| + |O_j|} \times \|u \nabla g\|^2 \nabla \lambda \times l\partial(O_i, O_j) \quad (11)$$

onde:

$|O_i|$ - área da região O_i

$|O_j|$ - área da região O_j

u_i - valor de intensidade da região i

u_j - valor de intensidade da região j

$l\partial(O_i, O_j)$ - comprimento da fronteira entre as regiões O_i e O_j

O critério foi calculado da seguinte maneira:

$$u_i = \frac{I}{|O_i|} \int_{O_i} g(x) dx \Rightarrow \int_{O_i} g(x) dx = |O_i| \times u_i \quad (12)$$

$$u_j = \frac{I}{|O_j|} \int_{O_j} g(x) dx \Rightarrow \int_{O_j} g(x) dx = |O_j| \times u_j \quad (13)$$

$$u_{i+j} = \frac{I}{|O_i| + |O_j|} \int_{(O_i + O_j)} g(x) dx = \frac{I}{|O_i| + |O_j|} \left(\int_{O_i} g(x) dx + \int_{O_j} g(x) dx \right) \quad (14)$$

$$E\left(\frac{K}{\partial(O_i, O_j)}\right) = \frac{|O_i| \times |O_j|}{|O_i| + |O_j|} \times \|u \nabla g\|^2 \nabla \lambda \times l\partial(O_i, O_j) \quad (15)$$

Quando não for mais possível juntar nenhuma região, deve-se aumentar o valor de lambda (λ). Um exemplo de uma imagem segmentada com vários valores de lambda pode ser vista na Fig.22.

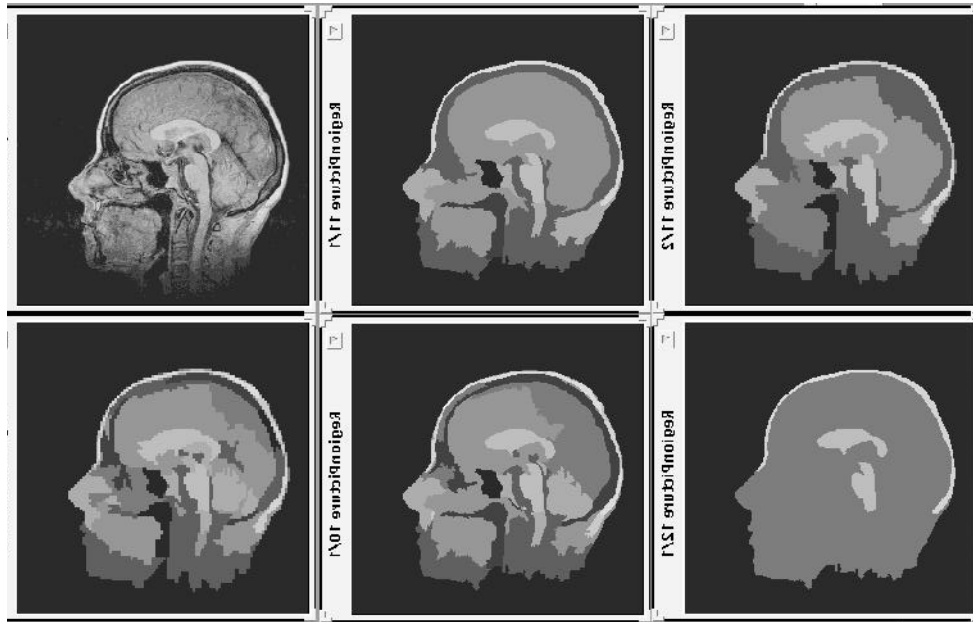


FIGURA 22: Seqüência de imagens segmentadas pelo algoritmo de Mumford&Shah com valores de lambda crescentes. Na primeira imagem, cada ponto representa uma região. Já na primeira segmentação, os pontos vizinhos foram agrupados numa mesma região segundo o critério de menor energia. Conforme é dado um valor maior para λ , mais regiões são agrupadas

Através deste algoritmo, obteve-se bons resultado no que diz respeito a segmentações de imagens radiológicas. A dificuldade encontrada na utilização deste algoritmo é o seu custo computacional que é extremamente elevado devido a grande quantidade de cálculos feitos até a obtenção dos resultados.

2.5.4 - Detecção de bordas

A detecção de bordas é uma das operações mais comumente utilizadas na análise de imagens. Uma borda é o limite entre um objeto e o segundo plano e indica o limite entre objetos justapostos. Isto significa que se numa imagem as bordas podem ser identificadas com exatidão, todos os objetos podem ser localizados e as propriedades básicas, tais como a área, o perímetro e a forma, podem ser mensurados. Desde que a visão computacional envolve a identificação e classificação dos objetos numa imagem, a detecção de bordas é uma ferramenta essencial.

A detecção de bordas é parte do processo de segmentação e, tecnicamente, pode ser descrita como sendo o processo de localização dos pixels da borda, assim, com o realce da borda, o contraste entre ela e o segundo plano será incrementado, tornando-a mais visível. Na prática, este processo procura determinar se uma borda passa sobre ou perto de um determinado pixel. Isto é feito examinando a taxa de troca de intensidade próxima ao pixel – alterações agudas são boas evidências de uma borda, alterações suaves irão sugerir o contrário.

Quando ocorre uma descontinuidade na intensidade da imagem, ou quando o gradiente tem uma variação abrupta, um operador que é sensível a estas mudanças operará como um detector de bordas. Um operador de derivada faz exatamente esta função. Uma interpretação de uma derivada seria a taxa de mudança de uma função, e a taxa de mudança dos níveis de cinza em uma imagem é maior perto das bordas e menor em áreas constantes. Se pegarmos os valores da intensidade da imagem e acharmos os pontos onde a derivada é um ponto de máximo, nós teremos marcado suas bordas. Dado que as imagens são em duas dimensões, é importante considerar mudanças nos níveis de cinza em muitas direções. Por esta razão, derivadas parciais das imagens são usadas, com as respectivas direções X e Y. Uma estimativa da direção atual da borda pode ser obtida usando as derivadas x e y como os componentes da direção ao longo dos eixos, e computar o vetor soma. O operador envolvido é o gradiente, e se a imagem é vista como uma função de duas variáveis $A(x,y)$ então o gradiente é definido como:

$$\nabla A(x, y) = \frac{\partial A}{\partial x}, \frac{\partial A}{\partial y} \quad (16)$$

Há um certo número de algoritmos de detecção de bordas e cada qual pode produzir um resultado de maior ou menor qualidade, dependendo das características da imagem na qual foi aplicado.

2.5.4.1 - Operador Canny

O detector de Canny, elaborado por Jonh Canny em 1986, é amplamente usado nos sistemas de visão computacional modernos. Ele definiu um conjunto de metas que um detector de bordas deveria ter. São elas:

- 1) Taxa de erro: o detector de bordas deveria detectar e achar somente bordas, nenhuma borda deveria faltar;
- 2) Localização: a distância entre os pixels de borda encontradas pelo detector de bordas e a borda atual deveria ser a menor possível;
- 3) Resposta: o detector de bordas não deveria identificar múltiplos pixels de borda onde somente exista um único pixel.

O detector de bordas de Canny é um filtro de convolução f que uniformiza o ruído e localizaria as bordas. O problema é identificar um filtro que otimize os três critérios do detector de bordas. Se considerarmos uma borda de uma dimensão variando no contraste e então convolucionando a borda com a função de uniformização de Gauss, o resultado será uma variação contínua do valor inicial ao final, com uma inclinação máxima no ponto onde existe um degrau. Se esta continuidade é diferenciada em relação a x , esta inclinação máxima será o máximo da nova função em relação a original. Isso pode ser observado na Fig.23.

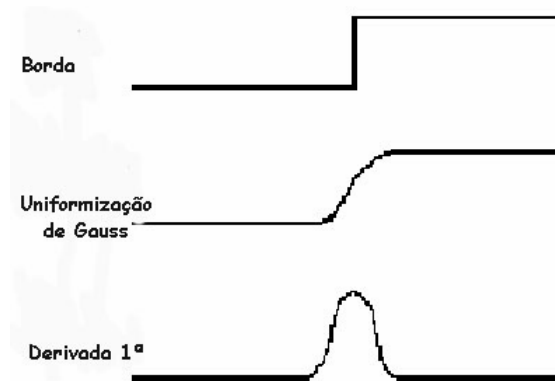


FIGURA 23: Detecção de bordas por Canny

Os máximos da convolução da máscara e da imagem indicarão bordas na imagem. Este processo pode ser realizado através do uso de uma função de Gauss de 2D na direção x e y . Os valores das máscaras de Gauss dependem da escolha do sigma na equação:

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \quad (17)$$

$$G'(y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y^2}{2\sigma^2}} \quad (18)$$

A aproximação do filtro de Canny para detecção de bordas é G' . Convolucionando a imagem com G' obtém-se uma imagem I que mostrará as bordas, mesmo na presença de ruído. A convolução é relativamente simples de ser implementada, mas é cara computacionalmente, especialmente se for em 2D. Entretanto, uma convolução de Gauss de 2D pode ser separada em duas convoluções de Gauss de 1D.

A intensidade computacional do detector de bordas de Canny é relativamente alta, e os resultados são geralmente pós-processados para maior clareza. Entretanto, o algoritmo é mais eficiente no processamento de imagens com ruídos ou com bordas difusas.

O funcionamento do algoritmo de Canny é dividido em estágios e mostrado abaixo:

Estágio um - suavização da imagem: os dados da imagem são suavizados por uma função gaussiana bidimensional, de largura especificada pelo usuário. Na prática, a convolução bidimensional com um valor alto leva muito tempo, o que obriga aproximar para duas funções gaussianas unidimensionais, uma alinhada ao eixo X e outra ao eixo Y, produzindo assim dois valores para cada pixel.

Estágio dois - diferenciação: assumindo as convoluções bidimensionais do estágio anterior, a imagem suavizada é diferenciada respeitando os eixos x e y . É possível calcular o gradiente da superfície suavizada da imagem convolucionada em qualquer direção a partir de um gradiente conhecido em qualquer uma das duas direções. Caso assumam-se a aproximação unidimensional no estágio um, então os valores suavizados do eixo x são convolucionados com a derivada de uma função gaussiana unidimensional de δ idêntico alinhado com y . Igualmente, os valores suavizados do eixo y são convolucionados com a derivada da função gaussiana unidimensional de δ idêntico alinhado com x . Do cálculo dos gradientes de x e y , a magnitude e o ângulo de declividade podem ser calculados a partir da hipotenusa e do arcotangente, de maneira similar ao operador de Sobel¹³.

Estágio três - supressão dos não-máximos: Encontradas as taxas de modificação de intensidade em cada ponto da imagem, as bordas precisam ser colocadas nos pontos de máximo, ou seja, deve-se retirar os pontos não-máximos. Um máximo local ocorre num

13. Operador Sobel: era o mais utilizado até os operadores baseados em funções gaussianas e seus derivados fossem reconhecidos extensamente. As máscaras de x e y dadas numa matriz, geralmente de 3×3 , são convolucionadas com a imagem para calcular os valores de D_x e D_y . Então a magnitude e o ângulo das bordas são calculados e armazenados com dois quadros de imagem separados.

pico na função de gradiente, ou onde a derivada da função de gradiente é 0. Entretanto, neste caso deseja-se suprimir a linha perpendicular não-máxima em relação à direção da borda, já que espera-se que a borda seja uma linha que percorre um contorno (esta suposição gera problemas para cantos).

Estágio quatro - marcação¹⁴ de bordas: o marcador usado no operador Canny usa um método chamado "histerese". A maioria dos marcadores usam um limite de marcação simples o que significa que, se os valores da borda flutuarem acima e abaixo deste valor, a linha aparecerá quebrada. Os contadores de histerese determinam um valor superior e inferior para o valor limite da borda. Considerando um segmento de linha, se um valor fica acima do limite superior, é imediatamente aceito. Se o valor fica abaixo do limite inferior é imediatamente rejeitado. Pontos que ficam entre os dois limites são aceitos se eles estão conectados a pixels que exibem respostas sólidas. A probabilidade de quebras é reduzida drasticamente, já que os pontos do segmento de linha precisam flutuar acima do limite superior e abaixo do limite inferior para a quebra ocorrer. Canny recomenda que a relação entre o limite superior e inferior seja de dois ou três para um.

O efeito do operador Canny é determinado por três parâmetros: a largura do kernel¹⁵ gaussiano usado na fase de suavização e os parâmetros de entrada superior e inferior usados na operação de realce. O incremento do kernel reduz a sensibilidade do detector para ruídos, o que ocasiona a perda de alguns detalhes mais delicados na imagem. A localização dos erros nas bordas detectadas também aumenta levemente. Normalmente, o parâmetro de entrada superior é colocado bem alto, enquanto o parâmetro inferior é bem baixo para bons resultados. Um parâmetro inferior alto pode causar quebras nas bordas devido a ruídos. Um parâmetro superior baixo aumenta o número de fragmentos de bordas falsos ou indesejados.

Um exemplo de utilização do operador Canny é visto nas imagens abaixo:

14. No contexto significa a operação sobre um ponto de imagem que produz uma imagem binária a partir de uma imagem em escala cinza. Um valor binário 1 é gerado quando o valor do pixel na imagem é está dentro do limite de valores de marcação. Caso contrário, é gerado um valor 0. O operador Canny modifica um pouco o conceito.

15. Kernel: pequena matriz de números que é usada em convoluções de imagens, kernels de diferentes tamanhos contendo diferentes padrões de números originam diferentes resultados em convolução. Também é comumente usada como sinônimo para elemento estruturado, o qual é um objeto similar usado em morfologia matemática. Um elemento estruturado difere de um kernel por possuir uma origem especificada.

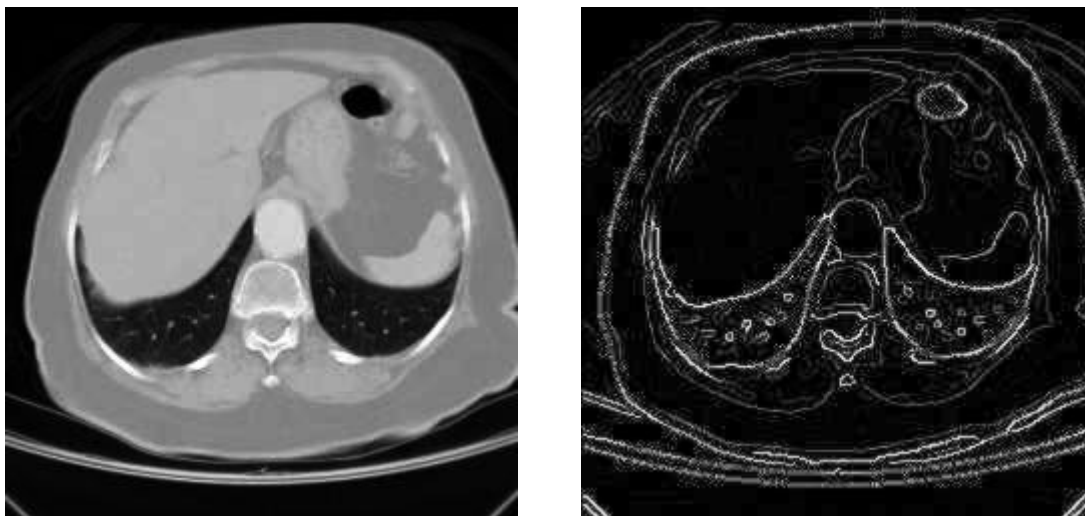


FIGURA 24: Imagem original à esquerda e o resultado do processamento com Canny à direita

2.5.4.2 - Filtro VDRF

VDRF é um filtro ótimo para a detecção de bordas. Este método procura evitar o problema que ocorre com o Canny, que é a contradição entre o efeito de suavização do filtro Gaussiano e a precisão da localização das bordas. VDRF é um filtro simétrico exponencial com tamanho de janela infinitamente largo e pode ser implementado como um simples algoritmo recursivo. As bordas detectadas pelo método DRF geram menos ruídos e possuem uma precisão e localização muito melhores.

2.5.5 - Ferramenta de programação visual: *Khoros*

Khoros foi originalmente desenvolvido por um grupo de pesquisas na *University of New Mexico* e mais tarde continuado por uma empresa denominada *Khoral Research*. *Khoros* é um pacote de software fornecido gratuitamente para fins de pesquisa e comercializado para aplicações comerciais. *Khoros* é talvez a ferramenta de análise de imagens mais citada na literatura e conta com contribuições dos mais variados grupos de pesquisa, inclusive do Brasil, onde a UNICAMP desenvolveu um dos mais completos pacotes de morfologia matemática para tons de cinza denominado MMACH.

Khoros é um sistema de software para Unix/Linux e é dividido em um conjunto de pacotes de utilitários denominados *Toolboxes*. Este conjunto pode ser estendido pelo usuário através da utilização de ferramentas de integração que acompanham o sistema.

Khoros segue dois objetivos básicos:

- a sua utilização como laboratório de testes de seqüências de análise de imagens. Para isto *Khoros* conta com uma poderosa ferramenta de programação visual denominada *Cantata*. *Cantata* é um desktop onde o usuário pode inserir ícones representando os diversos algoritmos de análise de imagens disponíveis e conectá-los uns aos outros, além de setar seus parâmetros de operação através de interfaces gráficas denominadas *panes*. Neste ambiente pode-se testar quaisquer seqüências de análise e seus parâmetros sem que se tenha de programar algo ou que se tenha que utilizar muitos programas diferentes, carregando arquivos intermediários de um para outro. Este é o melhor tipo de ambiente em análise de imagens para se testar uma possível solução para um problema. Como *Cantata* trata cada algoritmo de análise de imagens como um processo filho da interface gráfica, pode-se iniciar várias seqüências.

- a sua utilização na confecção de soluções/aplicações através da integração dos algoritmos individuais em programas existentes. Como cada algoritmo é um programa Unix *stand-alone*, pode ser usado como um processo chamado de outra aplicação desenvolvida pelo usuário, sendo os parâmetros passados via linha de comando de shell e os dados via pipes ou arquivos temporários.

Para a integração de novos aplicativos e algoritmos no sistema, o usuário precisa tê-los sob a forma de código em linguagem "C" ou outro compatível a nível de linkador e utilizar as ferramentas denominadas *Composer* e *Ghostwriter*. Com essas ferramentas o usuário inclui código para interpretação de linhas de comando e pipes específico do *Khoros* nos seus fontes e cria os *panes* necessários para o interfaceamento gráfico da aplicação via *Cantata*. Feito isto, basta recompilar e instalar a aplicação em um *Toolbox Khoros*.

Existem basicamente três versões do *Khoros*: *Khoros 1.0.6*, *Khoros 2.x* e *Khoros 2000*. Atualmente, a grande maioria dos sites que até há alguns anos distribuía *Khoros* não possui mais versões do mesmo e ele tem de ser buscado diretamente de *Khoros Research* na versão 2000. O *Khoros 1.0.6* é bem mais simples (e muito mais intuitivo) na interface e nas facilidades de conversão de formatos de arquivos de imagem, que a partir

da versão 2.0 ocorre automaticamente para a maioria dos formatos de imagem conhecidos, inclusive todos os formatos Unix, como pgm, ppm, xbm e Rasterfile. O conjunto de rotinas básicas de processamento de imagens disponível não mudou muito desde o lançamento do *Khoros 2.0*.

2.6 - Redes neurais artificiais

Para que se atingir a meta proposta fez-se necessário utilizar um método para a classificação dos segmentos candidatos à luz arterial e tecido aneurismático. Para tanto existem várias tecnologias que permitem fazer classificações, tais como conjunto de regras, redes bayesianas, redes neurais artificiais, dentre outras, que fornecem as ferramentas necessárias. Por ser um problema de certa forma bastante simples optou-se pelas redes neurais artificiais. Sua flexibilidade foi outro ponto que levou à sua escolha.

Para que se possa melhor compreender as redes neurais artificiais, é necessário saber que elas são capazes de extrair regras (informações) de uma série de dados, que nem sempre é possível compreender, o qual lhe permite fazer uma classificação baseada nos dados inseridos em seus neurônios de entrada. Para que se consiga obter resultados satisfatórios, é necessário treinar a rede neural artificial até que se chegue ao resultado esperado.

Proposta nos anos 50 pelo neurobiologista Warren McCulloch e pelo estatístico Walter Pitts, as Redes Neurais Artificiais (RNA), ou simplesmente Redes Neurais, são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através da experiência.

A estrutura da rede neural artificial é baseada no nosso sistema nervoso biológico, que é formado por um conjunto extremamente complexo de células, os neurônios. Eles têm um papel essencial na determinação do funcionamento e comportamento do corpo humano e do raciocínio. Os neurônios são formados pelos dendritos, que são um conjunto de terminais de entrada, pelo corpo central, e pelos axônios que são longos terminais de saída. Os neurônios se comunicam através de sinapses.

Sinapse é a região onde dois neurônios entram em contato e através da qual os impulsos nervosos são transmitidos entre eles. Os impulsos recebidos por um neurônio

A, em um determinado momento, são processados, e atingindo um dado limiar de ação, o neurônio *A* dispara, produzindo uma substância neurotransmissora que flui do corpo celular para o axônio, que pode estar conectado a um dendrito de um outro neurônio *B*. O neurotransmissor pode diminuir ou aumentar a polaridade da membrana pós-sináptica, inibindo ou excitando a geração dos pulsos no neurônio *B*. Este processo depende de vários fatores, como a geometria da sinapse e o tipo de neurotransmissor.

De maneira análoga, uma rede neural artificial é composta por várias unidades de processamento, cujo funcionamento é bastante simples. Essas unidades geralmente são conectadas por canais de comunicação que estão associados a determinado peso. As unidades fazem operações apenas sobre seus dados locais, que são entradas recebidas pelas suas conexões. Arquiteturas neurais normalmente são tipicamente organizadas em camadas, com unidades que podem estar conectadas às unidades da camada posterior.

Usualmente as camadas são classificadas em três grupos:

- 1) Camada de entrada: onde os padrões são apresentados à rede;
- 2) Camadas intermediárias ou oculta: onde é feita a maior parte do processamento, através das conexões ponderadas; podem ser consideradas como extratoras de características;
- 3) Camada de saída: onde o resultado final é concluído e apresentado.

Serão abordadas neste capítulo as características essenciais de uma rede neural artificial, ou seja, o processo de aprendizado e a sua topologia.

2.6.1 - Aprendizado de uma rede neural artificial

A propriedade mais importante das redes neurais é a habilidade de aprender com seu ambiente e com isso melhorar seu desempenho. Isso é feito através de um processo iterativo de ajustes aplicado a seus pesos, o treinamento. Diz-se que uma rede “aprendeu” quando ela atinge uma solução generalizada para uma classe de problemas.

O modo como ocorre o processo de treinamento é determinado pelo algoritmo de aprendizado, que nada mais é que um conjunto de regras bem definidas para a solução de um problema de aprendizado. Existem muitos tipos de algoritmos de aprendizado especí-

ficos para determinados modelos de redes neurais. Estes algoritmos diferem entre si principalmente pelo modo como os pesos são modificados.

São fatores importantes no processo de treinamento de uma rede neural artificial: a coleta de dados, a configuração da rede e a escolha do tipo de treinamento. A fase de coleta de dados é a fase onde se dá a construção da base de dados, que pode ser feita utilizando dados estatísticos ou simulando aleatoriamente situações que representem possíveis estados do sistema. Dependendo da natureza dos dados, a base de dados deve sofrer um pré-processamento. Caso contrário, pode-se não conseguir o desejado sucesso no treinamento.

Após a preparação da base de dados, a fase seguinte define a configuração da rede, onde se determina qual será a topologia (número de camadas, de neurônios por camada oculta, tipo de função de ativação e coeficientes da função de ativação) e os parâmetros do algoritmo de treinamento, critérios de parada e de avaliação. Existem metodologias para determinação de alguns parâmetros da configuração da RNA. Todavia, essa escolha é normalmente empírica, conseguida através de experiência ou comparando modelos diferentes.

O processo de treinamento é a fase onde a rede neural artificial se adapta aos estímulos, produzindo um desempenho para o qual foi projetada. Os tipos de treinamento existentes podem ser classificados como:

- 1) Aprendizado Supervisionado: quando é utilizado um agente externo que indica à rede a resposta desejada para o padrão de entrada;
- 2) Aprendizado Não Supervisionado: quando não existe um agente externo indicando a resposta desejada para os padrões de entrada;
- 3) Reforço: quando um crítico externo avalia a resposta fornecida pela rede.

2.6.2 - Topologias

Existem vários modelos de redes neurais, mas alguns receberam maior atenção dos pesquisadores tais como Bidirecionais, Madaline, Kohonen e Diretas (*feedforward*), que são diferenciados pelo modo de treinamento, interconectividade e disposição das camadas.

As redes bidirecionais são um caso particular das redes com ciclos. Esta rede possui neurônios não estáticos e uma consulta é feita ativando neurônios que representam as entradas referentes ao caso, deixando a rede evoluir até atingir o ponto de equilíbrio. A resposta é a ativação de um ou mais neurônios. Apresenta como vantagens o uso de representação de conhecimento localizada e a ausência de método de aprendizagem. As desvantagens são a possibilidade da rede não parar num ponto de equilíbrio e o tempo despendido para achá-lo pode ser longo.

As redes Madaline são um modelo multidimensional do modelo Adaline, criado por Widrow e Holf. Este modelo é semelhante ao Perceptron onde o neurônio possui uma saída proporcional à soma ponderada das excitações.

As redes de Kohonen “formam uma família de redes, com grande plausibilidade biológica que se auto-organizam através de mecanismos de competição”. Sua característica principal é apresentar a saída como sendo um grupo de neurônios, com um centro com excitação forte, a qual vai diminuindo com o distanciamento deste centro. Entradas similares possuem como saída regiões próximas dentro da rede.

As redes diretas são aquelas cujo grafo não tem ciclos. A representação mais comum é no formato de camadas, com neurônios interligados e dispostos em camadas sucessivas. As saídas dos nós da camada de entrada são ligadas pelos pesos sinápticos às entradas dos nós da camada posterior, e assim sucessivamente até a camada de saída. Por essa razão, este tipo de rede é denominada Direta (propagação para frente, *feedforward*). As camadas posteriores à camada de entrada e anteriores à camada de saída são denominadas camadas ocultas.

Uma camada pode ser definida como um conjunto de neurônios que recebe entradas de um mesmo local e tem as saídas também em um mesmo local. São as redes mais populares atualmente, principalmente pela facilidade de seus métodos de aprendizado. Elas são capazes de aproximar qualquer função não linear com precisão proporcional ao número de neurônios da rede. Porém não consegue representar todos os sistemas dinâmicos, mesmo quando utiliza neurônios dinâmicos. Quanto aos métodos de aprendizado, o algoritmo de retropropagação (*backpropagation*) é o mais utilizado, embora não seja o mais eficiente.

Durante o treinamento com o algoritmo de retropropagação, a rede opera numa seqüência de dois passos. Primeiro, um padrão é apresentado à camada de entrada da

rede. A atividade resultante flui através da rede, camada por camada, até que a resposta seja produzida pela camada de saída. No segundo passo, a saída obtida é comparada à saída desejada para esse padrão particular. Se esta não estiver correta, calcula-se o gradiente do erro com relação aos valores sinápticos da camada de saída. O erro é propagado a partir da camada de saída até a camada de entrada, e os pesos das conexões das unidades das camadas internas vão sendo modificados conforme o erro é retropropagado (origem do nome “retropropagação”).

2.7 - Visualização tridimensional de imagens

A cognição espacial humana não é expressa de maneira adequada pelas representações de imagens estáticas e bidimensionais. Para que o ser humano possa avaliar corretamente uma forma, não basta somente a representação em duas dimensões. Noções como vizinhança, distância, e interseção entre elementos são possíveis somente com a visualização em três dimensões. Para tanto, deve-se proporcionar ao espectador um ambiente onde ele possa fazer uma análise espacial da imagem e isso exige ferramentas poderosas para manipulação do dado espacial. De acordo com CAMARA (1996), "A análise espacial é a manipulação do dado geográfico de tal forma que é possível realizar computacionalmente modelos dos fenômenos do mundo real".

Para a construção de imagens, a técnica empregada em programas aplicativos de geração de objetos geométricos tridimensionais, usa principalmente o conceito de modelagem de objetos e imagens. Os sistemas empregados, em aplicativos gráficos, para a modelagem de sólidos, baseiam-se num conjunto de primitivas geométricas, por um lado bastante simples de serem implementadas e também bastante maleáveis para poderem modelar diferentes objetos. Os modelos de objetos, ficam armazenados em arquivos, possíveis de serem manipulados, interativamente por usuários. As formas geométricas implementadas são: pontos, segmentos de retas, linhas poligonais, polígonos, poliedros e formas mais complexas como arcos de curvas, superfícies curvas e superfícies quádricas. Sobre essas formas básicas podem ser aplicadas operações de translação, rotação, deformação de escala, movimentos de vértices e faces, mapeamento de superfícies com texturas, cores e iluminação.

A implementação mais complexa de modelagem é para a construção de modelos realistas. Neste caso são usados, principalmente, duas técnicas: a de modelagem aditiva e subtrativa, sendo que esta última se parece muito com a técnica de criação de esculturas. A modelagem de corpos humanos virtuais é extremamente complexa em função de possuírem como referência o próprio ser humano, que por sua vez, apresenta estrutura bastante irregular, desde o rosto até os membros que o compõem, incluindo seus detalhes como ossos, músculos, veias e tecidos.

Uma técnica alternativa para construir imagens em três dimensões é a utilização de uma linguagem que proporcione, através da descrição do mapeamento da coordenadas dos pontos e das linhas que constituem a seqüência das imagens bidimensionais e nas coordenadas apropriadas do dispositivo de visualização, a montagem de um volume tridimensional que represente este modelo. Para tanto, existe a linguagem VRML (*Virtual Reality Modelling Language* - Linguagem para Modelagem de Realidade Virtual) que, além de mostrar objetos em três dimensões, possibilita mecanismos de navegação pelo objeto, permitindo, entre outros recursos, a aproximação/distanciamento e a rotação do mesmo.

2.7.1 - Reconstrução de imagens 2D em 3D na medicina

A reconstrução de uma série de imagens radiológicas se faz importante pois auxilia a equipe médica na identificação espacial de achados anatomopatológicos. Para que se possa fazer esta reconstrução em 3D é necessário definir exatamente quais regiões deverão estar presentes na reconstrução. No sistema desenvolvido, a reconstrução em 3D tem sua importância, no que se refere a uma melhor compreensão da artéria aorta e outras estruturas relacionadas

2.7.2 - Triangularização

A triangularização é uma maneira de se criar faces para uma série de imagens, fazendo com que ela tenha uma representação em 3D. A forma triangular é a forma mais

utilizada nesta área da reconstrução em 3D, pois pode-se considerar a única forma que, quando utilizada unicamente, sua junção permite recriar praticamente todas as formas necessárias em uma reconstrução em 3D.

Para ilustrar, veremos o exemplo a seguir: para se fazer a reconstrução de duas imagens, conforme mostrado na Fig.25, num primeiro momento calcula-se um ponto intermediário entre dois pontos de uma mesma imagem em 2D (ponto intermediário entre 1 e 2 = 1,5). Baseado neste ponto, procura-se o ponto mais próximo dele na imagem subsequente levando sempre em conta suas distâncias em 3D.

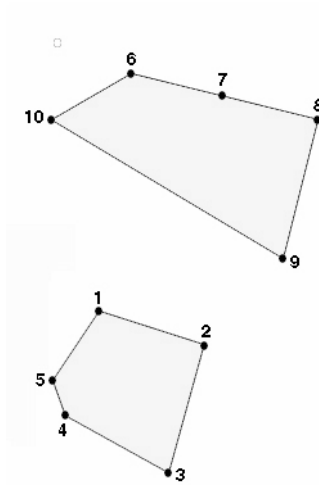


FIGURA 25: Duas imagens a serem reconstruídas em 3D

Pode-se observar a Fig.26 que faz uma representação das conexões feitas e na Fig.27 que é a reconstrução do exemplo após a primeira fase. O Quadro 2 permite uma melhor compreensão do resultado desta primeira parte da triangularização.

Pode-se também observar na linha *E* do Quadro 2 que é feita a conexão do último com o primeiro ponto (baseado na imagem em 2D).

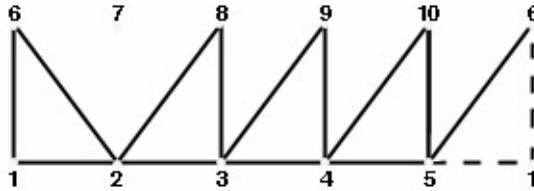


FIGURA 26: Representação das conexões após a primeira fase.

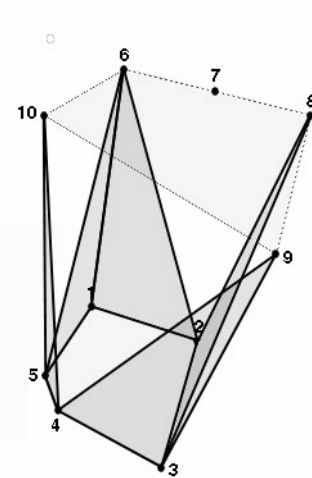


FIGURA 27: Imagem das conexões após a primeira fase

É importante notar que cada um dos valores mencionados no exemplo (1, 2, 3,...10) correspondem a uma coordenada em um espaço 3D (x, y, z).

QUADRO 2: Resultado da primeira fase da Triangularização

Triangularizações	ponto 1 (p1)	ponto 2 (p2)	ponto 3 (p3)
A	1	2	6
B	2	3	8
C	3	4	9
D	4	5	10
E	5	1	6

Nesta primeira fase faz-se a triangularização utilizando-se todos os pontos de uma das imagens em questão, a outra será feita na próxima fase. É importante salientar que o primeiro e o último ponto do polígono de uma imagem possuem a mesma coordenada, fazendo com que o polígono das imagens esteja sempre fechado.

Na segunda fase, são utilizados os pontos de *p2* resultantes da primeira fase para se completar as conexões. Utiliza-se um laço que percorra por todos os pontos da segunda imagem, fazendo a seguinte conexão:

$$p1 = n + 1$$

$$p2 = n$$

$p3 = p2$ de n no Quadro 2 (caso n não existe no Quadro 2, inserir $p2$ anterior) ou seja,

$$p1 = 7, p2 = 6 \text{ e } p3 = 2;$$

$$p1 = 8, p2 = 7 \text{ e } p3 = 2;$$

$$p1 = 9, p2 = 8 \text{ e } p3 = 3; \dots \text{etc.}$$

Desta forma têm-se todos os pontos conectados por triângulos que nos dará o resultado representado no *Quadro 3*.

QUADRO 3: Resultado da segunda fase da Triangularização

Triangularizações	ponto 1 (p1)	ponto 2 (p2)	ponto 3 (p3)
F	7	6	2
G	8	7	2
H	9	8	3
I	10	9	4
J	6	10	5

Uma representação das conexões pode ser observada na Fig.28, e a respectiva reconstrução das imagens em 3D pode ser visualizada na Fig.29.

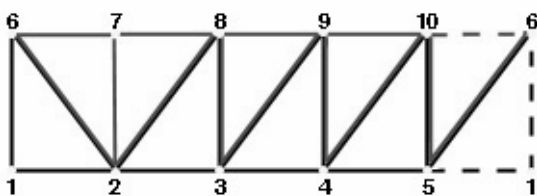


FIGURA 28: Representação das conexões após a segunda fase.

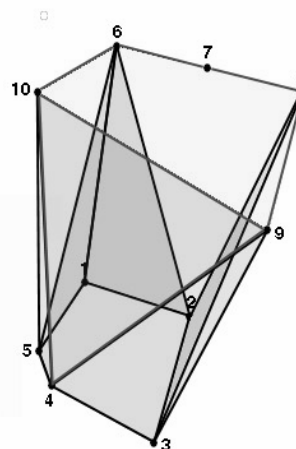


FIGURA 29: Imagem das conexões após a segunda fase

Assim, pode-se reconstruir imagens em 3D a partir de imagens 2D. Um exemplo mais complexo dessa reconstrução, utilizando a linguagem VRML para a sua visualização, pode ser observado na Fig.30.

Foram testadas várias maneiras de se fazer essa triangularização, sendo que a que trouxe melhores resultados foi previamente descrita.

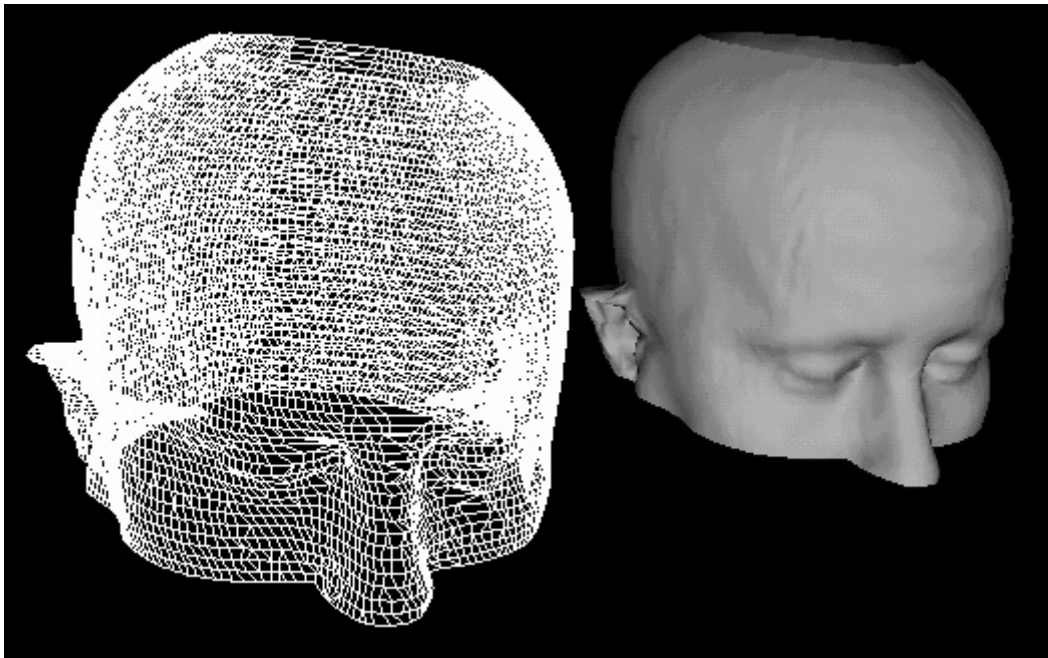


FIGURA 30: Demonstração de uma reconstrução complexa somente com as conexões (triangulações) e já com a representação das faces

2.8 - Disponibilização de reconstruções 3D

A linguagem adotada para a disponibilização de reconstruções 3D foi o VRML. Esta linguagem foi adotada devido a simplicidade na sua utilização e, para tanto, basta instalar um *plugin* junto ao *browser* de sua preferência. O VRML é uma linguagem que permite visualizar e reconstruir imagens ou mundos em 3D. É utilizada para apresentar a reconstrução em 3D feita pelo algoritmo de triangularização. A linguagem VRML é a responsável de apresentar os resultados da reconstrução em 3D feita do corpo humano, luz arterial e tecido aneurismático encontrados pelo sistema. Uma melhor descrição da linguagem VRML pode ser observada a seguir.

2.8.1 - VRML

VRML é uma linguagem independente de plataforma que permite a criação de cenários em 3D, visualização objetos por ângulos diferentes e interação com eles através de um *browser*. Seu lançamento foi em meados de 1994 quando Tim Berners-Lee, numa conferência, falou sobre a necessidade de um padrão para uma Web 3D. A linguagem foi concebida para descrever simulações interativas de múltiplos participantes em mundos virtuais disponibilizados na Internet e conectados com a WWW (*World Wide Web*). A primeira versão da linguagem não possibilitou muita interação do usuário com o mundo virtual. Nas versões posteriores foram acrescentadas características como animação, movimentos de corpos, som e interação entre múltiplos usuários em tempo real.

O código de VRML é um subconjunto de formato ASCII do *Open Inventor*, da *Silicon Graphics*, com características adicionais para navegação na *WEB*. Esta característica é semelhante às âncoras em HTML. Assim, pode-se criar âncoras em um ambiente virtual que levem a outros ambientes virtuais. A linguagem, em sua versão 1.0, trabalha com geometria 3D, que permite a criação de objetos baseados em polígonos, possuindo alguns objetos pré-definidos como triângulo, esfera, cilindro, cubo e cone, suportando transformações como rotação, translação e escala. Permite também a aplicação de texturas, luz sombreamento, etc. Outra característica importante da linguagem é o Nível de Detalhe (LOD, *Level of Detail*), que permite o ajustamento da complexidade dos objetos, dependendo da distância, em que eles se encontram do observador. Há pouca possibilidade de interação, tendo como principal objetivo a criação de mundos virtuais estáticos e criação de âncoras para outros ambientes.

A última versão desta linguagem é a 2.0, chamada *Moving Worlds VRML 2.0*. Esta linguagem tem como objetivo dar o suporte necessário para o desenvolvimento de mundos virtuais tridimensionais multi-usuário na Internet, sem precisar de redes de alta velocidade. Permite que os objetos do mundo virtual possam se movimentar e responder a eventos, baseados nas iniciativas do usuário. Permite também a utilização de objetos multimídia, como sons e filmes. As principais características desta versão são: melhoria dos mundos estáticos, interação, animação, comportamento baseado em *scripts* e prototipação de novos objetos VRML.

A programação em VRML baseia-se no paradigma da orientação a objetos, ou seja, toda a figura geométrica ou outro item inseridos num mundo são objetos que contêm atributos que dão as características destes objetos. A programação em VRML consiste basicamente num arquivo texto, que é lido por um browser que possua um *plug-in* para VRML.

Há grande diferença entre gráficos em duas dimensões e os de três dimensões. Embora seja mais difícil trabalhar com 3D, têm-se vantagens enormes em relação aos gráficos em 2D. Pode-se construir um mundo em três dimensões e povoá-lo com objetos e caminhar ao redor deles, bem como observar, com maior fidelidade, características referentes a sua textura, sons, entre outros. A única maneira de se conseguir tais peculiaridades num ambiente em duas dimensões é utilizando filmes ou animações. Porém possuem a deficiência de só poderem ser observadas pelo ângulo que o autor do filme as criou, ao contrário de um mundo 3D criado em VRML, onde pode-se, literalmente, ir para qualquer lugar do cenário.

Tudo que se precisa para criar uma cena em VRML é um editor de textos. Uma vez que os arquivos são salvos em formato ASCII com a extensão *.wrl*. A linguagem apenas descreve como os ambientes tridimensionais devem ser apresentados, pois os arquivos não são compilados.

Um desenho é composto de elementos básicos combinados: pontos, segmentos de reta e áreas (espaços preenchidos), cada um com determinadas características como cor, dimensionamento, tipo de traçado (tracejado, pontilhado, etc...). A adoção de um sistema de coordenadas faz-se importante pelo fato de possibilitar a descrição matemática do posicionamento relativo dos elementos que compõem o desenho. O sistema de coordenadas, também denominado sistema de referência, define uma referência (ou origem) em relação à qual se descrevem todos os posicionamentos. Existem os sistemas de coordenadas cartesianas e o de coordenadas polares. O sistema de coordenadas cartesianas divide-se em bidimensional (coordenadas X e Y) e o tridimensional (coordenadas X, Y e Z). Este trabalho usa o sistema de coordenadas cartesianas tridimensionais, pois objetos serão animados em três dimensões. O VRML utiliza o sistema coordenado universal-WCS (Fig.31) que é um sistema coordenado cartesiano direto, com o qual descreve-se o modelo a ser visualizado.

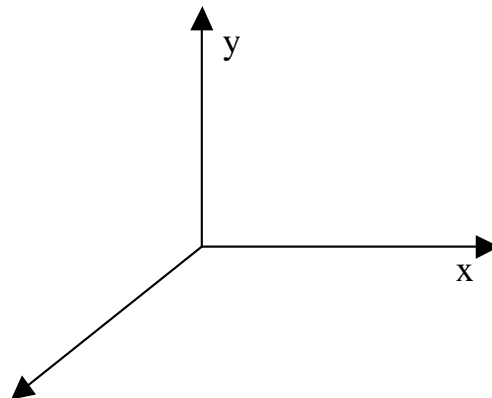


FIGURA 31: Sistema de coordenadas universal

2.8.2 - Modelagem de objetos em 3D

A criação de modelos computacionais dos objetos e cenas requer a escolha de uma forma de representação em termos de elementos primitivos e estruturas de dados. Em função do processo de modelagem e/ou das operações que serão realizadas posteriormente sobre os objetos, uma forma de representação será certamente melhor (ou mais eficiente) que outras.

A forma mais primitiva de representar um objeto, seja ele 2D e 3D, é a representação *wireframe* ou aramada, na qual o objeto é descrito apenas como um conjunto de vértices (pontos) e arestas (em geral segmentos de reta unindo os pontos).

Caso a aplicação trate objetos sólidos (objetos com superfícies que delimitem interior e exterior dos objetos) ao seu conjunto de vértices e arestas é agregado um conjunto de faces, cada uma delas constituída por arestas. Se os objetos forem de classes geométricas conhecidas como esferas, cilindros, elipsóides, não há a necessidade de sua representação através de uma aproximação por facetas planares; basta armazenar os parâmetros necessários para a sua interpretação, por exemplo, posição e raio para uma esfera, posição, raio da base e altura, para um cilindro.

2.8.3 - Utilização do computador em visualização 3D

Existe um grande problema na utilização do computador para a visualização de imagens tridimensionais. O computador é um recurso bidimensional. Não há como conseguir gerar efeito real de profundidade em um recurso bidimensional.

Na busca pelo realismo, muitos trabalhos tem sido feitos na simulação de fenômenos e objetos naturais. A motivação para tais trabalhos é a impossibilidade de representar tais objetos e fenômenos utilizando as técnicas conhecidas de modelagem geométrica aliadas aos algoritmos de síntese de imagens. Uma aparência convincente de madeira ou mármore pode ser obtida com técnicas de mapeamento de textura. Efeitos atmosféricos como nuvens, neblina e fumaça podem ser modelados com sistemas de partículas e fractais. A simulação da água e do movimento das ondas tem levado a inúmeros modelos aproximados, para evitar a dinâmica de fluidos computacionais com suas soluções proibitivas em termos de tempo. São encontrados modelos analíticos, modelos com campos de altitude (como terrenos) e modelos com sistemas de partículas. No entanto, existem algumas maneiras de simular a visão em 3D utilizando sombras, perspectiva, pontos iluminados e tonalidades, dando a impressão de profundidade.

2.8.4 - Animação

Para a criação de uma animação é necessário definir um cenário, um contexto, e, a partir daí, detalhar cenas e ações, ou seja, o que será animado e quais as ações que fará. Esta constitui a parte de modelagem da animação. Feito isso, parte-se para a animação propriamente dita.

A produção de cenas em VRML é baseada na definição de nodos. Cada nodo tem um nome, um tipo e define valores iniciais para seus parâmetros. Esses parâmetros podem ser de 2 tipos: *fields* e eventos. Os *fields* podem ser de dois tipos: *fields* (atributo do tipo *private*) e *exposedFields* (atributo do tipo *public*). Os eventos podem ser enviados de um nodo para outro através de um *eventOut* e recebê-lo através de um *eventIn*. Os eventos sinalizam mudanças causadas por estímulos externos e podem ser propagados

para os nodos usando rotas, que conectam um *eventOut* a um *eventIn* do mesmo tipo. Eventos e rotas definem a animação no mundo virtual.

Existem vários tipos de nodos em VRML, mas os particularmente importantes para animação são os sensores e os interpoladores. Sensores geram eventos baseados em ações do usuário, são responsáveis pela interação em VRML. Um sensor produz um *eventOut* a partir de uma ação do usuário (existem vários tipos de sensores e cada um reage a uma ação diferente), este *eventOut* pode ser dirigido, “roteado” para um outro *eventIn* e causar o início de uma animação.

Interpoladores definem pontos chave (*keyframes*) da animação que serão interpolados por uma função linear. A mistura de rotas, interpolação, aplicação de texturas, agrupamentos de nodos, podem gerar efeitos de animação muito interessantes. Porém, as rotas e os eventos são limitados. Para dar vazão a mais tipos de movimentos e interação, a linguagem VRML apresenta um outro tipo de nodo, chamado *script*, que permite conectar a animação a um programa na qual os eventos são processados.

2.8.5 - Raytracing e raycasting

A Computação Gráfica se divide, basicamente, em duas áreas: Modelagem e Visualização. A primeira está relacionada com a construção dos objetos da cena a ser visualizada, utilizando bases matemáticas para isso. A segunda busca uma forma de representação visual dos objetos construídos.

Raytracing é um poderoso método que permite a criação de excelentes imagens foto-realísticas em computadores. Atualmente, existem diversos algoritmos de visualização de superfícies e sólidos, porém nenhum deles é tão simples e tão poderoso quanto o ray tracing. O algoritmo se baseia numa idéia muito simples: um observador se senta em frente a uma tela plana transparente. De seus olhos partem diversos "raios visuais" que vão atravessar os pontos da tela e bater nos objetos tridimensionais, que foram definidos utilizando-se alguma técnica de modelagem. Pinta-se, então, o ponto da tela que foi atravessado pelo raio com a cor do objeto que foi atingido por este. Esta é a forma mais simples de se apresentar o algoritmo de ray tracing. Variações mais avançadas que incluem

sombras, reflexão e transparência serão apresentadas adiante. Em termos algorítmicos, pode-se apresentar o ray tracing da seguinte forma:

Algoritmo básico de ray tracing :

Para cada ponto da tela:

- Calcula-se uma linha reta unindo o olho do observador a este ponto
- Descobre-se as interseções desta reta com os objetos 3D que estão atrás da tela
- Pinta-se o ponto com a cor do objeto mais próximo

Alguns aspectos interessantes devem ser analisados:

- 1) O algoritmo de ray tracing gasta entre 75% e 95% de seu tempo determinando as interseções com os objetos, por isso, a eficiência da rotina de interseção raio-objetos afeta significativamente a eficiência do algoritmo. Atualmente existem hardwares que fazem estes cálculos com extrema rapidez.
- 2) Os objetos da cena a ser visualizada são descritos sob a forma de estruturas de dados. Essas estruturas têm uma certa forma, dependendo do método de modelagem a ser utilizado.
- 3) Quando o raio visual atinge um objeto visível, o ponto da tela a ser pintado possui características do ponto do objeto que foi atingido, mas não é necessariamente da cor do objeto. Diversos fatores influem no cálculo da cor do ponto, como a iluminação, por exemplo. Se um objeto for girado, a impressão visual da cor de um ponto deste muda, ficando mais escuro à medida que a luz que incide sobre ele fica mais perpendicular em relação à normal no ponto. A textura e rugosidade do objeto são fatores que também influenciam no cálculo da sua cor.
- 4) Observador (ou câmera) é constituído de um ponto focal (ou olho) e de uma tela (ou plano de projeção) composta de pequenos retângulos. Esta tela de retângulos geralmente é a própria tela do monitor e os retângulos são os seus pixels. Os objetos visíveis, ou cena, estão em frente à câmera, dentro da pirâmide de visão da Fig.32. Os raios visuais que são lançados passam pelos retângulos do plano de projeção e ligam o ponto focal à cena. Quando o ponto focal está localizado no infinito, os raios tornam-se paralelos e então a pirâmide de visão torna-se o paralelepípedo de visão da Fig.32.
- 5) Como pode-se perceber, os raios lançados no método de ray tracing seguem o

caminho inverso ao da luz. Daí surge a pergunta de por que não lançá-los na direção tradicional. A resposta é simples: de uma fonte luminosa partem bilhões de fótons, que são os formadores dos raios luminosos. Porém apenas uma pequeníssima parte deles chega aos olhos de um observador. Simular tal mecanismo seria impraticável para o computador, fazendo que uma simples cena levasse anos para ser renderizada. Lançando raios a partir do observador, garantimos que estaremos calculando apenas os raios que efetivamente serão captados pelo observador.

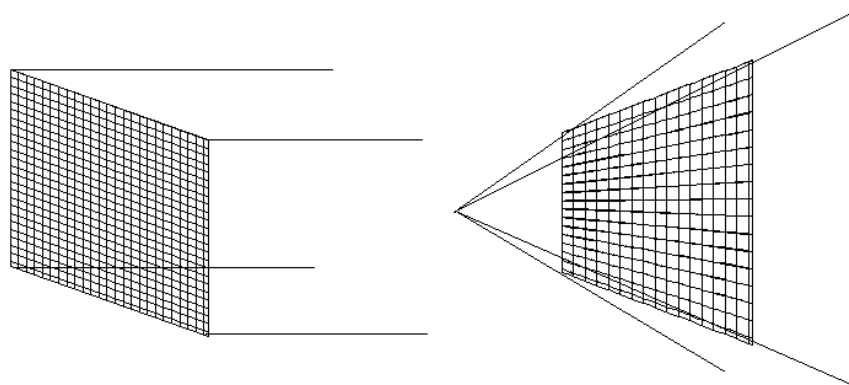


FIGURA 32: Pirâmide de visão

A Fig.33 nos mostra imagens geradas por um ray tracer que utiliza o algoritmo descrito acima. A versão simplificada do algoritmo de ray tracing é conhecida também como ray casting.

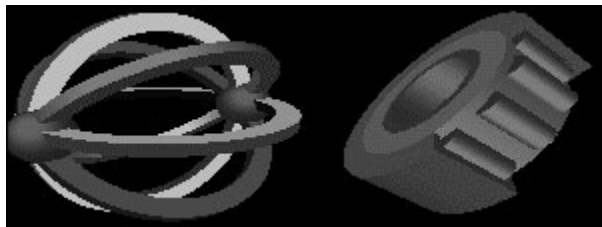


FIGURA 33: Exemplo de ray tracing

2.9 - Verificação de consistência em imagens e correção de falhas de segmentação baseada em conhecimento

É apresentada neste trabalho uma solução para resolver problemas surgidos da segmentação feita em imagens de cortes tomográficos. Esta solução foi desenvolvida por

Peter Kondrat, do departamento de informática. Universidade de Kaiserslautern em 1996, e está sendo aplicada no presente trabalho.

Os métodos de processamento de imagens utilizados na área da segmentação tendem muitas vezes a produzir segmentos cujas bordas vão além dos limites dos objetos que eles representam, ou seja, o segmento vaza. Isso significa que a um segmento são atribuídos além da área que se gostaria que fosse também áreas da imagem que já pertencem a outros componentes da imagem. Uma estratégia para se evitar esse problema é a de se gerar as assim chamadas sobre-segmentações. Nas sobre-segmentações uma imagem é dividida em muitos segmentos relativamente pequenos aonde também regiões que do ponto de vista semântico representam um mesmo componente de imagens são divididas em partes.

Estes problemas acabam por invalidar a segmentação, para resolver isto é necessário fazer uso de uma rotulação consistente, nesta, é feita uma verificação de toda a região onde se encontra o "objeto" que se deseja segmentar. Este procedimento, permite identificar os fragmentos de imagens não somente através de sua tonalidade, mas também através de outras várias informações, como posição, tamanho, etc, possibilitando desta forma a junção de fragmentos da mesma imagem.

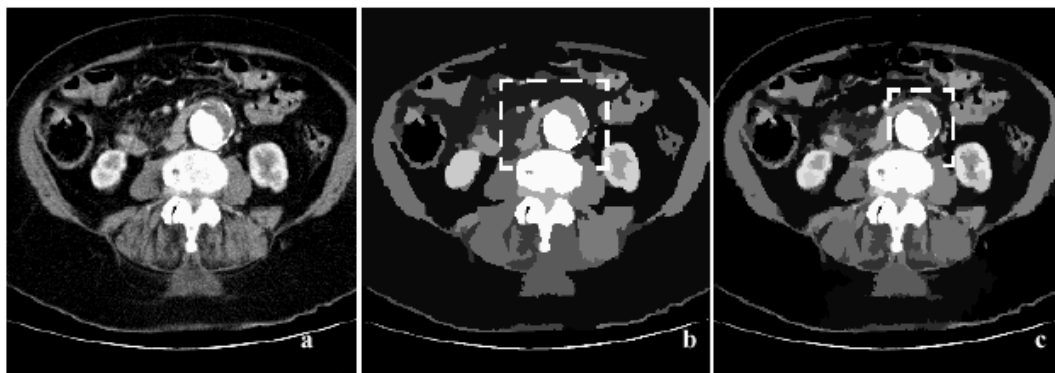


FIGURA 34: Corte original de uma TC da aorta (a) e duas diferentes segmentações realizadas com Mumford&Shah: (b) mostra um "vazamento" na segmentação no tecido aneurismático; (c) não apresenta "vazamento" mas divide o trombo mural em diferentes segmentos

2.9.1 - Avaliação de expectativas generalizada

Uma expectativa gerada através do auxílio de uma base de conhecimentos contém informações semânticas a respeito dos objetos que se espera reconhecer em uma imagem. Os algoritmos de processamento de imagem têm o objetivo de extrair informações mais abstratas de uma imagem, por exemplo, sob a forma de bordas ou segmentos unidos. A tarefa da avaliação de expectativas é a de satisfazer uma expectativa gerada a partir de uma base de conhecimentos através da utilização dos resultados da análise de imagens, isto é, encontrar um mapeamento ou uma atribuição adequada entre os componentes da expectativa por um lado e as informações extraídas da imagem por outro lado, ou então, tem por objetivo refutar a expectativa caso não seja possível encontrar-se um mapeamento adequado.

Para isto é primeiramente necessário que se encontre uma representação adequada das expectativas e também dos resultados do processamento de imagem. No âmbito deste trabalho deve ser concebido e implementado um algoritmo que dada uma determinada expectativa e uma determinada representação ele seja capaz de verificar se ela está satisfeita ou de refutá-la, caso seja possível implementar-se um algoritmo desses.

A avaliação de expectativas pode então, através da utilização do conhecimento de domínio de imagem existente, determinar quais segmentos de imagem poderiam pertencer a um mesmo objeto e então novamente unificá-los. Na verdade essa união poderia em muitos casos também ser atingida através de uma nova segmentação da imagem com segmentos mais adequados. Porém, a avaliação de expectativas através da utilização do conhecimento de domínio, tem a vantagem de poder utilizar muito mais informação nesse processo de atribuir sub-segmentos ou regiões de imagem a determinados componentes que um algoritmo de processamento de imagens e sabe-se que em muitos casos não se consegue utilizar ou encontrar parâmetros que permitam que seja gerada de uma vez só uma segmentação toda ela consistente. Em parte devido aos algoritmos de segmentação não conseguirem utilizar esse conhecimento.

O objetivo da análise de imagens é o reconhecimento de componentes em uma imagem. Existem também entre essas componentes algumas que são opcionais e que são de interesse bastante especial já que a sua existência ou não existência na verdade repre-

senta a informação que no fundo se procura através do processamento de imagem. Assim, no domínio de aplicação de Cyclops a existência de determinadas características pode ser um indicativo de determinada doença. E é geralmente um processo assim que se vai executar. A avaliação de expectativas pode então, quando lhe são dadas informações adequadas sobre possíveis patologias, procurar em uma imagem de uma forma explícita por indicadores de que essas características existam. Isso significa que encontrar estas características opcionais, mas que têm características patológicas, é mais importante do que não encontrá-las e dá-se preferência a procura por estas.

2.9.2 - Expectativas

O modelo deve ter uma descrição das estruturas anatômicas a serem encontradas na imagem, com uma lista de parâmetros descrevendo o aparecimento destas estruturas, tal que uma identificação automatizada e o casamento entre o modelo e segmentos resultantes sejam possíveis. Este modelo deve permitir também a análise de imagens médicas onde são procurados patologias, mas isso não está necessariamente presente na imagem, habilitando a descrição de estruturas "obrigatórias" e "opcionais". Obrigatório são todas as anatomias que têm que ser encontradas nesta imagem segmentadas em particular, opcional são todas as estruturas patológicas, que podem estar na imagem, mas não necessitam obrigatoriamente estar nela.

A técnica que tentará evitar que um segmento contenha áreas pertencentes a objetos diferentes, deve usar parâmetros mais sensíveis, desta forma a segmentação fará vários produtos de um objeto, mas nenhum que contenha partes de objetos diferentes. O método utilizado neste trabalho deve usar o modelo para "classificar" e para fundir este objetos separados da imagem ao mesmo objeto. Também deve prover uma forma de executar isto de um modo totalmente automático, sem intervenção do usuário a menos que uma correlação aceitável da segmentação não seja possível de ser encontrada.

A abordagem foi desenvolvida como uma extensão do método de rotulação consistente inexata desenvolvida por Haralick e Shapiro. Este método foi estendido para habilitar o uso de a) modelos complexos de resultados esperados onde cada unidade tem um rótulo b) habilitar o uso do rótulo "opcional" representando patologias. A representação

de unidades é terminado com um parâmetro padrão de segmentos extraídos por diferentes algoritmos, onde alguns deles são gerais e outros são tarefas de diagnóstico específicas.

A priori a rotulação é determinada pela rede neural backpropagation. Mudando a priori um rótulo o erro aumentará. Caso contrário, se são fundidos dois segmentos e o resultado é mais semelhante ao modelo do protótipo, o erro global diminui.

O modelo pode ser gerado de uma representação estática em uma base de dados ou por estrutura de redes bayesianas usando uma aproximação baseada no modelo de Shastri e parâmetros sobre a posição dos cortes de tomografias computadorizadas ou ressonância magnética. Baseado no tipo de exame que é executado e na posição dos cortes inicialmente examinados, a estrutura gráfica representando a expectativa de resultados da imagem processada é gerada e passada para o módulo de detecção de falhas. Cada laço deste gráfico contém um objeto descrevendo uma estrutura anatômica ou patológica por meio de um conjunto de parâmetros revisto em uma base de dados de domínio. Os parâmetros são descritos no Quadro 4.

QUADRO 4: Parâmetros usados para representar "views" de anatomias em imagens radiológicas

Parâmetro	Descrição	Tipo
Bounding Box	Retângulo paralelo aos eixos xy, descrevendo os mais simples cascos convexos possíveis para a imagem	Par de coordenadas x, y
Área	Área de pixels representando a anatomia. Pode ser qualquer declaração nos pixels ou nas coordenadas tomográficas	Real
Valor de cinza	Valores de cinza da anatomia. Valores absolutos de pixels.	Real
Vizinhos	Lista de todas as outras anatomias adjacentes	Lista de anatomias
Anatomia	Anatomia do domínio de aplicação representada pela imagem	Símbolo específico do domínio
Tipo	A estrutura pode ser opcional ou obrigatória. Estrutura opcional sempre se refere a patologias.	Símbolos

No Quadro 4, os parâmetros usados representam "views" de anomalias em imagens radiológicas como as armazenadas no domínio de uma base de dados de conhecimento. Os valores são gerados baseados nos valores significativos obtidos de uma base de imagens de tomografia computadorizada e ressonância magnética de diferentes pacientes.

Componentes não-veciniais não podem ser associadas com objetos que em uma imagem estão lado-a-lado. Isso representa uma restrição, objetos que na descrição da expectativa não são vizinhos não podem ser associados a áreas ou regiões de uma imagem que estão lado-a-lado. As componentes de expectativa são diferentes entre obrigatórias e opcionais. Isto é representado através do atributo *Tipo* da componente.

Deve ser observado que os valores concretos desses atributos em geral não correspondem exatamente aos valores dos objetos esperados na expectativa gerada (que representam um modelo, um valor ideal), mas sim, representam valores prototípicos a partir dos quais se pode desviar.

2.9.3 - Resultados do processamento de imagens

Para cada segmento gerado durante a segmentação uma lista de parâmetros é calculada, o qual é usado com a descrição de um segmento em uma rotulação consistente que será executado posteriormente. Para cada segmento durante o processo de rotulação, há parâmetros de descrição gerados parcialmente diferentes dos parâmetros da base de dados de domínio: área de *bouding box*, valor de cinza e vizinhos são computados o mesmo nodo; Anatomia e Tipo são substituídos por outros parâmetros.

O Quadro 5 mostra os parâmetros extras gerados por cada segmento e usados durante a rotulação. Para a pré-classificação da rede neural, alguns parâmetros de distribuição de valores de cinza como variância de valores de cinza são calculados. A tarefa da rotulação consistente é casar os resultados da segmentação com o modelo de domínio provida para esta imagem específica. Este casamento permite um erro, isto é, um salto para um threshold e é baseado nas diferenças entre parâmetros "ideais" de uma base de

dados já calculada. Se o erro sobe acima do threshold, a segmentação é considerada como falha.

QUADRO 5: Parâmetros gerados para cada segmento de imagem resultante de uma segmentação super sensitiva que difere dos protótipos de base de dados

Parâmetro	Descrição	Tipo
Centro de gravidade	Centro de massa de um segmento. Usado por um rótulo da unidade de restrição relacionada	Ponto x, y
Pré-classificação	Anatomia do domínio de aplicação provável sendo representada por este segmento, como classificado por um domínio de uma rede neural específica	Símbolo específico do domínio
Confiabilidade	Valor iniciado como confiável é o símbolo pré-classificado. Baseado no erro da rede neural.	Real

Ao contrário dos atributos descritos anteriormente e que, em parte, tem os mesmos nomes e que descreve as componentes de expectativa, os valores dos atributos associados ao segmentos descrevem valores concretos daqueles segmentos específicos e calculados para eles.

2.9.4 - Avaliação de expectativas

São atribuídos aos componentes de uma expectativa segmentos de uma imagem. A satisfação de uma expectativa é medida através de um mapeamento destes componentes quando para todo componente obrigatório da expectativa pelo menos uma imagem for encontrada e mapeada.

Neste processo são colocados uma série de requisitos para esse mapeamento: os atributos das componentes de expectativa precisam corresponder aos atributos dos segmentos de uma forma adequada. Como em imagens do mundo real muitos fatores externos e não controláveis agem sobre a aparência de uma imagem será muito raro haver uma correspondência exata entre os atributos de uma expectativa e os atributos calculados para um conjunto de segmentos de imagem. Portanto é necessário que se encontre

uma medida para a correspondência entre um segmento e uma componente de expectativa.

Esta medida é realizada em Cyclops através de uma medida de erro que permite que nos atributos utilizados existam diferenças relativas mas que porém em casos de desvios muito grandes façam com que este erro da medida de erro aumente de forma bastante forte. Além disso a necessidade de se mudar ou se desviar da pré-classificação de um segmento através da alteração do rótulo desse segmento também é castigada com um aumento bastante grande do erro. Esta medida de erro serve para então reduzir o espaço das possíveis satisfações de uma expectativa de tal forma que somente aqueles conjuntos de atribuições ou mapeamentos são permitidos nos quais a soma de todos os erros não ultrapasse um determinado limite.

Enquanto que a medida de erro utilizada depende de forma muito forte das necessidades do domínio de aplicação específico existe ainda mais uma restrição independente de domínio a respeito dos mapeamentos aceitáveis que deve ser considerada. Em determinadas situações a avaliação de expectativas também deve unificar diversos segmentos sob uma única componente de expectativa. Isto no entanto somente faz sentido quando os segmentos unificados são vizinhos entre si. Se relações de vizinhança entre todos os segmentos da imagem forem representados como um grafo, todos os segmentos atribuídos a um componente de expectativa precisam representar um grafo conexo. Isto faz sentido quando através da unificação de segmentos o erro resultante se reduz, ou seja, vários segmentos unificados em um segmento maior têm seus atributos como novo segmento maior recalculados e esse conjunto de atributos está mais próximo dos atributos ideais do componente da expectativa.

2.9.5 - Método de rotulação consistente

O problema de rotulação consistente CLP (*Consistent Labeling Problem*) é definido como a seguir. Dado um determinado problema de rotulação consistente $CLP=(U,L,T,R)$, deseja-se procurar um conjunto de todas as rotulações consistentes $f:U \rightarrow L$ que satisfaça o conteúdo especificado por T e R. Se não houver a possibilidade de haver uma rotulação consistente, o algoritmo retornará o conjunto vazio. O método

geral para resolver este problema é uma busca em árvore por *backtracking*. Como é um algoritmo de busca em árvore, pode ser melhorado adicionando-se métodos não heurísticos para podar a árvore. Este método pode incluir um procedimento chamado checagem à frente (*forward-checking*) e outro chamado relaxamento discreto (*discrete relaxation*) para ajudar acelerar a busca em árvore.

2.9.5.1 - A árvore de busca em *backtracking*

Uma árvore de busca em *backtracking* inicia com a primeira unidade de U. Esta unidade pode potencialmente combinar-se com cada rótulo do conjunto L. Cada uma dessas potenciais atribuições é um nodo de nível 1 da árvore. O algoritmo seleciona um destes nodos, faz as tarefas, seleciona a segunda unidade de U, e começa a construir os filhos do primeiro nodo, que são os nodos que mapeiam a segunda unidade de U para cada possível rótulo de L. Neste nível alguns dos nodos podem ser rejeitados porque eles violaram as condições. O processo continua para o nível $|U|$ da árvore. Os caminhos do nodo raiz para qualquer nodo escolhido do nível $|U|$ são os rótulos consistentes.

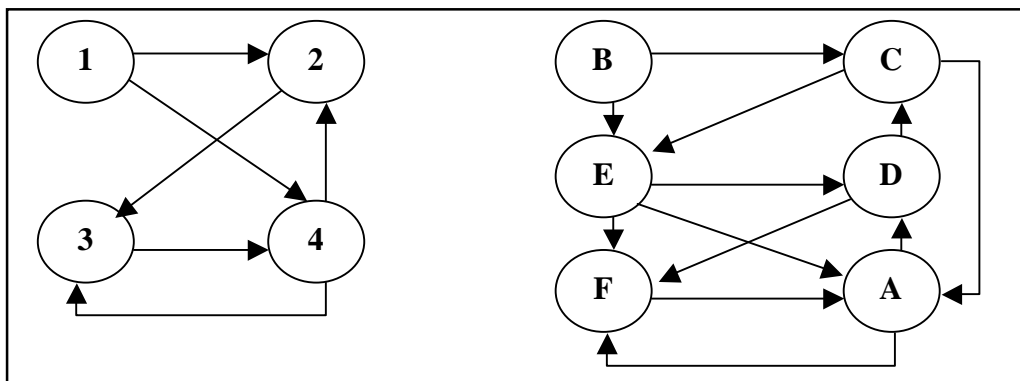


FIGURA 35: Problema digraph-matching - para procurar um sub-gráfico G2 que é isomórfico do gráfico G1

A Fig.35 ilustra um problema simples de *matching* de diagramas, e a Fig.36 mostra uma sessão da árvore de busca em *backtracking* tentando encontrar uma solução. O algoritmo para a árvore de busca em *backtracking* é fornecido abaixo. As entradas para o algoritmo são U, o conjunto de unidades; L, o conjunto de rótulos; T, a relação unidade-restrições; R, relação de restrições unidade-rótulo; e F, a rotulação acumulada

realizada até agora. Na chamada inicial para o procedimento, F deverá ser um conjunto vazio. Cada instanciação deste procedimento recursivo adiciona um par unidade-rótulo para a rotulação. Quando uma rotulação consistente completa é encontrada, o mesmo é impresso pelo procedimento.

```

procedure treesearch (U, L, f, T, R):
  u := first(U);
  for each l ∈ L do
    f' := f ∪ {(u, l)};
    OK := true;
    for each N-tuple (u1, ..., un) in T containing component u and whose
    other components are all in domain (f) do
      if ((u1, f'(u1)), ..., (un, f'(un))) is not in R
      then begin OK := false; break end;
    end for;
    if OK then
      begin
        U' = remainder(U);
        if isempty (U')
        then output (f')
        else treesearch (U, L, f', T, R);
      end
    end for
  end treesearch;

```

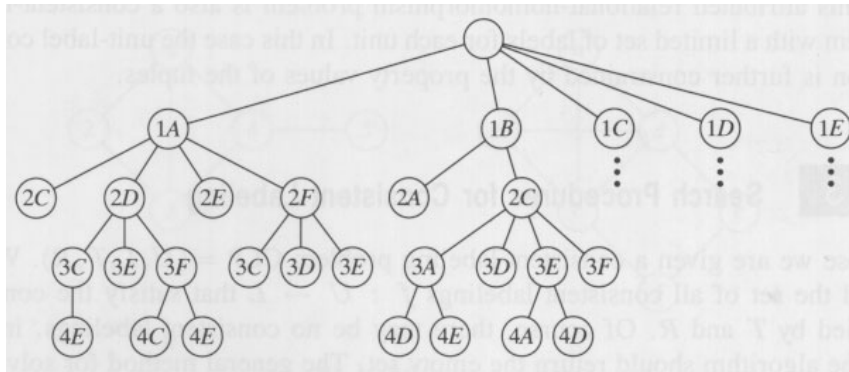


FIGURA 36: Árvore de busca em backtracking

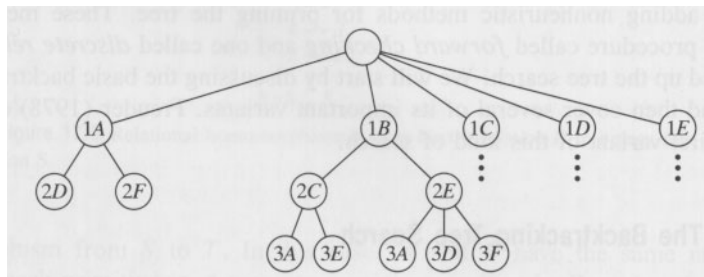


FIGURA 37: Árvore de busca em backtracking com checagem para frente

Nas Fig.36 e Fig.37 é mostrada a porção da árvore de procura para resolver o problema de *match* gráfico da Fig.35 usando *backtracking* simples e *backtracking* com checagem a frente. Nenhuma solução é encontrada na porção mostrada, visto que o mapeamento correto começa com (1, E).

2.9.5.2 - *Backtracking* com *forward-checking*

A árvore de busca *backtracking* tem complexidade de tempo exponencial, e embora não existam algoritmos polinomiais conhecidos em caso geral, existe um número de algoritmos que podem cortar o tempo de busca pela redução do tamanho da árvores onde é feita a busca. A checagem à frente (*forward-checking*) é um desses métodos. É baseado na idéia de que uma vez que um par de unidade-rótulo (u, l) é instanciada para um nodo em uma árvore, as condições impostas para a relação tornam impossível a instanciação de alguns pares unidades-rótulo (u', l') . Supondo ser (u, l) altamente instanciável em uma árvore e que a subárvore abaixo deste nodo contenha nodos com os primeiros componentes sendo u_1, u_2, \dots, u_n, u' . Embora (u', l') seja impossível para qualquer instanciação de u_1, u_2, \dots, u_n, u' , serão feitas tentativas pela árvore de busca *backtracking* em todo cominho que alcance este nível da árvore.

A Fig.38 ilustra este problema com um segmento de uma árvore de busca *backtracking* para o problema das 6 damas. Neste segmento os rótulos 1, 3, 5, e 6 para unidade 5 aparecem duas vezes no nível 5 da árvore. Mas a instanciação do par (1,1) no nível 1 rejeita os rótulos 1 e 5 para a unidade 5, e a instanciação do par (2,3) no nível 2 rejeita os rótulos 3 e 6 para a unidade 5. Na árvore de busca *backtracking* plena, isto é não detectado até que a unidade 5 seja instanciada no nível 5 da árvore. Desta forma testes redundantes são realizados para detectar uma dupla condição no nível 5 que possa ser prevista antecipadamente na árvore.

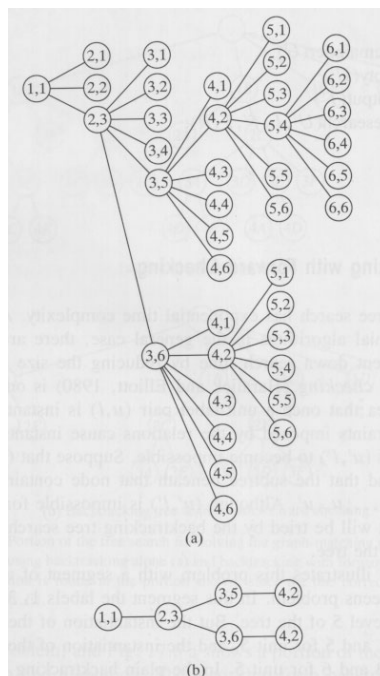


FIGURA 38: Segmento da árvore de busca para resolução do problema das 6 damas usando backtracking (a) e backtracking com checagem para frente (b)

O princípio da *forward-checking* é rejeitar um par (u', l') que é incompatível com um par (u, l) no instante que (u, l) é instanciado, e manter um registro daquela informação. A estrutura de dados utilizada para armazenar informações é chamada *future-errors-table* (FTAB - tabela de erros futuros). Há uma tabela de erros futuros para cada nível de recursão da árvore de busca. Cada tabela é uma matriz possuindo uma linha para cada elemento de U e uma coluna para cada elemento de L . Para qualquer unidade não instanciada ou unidade futura $u' \in U$ e rótulo potencial $l' \in L$, $FTAB(u', l')=1$ se ainda é possível instanciar (u', l') , dado o histórico das instanciações já realizadas. $FTAB(u', l')=0$ se (u', l') já foi rejeitada por já pertencer a alguma atribuição. Quando um par (u, l) é instanciado pela árvore de busca *backtracking*, o procedimento de *forward-checking* é chamado para examinar todos os pares (u', l') das unidades futuras e seus rótulos remanescentes possíveis. Para cada par (u', l') que é incompatível com a atribuição de (u, l) , *forward-checking* define $FTAB(u', l')$ para 0. Assim, após o *forward-checking* ser completada, todos os futuros pares unidades-rótulos (u', l') que são incompatíveis com a atribuição corrente (u, l) ou qualquer atribuição prévia tem suas posições definidas FTAB em zero. Se para qualquer unidade futura u' , $FTAB(u', l')$ se torna 0 para todos rótulos l' (pertence) L , então a instanciação de (u, l) falha imediata-

mente, porque não há rótulos para unidade u' que sejam compatíveis com as atribuições prévia e corrente. A Fig.38 mostra como a árvore de busca *backtracking* gerada é reduzida pelo *forward-checking* no exemplo do problema das 6-damas, onde os 39 nodos foram reduzidos para 6. A Fig.36 mostra a árvore gerada pela árvore de busca *forward-checking* para a mesma porção do problema de matching em diagramas como mostra na Fig.35. Aqui os 28 nodos mostrados foram reduzidos para 14. O Quadro 6 mostra o estado da tabela de erros futuros durante a busca na árvore por *forward-checking* na Fig.38b.

QUADRO 6: Estado da tabela de erros futuros durante a busca em árvore *backtracking* com *forward-checking*

1A

	A	B	C	D	E	F
2	X	X	0	1	0	1
3	X	X	1	1	1	1
4	X	X	0	1	0	X

2D

	A	B	C	D	E	F
3	X	X	1	X	0	1
4	X	X	0	X	X	X

2F

	A	B	C	D	E	F
3	X	X	0	0	0	X
4	X	X	0	1	0	X

Falha

1B

	A	B	C	D	E	F
2	0	X	1	0	1	0
3	1	X	1	1	1	1
4	0	X	1	0	1	X

QUADRO 6: Estado da tabela de erros futuros durante a busca em árvore *backtracking* com *forward-checking*

2C

	A	B	C	D	E	F
3	1	X	X	0	1	0
4	0	X	X	1	0	X

3A

	A	B	C	D	E	F	
4	0	X	X	0	0	X	Falha

3E

	A	B	C	D	E	F	
4	0	X	X	0	0	X	Falha

2E

	A	B	C	D	E	F
3	1	X	0	1	X	1
4	0	X	1	0	X	X

3A

	A	B	C	D	E	F	
4	0	X	0	0	X	X	Falha

3D

	A	B	C	D	E	F	
4	0	X	0	0	X	X	Falha

3F

	A	B	C	D	E	F	
4	0	X	0	0	X	X	Falha

O procedimento de busca em árvore por *forward-checking* é dado abaixo. Os parâmetros de entrada U , L , f , T e R são os mesmos utilizados para a árvore de busca em *backtracking*. O parâmetro de entrada adicional FTAB é a tabela de erros futuros, o qual é implementado como uma matriz. A tabela inicial de erros futuros será um vetor $|U| \times |L|$ com todos os conjunto de elementos do vetor definidos em 1. Antes de cada chamada recursiva, o procedimento copia a tabela de erros futuros corrente FTAB para uma tabela temporária NEWFTAB, atualiza a tabela temporária utilizando *forward-checking*, e passa a tabela temporária para a próxima instânciação. Isto essencialmente cria uma pilha de uma tabela de erros futuros. Em linguagens na qual listas são estruturas de dados primárias, a futura tabela de erros poderia ser implementada como uma lista de listas ou um vetor de listas.

```

procedure forward_checking_treearch (U, L, f, FTAB, T, R);
u := first(U);
for each l ∈ L do
  if FTAB(u,l) == 1 then
  begin
    f' := f ∪ {(u, l)};
    U' := remainder(U);
    if isempty (U') then output (f')
    else
    begin
      NEWFTAB := copy (FTAB);
      OK := forward_check(NEWFTAB, u, l, U', L, T, R, f');
      if OK then forward_checking_treearch (U, L, f, FTAB, T, R);
    end
  end
end for
end forward_checking_treearch;

function forward_checking(FTAB, u, l, future_units, L, f', T, R);
for each u' ∈ future_units do
  forward_checking := false;
  for each l' ∈ L with FTAB (u', l') == 1 do
    if compatible (u, l, u', l', T, R, f')
    then forward_check := true
    else FTAB (u', l') := 0
  end for
  if forward_check == false then break
end for
end forward_check;

```

2.9.6 - Complexidade

O problema de rotulação consistente é um problema NP completo. Assim qualquer algoritmo para resolver este problema geral em um processador seqüencial apresenta complexidade exponencial. No pior caso, onde $R = (U \times L)^n$ cada nodo não folha terá exatamente $|L|$ sucessores. Assim haverá um nodo (raiz) no nível 0, $|L|$ nodos no nível 1, $|L|^2$ nodos no nível 2,... e $|L|^m$ nodos no nível M, para um total de $(|L|^{m+1}-1)/(|L|-1)$ nodos. Na maioria dos problemas reais as condições causam algum *backtracking*. Enquanto a árvore completa não for gerada, a árvore de busca em *backtracking* ainda terá complexidade exponencial. A *forward-checking* e *look-ahead checking* podem ambos reduzir drasticamente o número de nodos buscados mas não alteram a complexidade global do problema. É mais interessante considerar o número atual de checagens de consistência, o qual é diretamente relacionado ao tempo de execução do algoritmo, tanto em aplicações particulares ou para um modelo abstrato que mostra a probabilidade de uma dada checagem de consistência ter sucesso ou falha.

2.9.6.1 - O problema de rotulação consistente inexato

O problema da rotulação consistente requer todas as condições de um problema sejam satisfeitas por alguma solução. Quando nos tentamos aplicar técnicas para resolver problemas de rotulação consistente para dados extraídos de imagens reais, logo se torna aparente que está se esperando muito. Supondo que se esteja tentando encontrar uma cópia de um modelo, expresso como um desenho de uma linha, em uma estrutura de dados de um segmento de linha extraído de uma imagem real. Embora um humano possa reivindicar que consegue "ver" todas as linhas do modelo na imagem original, a estrutura extraída geralmente terá linhas que estarão faltando, parcialmente faltando, linhas extras, e linhas distorcidas. Assim a busca em árvore para procurar um mapeamento das linhas do desenho para os segmentos extraídos da imagem é provavelmente falha se insistir em um rótulo para cada unidade e cada exigência sendo satisfeita. Para tais problema da vida real, é preciso se formular um modelo de rotulação consistente que permita levar alguns

erros em conta. O modelo resultante é chamado de "Problema de Rotulação Consistente Inexato" conforme HARALICK&SHAPIRO (1980).

Um problema de rotulação consistente inexato tem a forma $ICLP=(U,L,T,R,w,e)$, U, L, T , e R são os mesmo que no problema de rotulação consistente normal ou exato. A função w é um mapeamento de $T \times L_n$ para um intervalo $| 0 \ 1 |$. Em geral, $w(u_1, \dots, u_n, l_1, \dots, l_n)$ é o erro que resultaria ao associar unidades (u_1, \dots, u_n) com rótulos (l_1, \dots, l_n) . A função w normalmente dependerá da aplicação do problema. Uma forma útil de definir w é atribuir um peso $weight(t)$ para cada linha $t \in T$. Os pesos modelam o conceito de que algumas relações são mais importantes que outras. Então $w(u_1, \dots, u_n, l_1, \dots, l_n)$ é definido como $weight[(t_1, \dots, t_n)]$ para qualquer conjunto de rótulos (l_1, \dots, l_n) . Quando as aplicações não ditam um conjunto óbvio de pesos, a cada linha pode ser atribuído o peso $1 / | T |$. Uma rotulação consistente inexata é um mapeamento $F: U \rightarrow L$ onde a soma dos erros incorridos por F em todas as n -linhas de unidades que exigem uma outra é menor do que um dado limiar e . Assim na notação acima, o mapeamento F precisa satisfazer:

$$\sum_{(u_1, \dots, u_n) \in T} w[u_1, \dots, u_n, f(u_1), \dots, f(u_n)] \leq \varepsilon \quad (19)$$

Todos os procedimentos de busca em árvore dados acima podem ser estendidos para uma estrutura de rotulação consistente inexata. Será demonstrada a extensão para o caso da árvore de busca por *forward-checking*. Além de conhecer a função de peso w e o limiar de erro ε , a árvore de busca for *forward-checking* inexata trabalha com três tipos de erros: passado, corrente e futuro. A variável *past_error* é uma entrada para o procedimento de busca em árvore, representando o erro de um mapeamento parcial que foi construído até agora. A árvore de busca deve ser chamada inicialmente com um valor 0 para *past_error*, e ao valor da variável nunca é permitido exceder o limiar de erro ε . Como pares de unidades-rótulos são adicionadas ao mapeamento e algumas exigências não são satisfeitas pela função resultante, *past_error* será incrementado.

A variável *current_error* é uma variável local para o procedimento de busca em árvore representando o erro associado com o par corrente (u, l) . Isto é o erro que a adição do par (u, l) ao mapeamento parcial iria somar ao erro já associado com o mapeamento parcial. Na hora de considerar um par (u, l) , a árvore de busca não tem que computar seu

current_error. Ele foi gradualmente computado por todo o procedimento de atualização, o qual é chamado cada vez que um par é adicionado ao mapeamento. O valor do *current_error* para um par (u, l) é encontrado na tabela de erros futuros na posição FTAB (u, l) . Assim na árvore de busca por *forward-checking* inexata, a tabela de erros futuros contém números reais ao invés de 0s e 1s. A tabela é inicializada em zero (sem erros), e o valor de um par (u', l') aumenta sempre que um par unidade-rótulo (u, l) que é inconsistente com (u', l') é somado ao mapeamento sendo construído. Além disso, FTAB é ampliado por um vetor extra MINERR. MINERR(u) é usado para guardar o erro mínimo de todos os rótulos para uma determinada unidade. É também inicializado com zero.

Finalmente, a terceira variável *future_error* representa um possível erro que pode ser incorrido pela instanciação de futuras (ainda não instanciada) unidades. Desde que uma tabela de erros futuros associa um erro acumulado com cada unidade futura e possível rótulo baseado na compatibilidade entre esse futuro par unidade-rótulo e um mapeamento parcial, *future_error* pode ser estimado pela soma de todas unidades futuras u de MINERR(u) - o erro mínimo para qualquer rótulo de u . Esta soma é garantida não ser maior que o erro futuro real, o qual tem para levar em conta não somente o erro causado pela interação das unidades futuras com unidades passadas, mas também o erro causado pela interação entre as unidades futuras. O algoritmo de *forward-checking* não investiga o tipo posterior do erro; isto é deixado para procedimentos *look-ahead* tal como psi. A variável *future_error* é também definida em zero para a chamada inicial ao procedimento de busca em árvore.

Os procedimentos seguintes - *inexact_forward_checking_treesearch* e *inexact_forward_checking* - são modelados depois da *forward_checking_treesearch*, *forward_checking*, e procedimentos compatíveis dados na seção anterior. Eles usam o conceito de valores reais para a tabela de erros futuros (FTAB ampliado pela MINERR) e erros passados, erros correntes e erros futuros para encontrar rotulações consistentes inexatos dado o conjunto de unidades U , o conjunto de rótulos L , a relação unidade-condição T , a relação de condição unidade-rótulo R , a função de peso w , e o limiar de erro ϵ .

```

procedure inexact_forward_checking_treesearch (U, L, f, FTAB,
MINERR, T, R, w, past_error, future_error, epsilon);
u := first(U);
for each l  $\in$  L do
    current_error := FTAB(u, l);
    if past_error+current_error+future_error - MINERR(u) <= epsilon
then

```

```

begin
  f' := f ∪ {(u, l)};
  U' := remainder(U);
  if isempty (U') then output (f')
  else
    begin
      NEWFTAB := copy (FTAB);
      new_future_error := inexact_forward_check(NEWFTAB,
MINERR, u, l, U', L, T, R, w, past_error, f');
      if new_future_error+past_error+current_error-
MINERR(u) <= epsilon then inexact_forward_check_treearch(U', L,
f', NEWFTAB, MINERR, T, R, w,
past_error+current_error, new_future_error, epsilon);
    end
  end
end for
end inexact_forward_checking_treearch;

function inexact_forward_checking (FTAB, MINERR, u, l,
future_units, L, T, R, w, past_error, f');
inexact_forward_check := 0;
for each u' ∈ future_units do
  smallest_error := 9999999;
  for each l' ∈ L with past_error+inexact_forward_check+FTAB
(u', l') <= epsilon do
    error := inexact_compatible(u, l, u', l', T, R, w, f');
    FTAB (u', l') := FTAB (u', l') + error;
    if FTAB (u', l') < smallest_error
    then smallest_error := FTAB (u', l')
  end for
  MINERR(u) := smallest_error;
  inexact_forward_check := inexact_forward_check +
smallest_error;
  if inexact_forward_check + past_error > epsilon then break
end for
end inexact_forward_check;

```

Supondo que adicionamos um peso de $1/6$ para cada um dos arcos do diagrama $G1$ na Fig.35 e desejamos procurar um subgráfico do diagrama $G2$ que é "quase isomórfico" para $G1$. Se definirmos "quase isomórficos" como não faltando mais que dois arcos, podemos definir ϵ como $2/6$ e resolver o correspondente problema da rotulação consistente inexato. O Quadro 7 ilustra o procedimento de *forward-checking* para uma pequena porção da árvore de busca abaixo do nodo (1, B) para resolver este problema. Observa-se que a versão inexata da busca em árvore não pode eliminar tão rapidamente como a versão exata. O rótulo D para a unidade 4 não é eliminado quando (1, B) é instanciado primeiro. Ao invés disto é simplesmente acumulado um erro de $1/6$. Isto não é até (1, B) e (2, A) terem sido instanciados que (4, D) é rejeitado, desde que a soma destes erros de

$2/6$ e os erros já incorridos por $F=\{(1,B), (2, A)\}$ de $1/6$ somado a $3/6$, o qual é maior que ϵ .

QUADRO 7: Parte da tabela de erros futuros para o problema da Fig.35

1B

	A	B	C	D	E	F	
2	1/6	X	0	1/6	0	1/6	Erro passado=0
3	0	X	0	0	0	0	Erro presente=0
4	1/6	X	0	1/6	0	1/6	Erro futuro=0

2A

	A	B	C	D	E	F	
3	X	X	1/6	0	1/6	0	Erro passado=0
4	X	X	0	2/6	0	1/6	Erro presente=1/6
							Erro futuro=0

2.9.7 - Extensões ao modelo de Haralick&Shapiro

A partir das tarefas da avaliação de expectativas descritas na sessão vamos ter algumas características especiais na solução do CLP especial descrito acima. Essas características são:

Merging: como merging entende-se a unificação ou fusão de dois segmentos ou componentes de uma imagem em um único segmento ou componente. Merging acontece quando a um componente da expectativa são atribuídos mais de um segmento de imagem.

Rótulos: os componentes de uma expectativa podem ser ou opcionais ou obrigatórios. O algoritmo de Haralick e Shapiro no entanto parte do princípio de que todos os rótulos de uma certa forma são opcionais. Uma solução possível deste problema poderia ser o fato de que no momento de se fazer o cálculo do erro futuro acerca de rótulos obrigatórios ainda não atribuídos (instanciados) ao invés de fazê-los como Haralick e Shapiro o fazem sobre as unidades ainda não rotuladas.

Unidades: O problema complementar ao item acima é encontrado também nos segmentos de imagem (nas unidades). Enquanto Haralick e Shapiro partem do princípio que toda unidade precisa receber um rótulo é possível em Cyclops que para um determinado segmento de imagem não se encontre nenhum rótulo (por exemplo, se for algum ruído na imagem). Isso então faz com que as unidades sejam utilizadas somente de forma indireta para a estimativa do erro futuro, quando para um rótulo obrigatório ainda não instanciado só existir uma só unidade que sirva, assim será feito com que o erro que vai ocorrer desta atribuição seja obrigatoriamente incluído na soma dos erros.

Independência do domínio de aplicação: uma característica muito importante do sistema Cyclops é a sua capacidade de ser adaptado às mais diferentes aplicações e domínios de aplicação dentro da área de análise de imagens. Existe a possibilidade de em Cyclops se utilizar, dependendo do domínio da aplicação, o uma base de conhecimento de domínio específica para aquele domínio que possui conhecimento codificado específico para aquela aplicação. Nas sessões anteriores foi citado várias vezes que também na avaliação de expectativas componentes dependentes de domínio podem ser necessárias que serão representadas adequadamente na base de conhecimento de domínio de acordo com a necessidade que venha a ocorrer. Ademais é necessário que estas partes dependentes de domínio específica da avaliação de expectativas sejam criadas, modificadas e adequadas de forma simples e transparente.

A linguagem de implementação utilizada em Cyclops, Smalltalk, permite aqui que se realize uma solução bastante elegante: existe a possibilidade para que se criem os assim chamados blocos (de código), que são incluídos em vários pontos do algoritmo que então durante o tempo de execução são utilizados pelo programa no ponto adequado. Esses pontos podem ser utilizados como funções "normais" ou chamados como métodos com parâmetros e resultados podem ser devolvidos para o métodos que os chamou. Por outro lado estes blocos podem ser trabalhados como dados, quer dizer, eles podem ser modificados de forma extremamente fácil, da mesma forma como texto e podem ser colocados em um banco de dados de onde são buscados de acordo com a aplicação que se tem no momento.

Desta forma é possível que blocos dependentes de domínio específicos para aplicações determinadas sejam passados como parâmetros da avaliação de expectativas. Através da determinação de um esquema único para a chamada destes blocos em especial

com vistas à passagem dos parâmetros adequados faz-se com que se garanta a consistência dos blocos com o sistema como um todo sem que se crie restrições muito fortes para a adaptação do sistema em domínios de aplicação diferentes.

As modificações para implementar essas características se referem essencialmente aos seguintes pontos.

- . estimativa do erro esperado.
- . levar em conta mais de uma unidade em um mesmo nível da árvore de busca
- . unidades não rotuladas, ou seja, dar continuidade à recursão dentro da árvore sem que seja criado um par unidade-rótulo.
- . a existência de rótulos obrigatórios.
- . unificação de unidades somente quando através dessa unificação haja uma redução do erro.

A seguir vamos detalhar esses pontos.

2.9.7.1 - Estimativa do erro

A estimativa de erro não pode mais ser realizada com base nas unidades não rotuladas. Muito mais necessário é levar em consideração, agora, os rótulos obrigatórios ainda não instanciados.

Uma outra dificuldade ocorre através do fato de que o erro associado a um rótulo já instanciado pode se reduzir quando outras unidades também receberem esse mesmo rótulo, pelo fato de o conjunto das unidades estar mais próximo do protótipo na expectativa. A unificação de vários segmentos de imagem que se faz então, pode levar ao fato de que os atributos do novo segmento que surgiu (segmento mergeado) estão mais próximos dos atributos ideais da expectativa. Assim, em função disso, a medida de erro do novo segmento é menor do que a medida de erro que se obteve antes. Isso é diferente do que a rotulação consistente de Haralick e Shapiro aonde o erro aumenta de forma contínua. Portanto, para o cálculo da estimativa de erro futuro é necessário que se subtraia a redução do erro decorrente de fenômenos como esse quando se calcula a estimativa de erro. Isso significa que, quando se calcula a estimativa de erro, o erro que surgirá dos rótulos

ainda não atribuídos deve-se subtrair desse valor o valor da redução de erro resultante de um merging.

2.9.7.2 - Levar em conta mais de uma unidade

Para realizar o merging de segmentos é necessário que se façam modificações que se dividem em dois grupos: primeiro é necessário determinar quais são as unidades que devem ser levadas em conta no atual nível da árvore de busca. Nesse contexto procedemos da seguinte forma. Dentro das unidades vizinhas de um rótulo L entendemos todas as unidades não rotuladas que são vizinhas de uma unidade ou de várias unidades rotuladas com o rótulo L . Procedemos então da seguinte forma:

1) para todos os rótulos L já atribuídos a alguma unidade é escolhida para cada rótulo uma unidade U_xL ótima a partir das unidades vizinhas de L e procede-se da seguinte forma, o conjunto U' união de todos para onde L foi atribuído com o conjunto unitário U_xL .

2) caso U' seja vazio, ou seja, no caso em que ainda não há nenhum rótulo atribuído a alguma unidade ou no caso onde todos os rótulos atribuídos a alguma unidade não possuem mais nenhuma unidade rotulada em sua vizinhança. Assim escolhe-se uma unidade ótima a partir do conjunto das unidades sem rótulo e o conjunto U' recebe então o conjunto unitário U . A escolha de uma unidade ótima ocorre nesse processo de forma dependente de domínio. O conjunto U' recebe então todas as unidades a serem olhadas neste nível da árvore de busca, ou seja, para cada rótulo, uma unidade ótima (um conjunto que possui unidades que são candidatas a todos os rótulos diferentes dos rótulos que já foram dados).

3 - DESENVOLVIMENTO DO TRABALHO

Para a detecção das anatomias básicas (esqueleto e limites corporais) foram utilizadas técnicas simples de limiarização (thresholding), segmentação e detecção de bordas, sendo portanto, utilizadas técnicas das duas principais abordagens para a segmentação de imagens, a baseada em bordas e a baseada em regiões. Para a classificação e identificação dos segmentos recorreu-se às redes neurais. A correção da segmentação foi feita através do módulo de CLP. Após isso gera-se o polígono dos segmentos, que pode sofrer ajustes manuais. Depois os quais são unidos através de triangularização. É calculado a linha baricêntrica e planos perpendiculares a ela para a medição. O volume resultante é mensurado e exportado em formato VRML.

3.1 - Obtenção e visualização das imagens

As imagens utilizadas para este trabalho foram geradas por um tomógrafo Picker, modelo PQ5000. A conversão para o formato DICOM é realizada pelo software do tomógrafo. Para os propósitos do projeto foram utilizadas seqüência de imagens da região abdominal, cada uma armazenada em um arquivo. Tais arquivos foram importados para o banco de dados DICOM para serem visualizados através do aplicativo *Dicom Series Editor* e gerenciados pelo *Dicom Editor*.

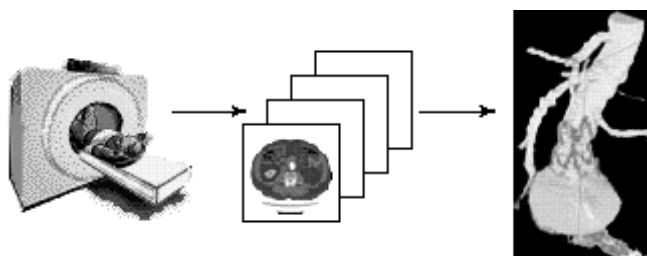


FIGURA 39: Aquisição dos cortes tomográficos

O *Dicom Editor* é um aplicativo cliente do banco de dados DICOM responsável pela visualização e gerenciamento dos dados dos pacientes e suas respectivas séries de imagens. Trata-se de um módulo desenvolvido anteriormente dentro do projeto Cyclops e que é encarregado de fazer a comunicação com o banco de dados para trazer as informações solicitadas pelo usuário. Essas informações podem ser relativas ao paciente (nome, idade, sexo, data de

nascimento, etc.), aos exames que ele realizou (data e hora do exame) e à série de imagens de um determinado exame (tipo de contraste utilizado, data de aquisição da imagem, entre outros).

Na Fig.40 é mostrado o *Dicom Editor*, o qual apresenta os pacientes e seus respectivos estudos e séries. Na janela à direita são exibidos vários dados relacionados ao paciente, estudo, série, equipamento, etc. Na janela inferior tem-se a visualização de miniaturas dos exames.

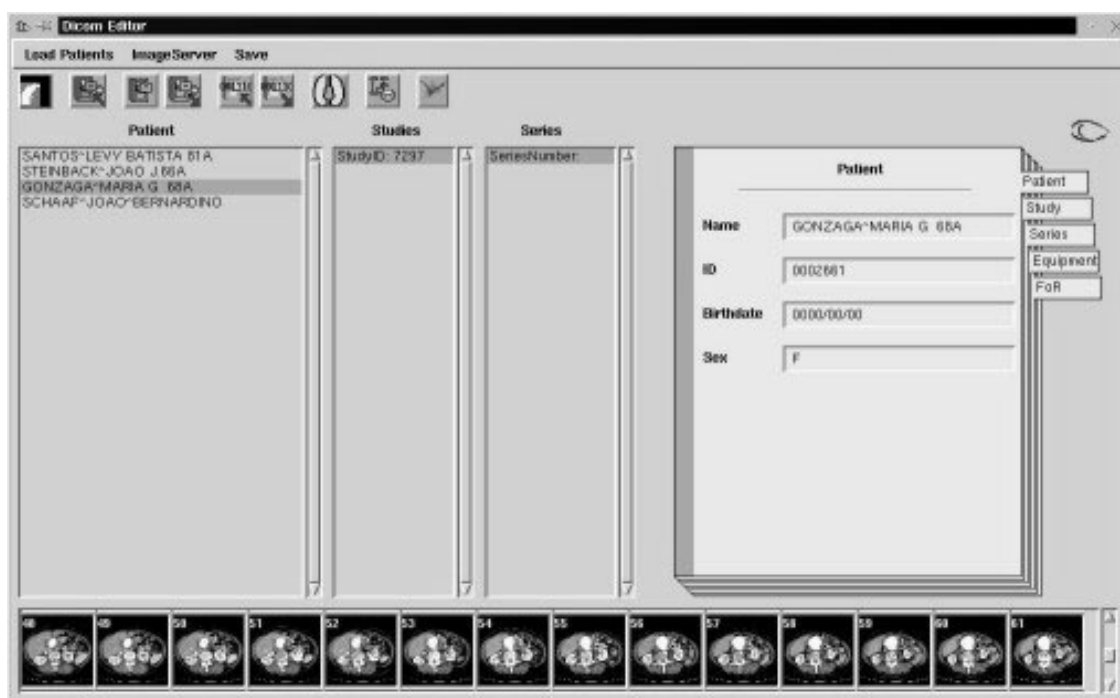


FIGURA 40: DicomEditor

3.1.1 - Visualização

O *Dicom Series Editor* é um manipulador de imagens DICOM, onde tem-se a possibilidade de observar a série de imagens e executar tarefas relacionadas à análise e ao tratamento destas imagens. As possibilidades de tratamento da imagem são bastante limitadas, restringindo-se à ampliação ou redução da faixa de tons de cinza presente na paleta de cores da imagem. Numa imagem comum de oito bits, os tons de cinza estão distribuídos ao longo das 256 posições da paleta. Pode-se reduzir esta faixa distribu-

indo-se os valores da intensidade de cinza ao longo uma faixa menor de posições, atribuindo o valor da cor preta às posições inferiores à faixa estipulada e valores de branco às posições superiores. Isto é bastante útil quando tem-se uma imagem onde o contraste entre os órgãos é pequeno. Com a redução da faixa de tons de cinza, os órgãos com uma intensidade de cinza mais clara, tornar-se-ão mais esbranquiçados e conseqüentemente, de mais fácil visualização.

Neste projeto, foi necessário ampliar a funcionalidade deste aplicativo para o sistema conseguir realizar a análise das imagens de aneurisma de aorta abdominal, ou seja, conseguir processar as imagens e produzir como resultado as dimensões da aorta na região do aneurisma e a sua reconstrução tridimensional. Para tanto, foi construída uma subclasse de *DicomSeriesEditor* (classe que implementa o *Dicom Series Editor*) denominada *AneurismAnalyser*. Esta classe pode ser definida como o cerne do sistema desenvolvido, pois é ela o responsável pela execução das principais tarefas do sistema.

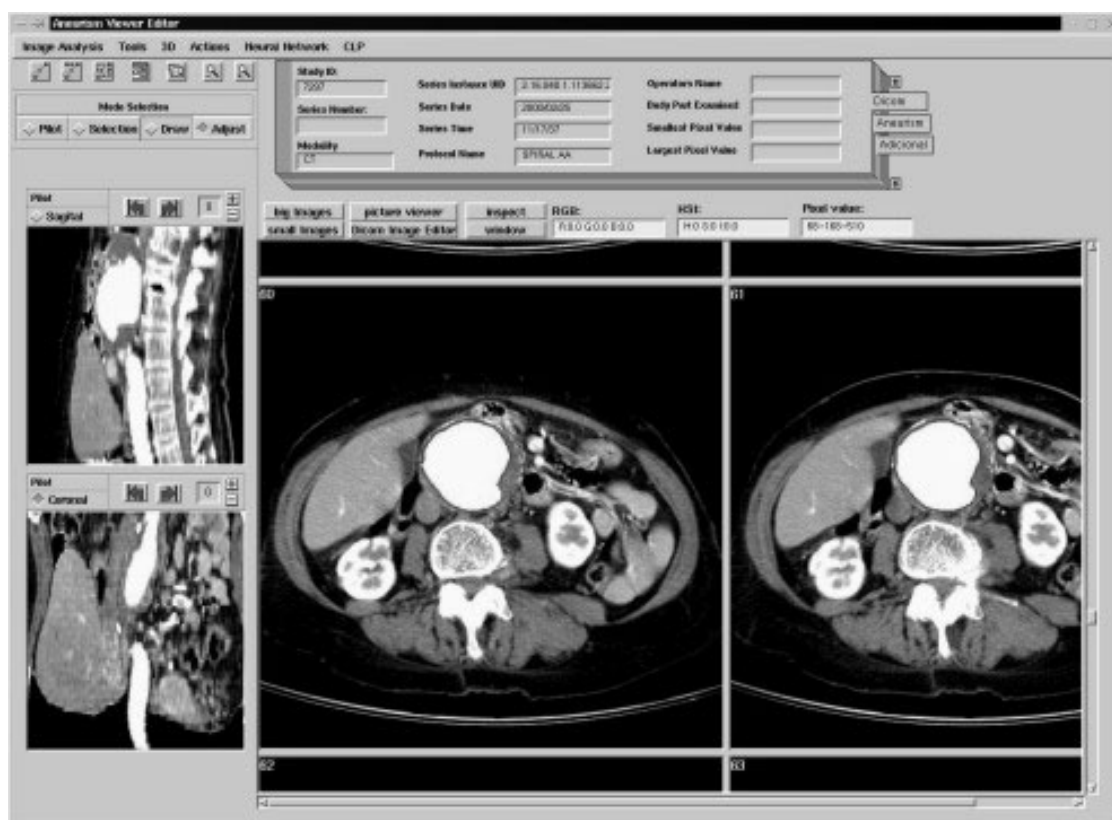


FIGURA 41: Aneurism Analyser Editor

A visualização das imagens das tomografias no *Aneurism Analyser Editor* pode ser feita de três maneiras distintas, baseadas na forma como as imagens foram carregadas do banco de dados.

Na primeira forma, as imagens são carregadas sem nenhuma alteração no seu formato ou na paleta de cores, ou seja, as imagens são de 16 bits e a paleta possui 65536 tons de cinza. As imagens apresentadas são vistas como foram obtidas.

Na segunda maneira, as imagens são carregadas como imagens de 8 bits e com uma paleta de 256 tons de cinza. Isto produz imagens onde a diferenciação visual entre os órgãos torna-se um pouco mais difícil, pois a quantidade de tons de cinza é menor que na imagem de 16 bits, o que produz uma pequena perda de informações. Porém, como o tamanho da imagem é reduzido, há uma certa economia de memória.

A terceira maneira leva em conta o *window*, isto é, as imagens são carregadas como imagens de oito bits e a paleta de cores é alterada levando em conta valores de *windowWidth* e *windowCenter*. Tais valores indicam a largura da paleta (quantidade de diferentes tons de cinza) e o seu centro na escala. Mas estes parâmetros não indicam valores na paleta e sim valores na escala *Houndsfield*. Esta escala define valores entre -1000 (corresponde ao preto) e +1000 (corresponde ao branco). Sendo assim, é necessária uma conversão entre a escala de cores de uma imagem de 16 bits e a escala *Houndsfield* para poder ocorrer a conversão para oito bits, ocorrendo uma redução menor na quantidade de informação quando comparado à redução direta de 16 para oito bits. O resultado da conversão é uma imagem onde a artéria aparece destacada dos demais órgãos devido ao líquido de contraste injetado na corrente sanguínea do paciente. Este é o objetivo dos valores de *windowWidth* e *windowCenter*.

Após ter-se as imagens dos cortes deve ser feita uma análise para a detecção da luz arterial e do tecido aneurismático. Por possuir um formato que não é muito diferenciado de uma pessoa para a outra, os parâmetros encontrados para a detecção da artéria deverão funcionar de forma satisfatória para todos os casos.

3.2 - Segmentação das imagens

A detecção da luz arterial e tecido aneurismático dentro de uma imagem de tomografia computadorizada por um software demanda uma análise da mesma para que sejam separadas as diversas formas que a compõem. Para que a artéria possa ser reconhecida, foi necessário um método que permitisse distingui-la dentre os demais órgãos. O primeiro passo para esta distinção foi um pré-processamento e posterior segmentação da imagem.

Foi adotado o procedimento de localizar primeiramente a luz arterial para depois localizar e identificar outras estruturas. Isso é justificado por dois motivos: primeiramente pelo fato de que localizada a artéria, torna-se mais fácil a localização do tecido aneurismático e estruturas como o trombo mural, pois essas localizam-se em áreas adjacentes à artéria; segundo: essas outras estruturas (que eventualmente podem nem existir ou não serem necessárias) somente são detectadas através de técnicas mais avançadas e computacionalmente mais custosas. É importante reduzir a área analisada o máximo possível para que o processamento da imagem possa ser concluído em tempo útil.

Na primeira etapa define-se um segmento de interesse (SOI) de 256x256 pixels a partir do centro da imagem. Essa área foi suficiente para todas os aneurismas analisados, mesmo nos casos onde o segmento aneurismático foi extremamente grande. A definição de um SOI tem por objetivo reduzir o tempo de processamento para valores aceitáveis.

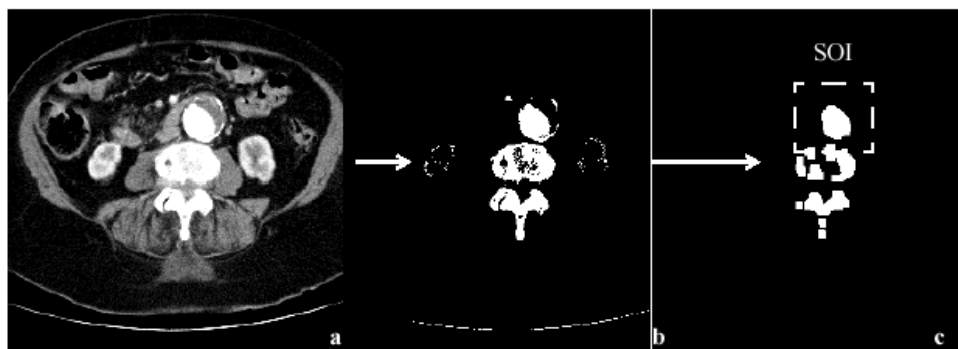


FIGURA 42: Corte original de TC (a) resultando da segmentação por limiarização (threshold) (b) e operações de morfologia matemática na imagem binária (c) mostrando o SOI após a classificação da região por uma rede neural

Após a definição do SIO procede-se com a detecção da luz da artéria utilizando-se técnicas baseada em segmentação por limiarização corrigida através da utilização de morfologia matemática.



FIGURA 43: Segmentação utilizando limiarização (thresholding)

A técnica de limiarização consiste em definir um valor limiar para a cor de um pixel. Pixels com valor de cor acima do limiar se tornam pretos e de valor abaixo se tornam brancos. Isso permite a segmentação de estruturas através do seu valor de densidade radiológica. Para eliminar os ruídos resultantes é realizada uma operação de morfologia chamada fechamento morfológico (*closing*) seguido de uma operação de abertura morfológica (*opening*). O principal tipo de ruído que essas duas aplicações de morfologia tem como objetivo eliminar, são pequenos segmentos que erroneamente ligam a artéria com outras estruturas anatômicas.



FIGURA 44: Processamento da imagem limiarizada com abertura morfológica

O passo seguinte é realizar a segmentação sobre o resultado da aplicação dos métodos de morfologia matemática do SOI. Para isso foi utilizada a técnica *watershed*. A obtenção dos valores para os parâmetros do *watershed* foi baseada na experimentação. Mais de 15 parâmetros (mostrados no Quadro 8) tiveram que ser ajustados para produzir um resultado satisfatório. Na detecção de bordas, foi modificado o valor do desvio padrão de Gauss até que fossem atingidos resultados satisfatórios. O melhor resultado obtido continha um desvio padrão (denominado *sigma* nos parâmetros passados ao operador Canny) foi 1,3. O operador Canny usado foi o *vcanny*, que faz parte do *Khoros*.

QUADRO 8: Parâmetros para o watershed

Parâmetro	Significado	Valor Utilizado
maxGrad	Gradiente máximo do voxel a ser segmentado pelo método	255
minSize	Tamanho mínimo do segmento	100
lowBackground	Menor valor do intervalo do fundo	0
highBackground	Maior valor do intervalo do fundo	0
dynSefVoxThresh	Valor adicionado ao seg-voxel threshold	1
voxSegMergeThresh	0 <= Coeficiente de não-homogeneidade <=100 100 - poucos segmento não homogêneos 0 - muitos segmentos homogêneos	20
vsMeanDiff	Utilizar a diferença do segmento médio como critério de união entre voxel-segmento	Verdadeiro
vxDistrDiff	Utilizar a diferença do tom de cinza como critério de união voxel-segmento	Falso
segSegMergeThresh	0 <= Coeficiente de homogeneidade <=1 1 - muitos segmento homogêneos 0 - poucos segmentos não homogêneos	1
ssMeanDiff	Utilizar a diferença do segmento médio como critério de união entre segmento-segmento	Falso
ssMinMaxDiff	Utilizar a diferença entre segmento mínimo e máximo como critério de união entre segmento-segmento	Falso
ssDistrDiff	Utilizar a diferença do tom de cinza como critério de união entre segmento-segmento	Verdadeiro

QUADRO 8: Parâmetros para o watershed

Parâmetro	Significado	Valor Utilizado
meanGrayOut	Gerar arquivo de saída contendo os valores do tom médio de cinza de cada segmento	Falso
watershed	Realizar a segmentação por <i>watershed</i> após o método	Verdadeiro

Essa segmentação foi realizada pelo programa *adaptWaterGrow* do *Khoros*. Após a segmentação são gerados vários arquivos, cada um contendo um único segmento. A segmentação é um processo relativamente complexo, já que nem sempre o que o programador considera como sendo o melhor resultado é o resultado que mais facilmente pode ser trabalhado pelo programa.

Encerrado o processo de segmentação, tem-se uma única imagem que contém todos os segmentos, mas ainda de difícil manipulação para o computador. Para que o computador possa localizar o aneurisma e a artéria, é necessário separar todos os segmentos. Este processo é executado pelo método *getSegments*. Este método cria um número N de imagens, (sendo N igual ao número de segmentos da imagem) cada qual contendo um fundo totalmente preto com apenas um segmento dentro dele. Ao final do processo, temos todos os objetos da imagem devidamente separados.

O tamanho das imagens com os segmentos deve ser ajustado para uma imagem de no máximo 256x256 pixels devido ao fato do operador Canny não trabalhar com imagens maiores. Também por causa deste operador a imagem não pode possuir mais de oito bits de informação de cor.

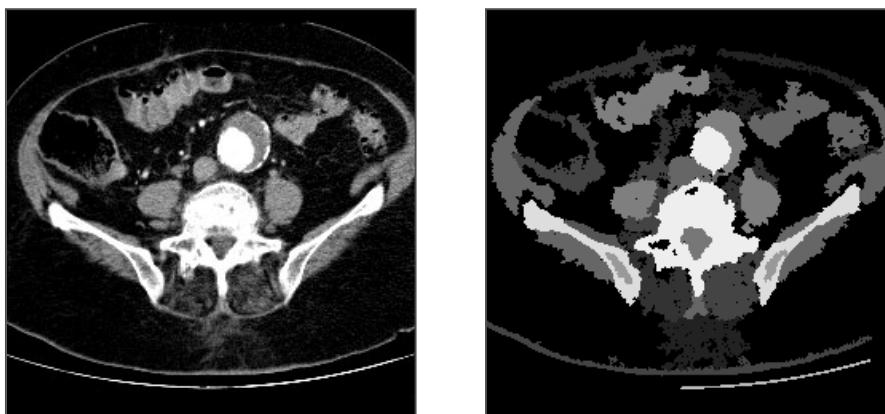


FIGURA 45: Exemplo de segmentação utilizando watershed

A conversão de cores corresponde na modificação dos valores de pixels que compõem a imagem. Estes valores são convertidos de acordo com uma regra de três simples. Uma imagem é representada por uma matriz de pixels, onde cada elemento da matriz possui um valor que representa o índice da paleta de cores da imagem. A paleta de cores é um vetor, onde cada posição contém um valor da intensidade da cor. Este valor é formado por um conjunto de três valores, um que representa a intensidade de azul, um que representa o vermelho e outro que representa o verde. Um valor de cinza é representado quando os três valores são iguais. Por exemplo, o preto é representado pelo valor 0 para todas as intensidades e o branco, pelo valor 1 para todas as intensidades. Uma imagem de 8 bits significa que cada pixel possui 8 bits (1 byte) para representação de seu índice na paleta de cores. Assim, esta imagem pode possuir no máximo 256 cores. Mas, nem todas as imagens utilizam todas as posições da paleta. Por exemplo, as imagens utilizadas para teste neste projeto eram de 12 bits, o que permitia a elas possuírem até 4096 cores, mas apenas 2441 cores eram utilizadas.

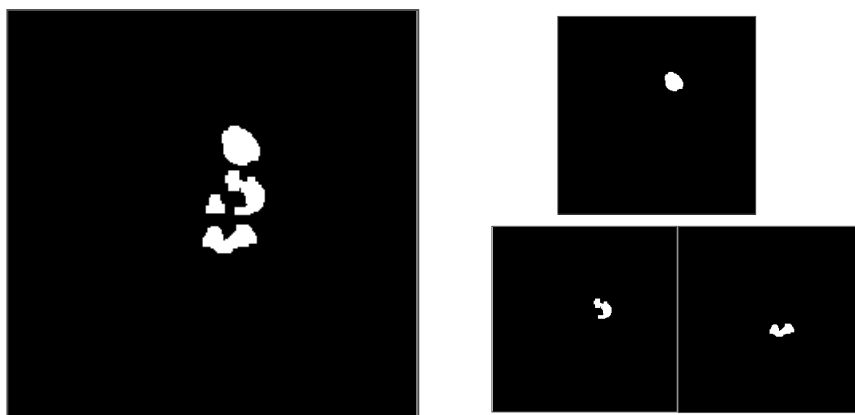


FIGURA 46: Geração dos segmentos

3.3 - Identificação dos segmentos

Os resultados dessa segmentação são submetidos a uma seleção por testes de plausibilidade anatômica. Na análise da plausibilidade anatômica todos os segmentos de um corte candidatos a serem rotulados como luz de artéria são classificados por uma rede

neural onde parâmetros como forma, diâmetro e posição são considerados. Segmentos rejeitados na classificação são descartados.

A abordagem por redes neurais foi escolhida por simplicidade e eficácia. Caso um conjunto de regras fosse utilizado, um estudo mais aprofundado teria sido necessário. Para tanto, teria que ser despendido um grande tempo de pesquisa para descobrir um conjunto de regras que funcionasse na maioria dos casos. Em relação a redes bayesianas, seria necessário recolher um conjunto de casos bastante grande. Já no caso das redes neurais, a única dificuldade foi encontrar parâmetros ou informações das imagens que sejam relevantes para a rede neural.

No que se refere a escolha da topologia utilizada, o fato de ser perfeitamente possível saber o resultado esperado de cada uma das imagens submetidas a rede neural, fez com que se escolhesse as redes diretas com o algoritmo de aprendizado *backpropagation*. Esta rede neural é uma rede simples e que permite um treinamento supervisionado, ou seja, pode-se informar a saída desejada para a rede neural.

A primeira camada possui 10 neurônios e a intermediária 20. A camada de saída possui dois neurônios. Para que a rede neural entenda os parâmetros, eles devem possuir um valor entre 0 e 1, daí o motivo para a normalização. A rede neural é um programa externo ao aplicativo, assim como os aplicativos de segmentação. A rede foi gerada e treinada com o auxílio do software SNNS¹.

Para a obtenção da resposta foi adotada uma convenção, se a rede apresentar o valor zero no primeiro neurônio e um no segundo, seria considerada a resposta como sendo negativa. Caso ocorra o contrário, considera-se a resposta como positiva. Como a rede neural nem sempre apresenta os valores inteiros como resultado, convencionou-se que o neurônio que tivesse um valor de saída maior possuiria o valor um e o outro zero. Isto é válido, já que a rede apresentou respostas com uma certeza sempre na faixa dos 95% (apresentar 0,05 no primeiro neurônio e 0,95 no segundo ou vice-versa).

O treinamento da rede neural foi feito utilizando-se 3 séries de abdômen, cada uma de um paciente diferente. De cada série foram selecionados 96 cortes, totalizando 288

1. SNNS (Stuttgart Neural Network Simulator) é um software simulador de redes neurais para estações de trabalho UNIX, desenvolvido no Institute for Parallel and Distributed High Performance Systems (IPVR) da Universidade de Stuttgart. O objetivo do projeto SNNS é criar um ambiente de simulação eficiente e flexível tanto para a pesquisa quanto para aplicações de redes neurais.

padrões de treinamento. Cada padrão possui dez valores de entrada e dois valores de saída, valores gerados pelo processo de normalização descrito anteriormente. Dos 96 padrões de cada série, 50% corresponderam à artéria e 50% a outras estruturas provenientes da segmentação, como ossos, órgãos, músculos e gordura. Estes padrões foram retirados de imagens devidamente selecionadas dentro da série. As imagens representavam desde artérias facilmente identificáveis até artérias com formato irregular, passando por casos intermediários. O programa SNNS treina a rede executando os padrões um número de vezes determinado pelo usuário. Cada vez que todos os padrões são executados é denominado um ciclo de treinamento. O usuário determina quantos ciclos são executados. O programa ao final, mostra a taxa de erro da rede. Taxas de erro menores que 5% são aceitáveis. A rede neural desenvolvida apresentou uma taxa de erro menor que 0,005% após 10000 ciclos de treinamento. Convém dizer que esta taxa de erro é calculada de acordo com os padrões apresentados, ou seja, a taxa de erro é de 0,005% para os 288 padrões de treinamento, enquanto a taxa superior a 95% de certeza é relativa a todas as imagens apresentadas para a rede neural.

Os parâmetros de entrada para a rede neural são mostrados no quadro abaixo.

QUADRO 9: Parâmetros de entrada para a rede neural

Parâmetro	Significado	Valor Utilizado
Área	Área da imagem	0-1000
Perímetro	Perímetro da imagem	0-2000
Tom médio de cinza	O valor médio do tom de cinza que aparece na imagem	0-255
Circularidade	Coeficiente adotado como sendo a razão entre o cálculo do perímetro pelo raio obtido através do cálculo da área. A fórmula para esse cálculo é $\frac{P}{2\sqrt{a\pi}} \quad (20)$	-
Tom mínimo de cinza	O menor valor de tom de cinza que aparece na imagem	0-255
Tom máximo de cinza	O maior valor de tom de cinza que aparece na imagem	0-255
Mediana	O valor da mediana da imagem	0-255

QUADRO 9: Parâmetros de entrada para a rede neural

Parâmetro	Significado	Valor Utilizado
Desvio padrão dos tons de cinza	Variância dos tons de cinza	5-16
Variância dos tons de cinza	Variância dos tons de cinza	0-255
Posição	Posição do segmento da imagem como um todo	0 - Posição do último pixel do segmento

A execução da rede também acontece com um aplicativo anexo ao SNNS, o *batch-man*. Ele executa uma rede neural armazenada em um arquivo texto e através de padrões de entrada armazenados em outro arquivo, gera um arquivo de saída contendo a resposta da rede para as entradas apresentadas.

Após realizar a normalização dos dados, o software guarda quais são as imagens que passaram pela normalização e irão para o teste na rede. Executada a rede neural, o software verifica qual das imagens foi classificada como sendo a imagem da artéria e a envia como resposta.

Todos os processos acima descritos são executados pelos métodos *getSegmentsToNeuralNetwork*, *createBatFile*, *createInputFile*, *loadResultsFromNeuralNetworks*, *executeNeuralNetwork*, *getAneurismImages* e *normalizeValue*.

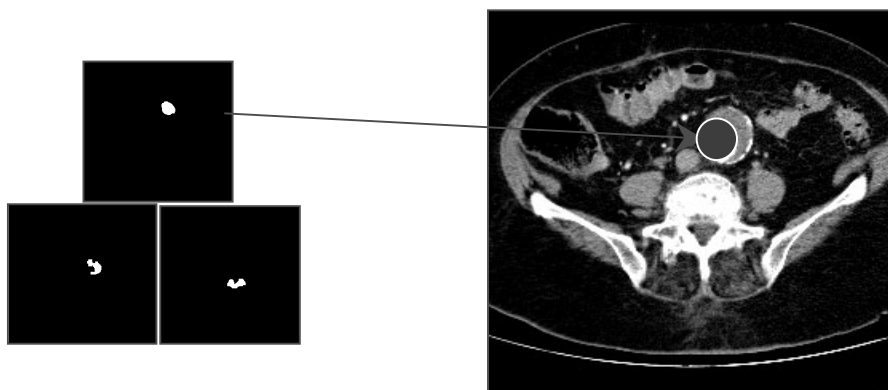


FIGURA 47: Identificação de um segmento pela rede neural como a luz da artéria

Outro problema que ocorreu foi com relação à identificação de dois segmentos do mesmo corte como luz arterial. Isso acontece devido à presença de sacos aneurismáticos,

onde existe uma luz verdadeira e outra segmento correspondente ao saco aneurismático. Nesse caso a rede neural retorna um grau de certeza menor para a luz falsa. É necessário a diferenciação entre as duas "luzes" pelo fato de uma delas (a verdadeira) continuar nos próximos cortes e a falsa terminar abruptamente em algum corte abaixo, nesse caso a reconstrução deverá fechá-lo. Os cálculos para a geração das medidas devem ser realizados sobre a luz verdadeira.

3.4 - Detecção do tecido aneurismático

A detecção do tecido aneurismático é mais trabalhosa, visto que o trombo mural apresenta densidades radiológicas semelhantes às de outras anatomias, as quais às vezes tocam o aneurisma, formando um só segmento.

Com o segmento correspondente à luz arterial já previamente identificado define-se agora um segmento de interesse de 128x128 pixels ao redor do centro do segmento arterial. Essa região é suficiente para envolver toda a artéria, incluindo a luz arterial, trombo mural, parede arterial e circulação periférica. A definição de uma área tão pequena é necessária pelo fato da segmentação por Mumford&Shah ser um processo de altíssimo custo computacional e que apresenta um tempo de execução exponencial com relação à área da imagem a ser segmentada.

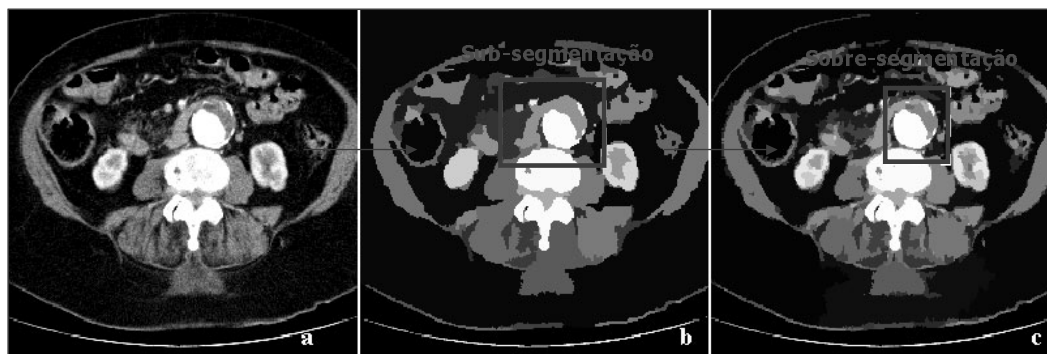


FIGURA 48: Corte original de uma TC da aorta (a) e duas diferentes segmentações realizadas com Mumford&Shah: (b) mostra um "vazamento" na segmentação no tecido aneurismático; (c) não apresenta "vazamento" mas divide o aneurisma em diferentes segmentos

A detecção do tecido aneurismático é semelhante à luz arterial, diferindo na forma de segmentação, a qual ocorre em dois estágios. Um estágio inicial é realizado através de

limiarização corrigida e um segundo estágio é realizado através de uma segmentação posterior da regiões resultantes através do crescimento de regiões com divisor de água. Utiliza-se o algoritmo de Mumford&Shah com parâmetros mais sensíveis o que acaba causando uma sobresegmentação. Foi escolhido esse método para evitar o "vazamento" de segmentos para estruturas vizinhas com valores de HU (*Hounsfield Units* - unidades radiológicas) similares aos do tecido aneurismático, fornecendo portanto uma segmentação estável.

Os segmentos são pré-classificados por uma rede neural e em seguida submetidos à análise de plausibilidade anatômica em um módulo especialmente desenvolvido baseado em uma extensão do método de rotulação consistente inexato (extCLP) já explicado. Foi construída uma rede neural específica para essa função, similar à utilizada para detectar a luz arterial, inclusive com os mesmos parâmetros para os neurônios da camada de entrada. A diferença é o conjunto de treinamento. O número de padrões de treinamento foi semelhante ao da rede anterior.

3.5 - Correção da segmentação através do módulo extCLP

Como a segmentação foi realizada com parâmetros super-sensíveis, o tecido aneurismático acaba sendo dividido em vários pedaços que necessitam ser unidos em um só segmento.

O módulo extCLP funde os diferentes segmentos pertencentes ao tecido aneurismático em um só e descarta os outros tecidos. O método extCLP atua realizando uma checagem de consistência, apontando segmentações erradas e habilitando uma reparametrização do algoritmo de segmentação.

Na Fig.49 temos o resultado da segmentação com os grupos de segmentos pertencentes à mesma estruturas já unidos. Em Fig.49a é mostrada a luz arterial, em Fig.49b é exibido o trombo mural e em Fig.49c tem-se tecido adjacente, o qual normalmente é fundido com o trombo mural.

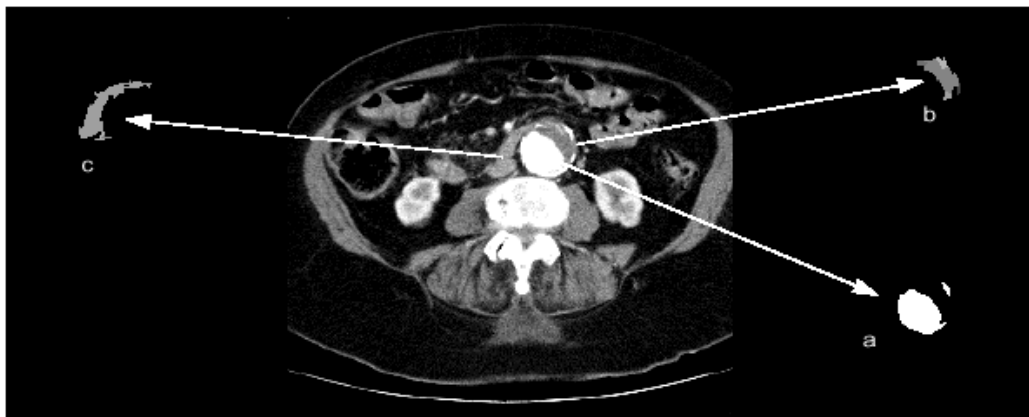


FIGURA 49: Imagem original (centro) e alguns segmentos divididos em três grupos: (a) representa a luz da artéria e circulação periférica, (b) representa o grupo de segmentos resultantes da sobresegmentação do tecido aneurismático que teve que ter sido realizada com o objetivo de evitar que os mesmos "grudassem" com o tecido adjacente (c) durante a segmentação

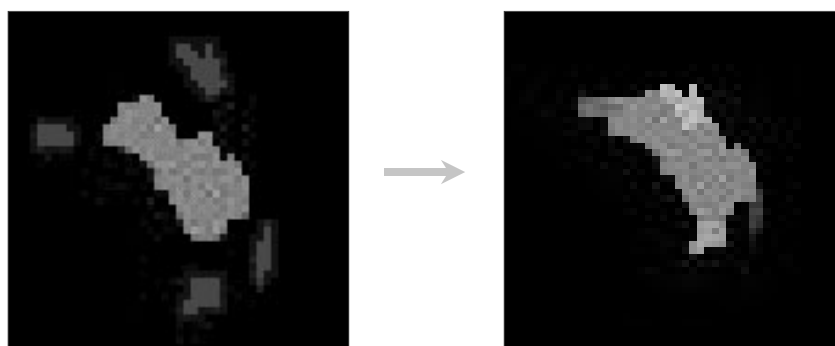


FIGURA 50: Os segmentos identificados como sendo parte do tecido aneurismático, os quais são fundidos num único segmento

3.6 - Detecção de bordas

A rede neural retorna uma lista de imagens classificadas como sendo os segmentos correspondentes à artéria existente em cada uma das imagens da série da tomografia computadorizada. Mas é necessário que se leiam os pontos das bordas da imagem, para que se possa calcular onde fica o seu centro e para se proceder com a reconstrução do volume em 3D.

A imagem possui um fundo totalmente preto e o segmento possui um tom de cinza que foi gerado no pré-processamento para a rede neural. Para armazenamento em memó-

ria, a imagem corresponde a uma matriz de pixels, como já foi mencionado. Assim, a imagem possuirá uma parte dos pixels com valor 0, correspondendo ao fundo, e outros pixels com valor diferente de 0, correspondendo ao tom de cinza da imagem. Mesmo assim, para facilitar a detecção da borda, a imagem é transformada em uma imagem preto e branco. Esta imagem em preto e branco é enviada para um programa do *Khoros*, o *vdrf*, um filtro ótimo para detecção de bordas de uma imagem. Para o *vdrf* é passado um arquivo de entrada que deve estar no formato VIFF, formato padrão das aplicações *Khoros*. Retorna também um arquivo neste formato com as bordas da imagem.



FIGURA 51: Detecção de bordas utilizando o filtro VDRF

3.7 - Geração do polígono

Para o cálculo do polígono que corresponde a artéria (sabe-se que, apesar da artéria ter um formato próximo ao circular, os computadores devido a suas características, não podem trabalhar com formatos contínuos; assim, eles transformam círculos e curvas em polígonos com grande número de lados, para que o formato fique o mais próximo do circular), o software calcula quais são os vértices do polígono. O ponto relativo a cada vértice do polígono é igual a posição do pixel na imagem. Convém lembrar que todos os pixels são pontos em duas dimensões.

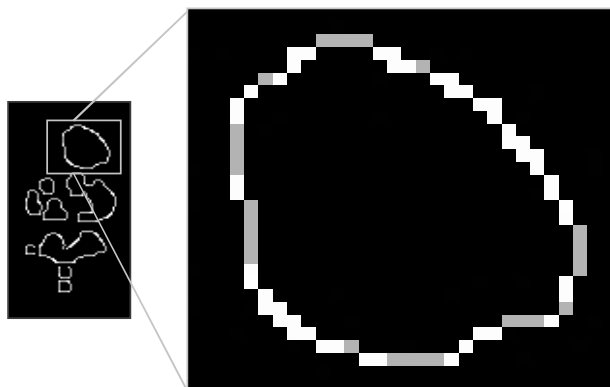


FIGURA 52: Geração do polígono ao redor da borda

Após a obtenção do polígono em duas dimensões, é executado um método para gerar os polígonos em três dimensões. A terceira coordenada é determinada de acordo com a posição da imagem no conjunto, ou seja, pela ordem da fatia na série. A distância entre uma fatia e outra é obtida da própria imagem DICOM, já que este é um dado que faz parte da estrutura da imagem.

Devido a disposição dos eixos em VRML, é necessária uma pequena modificação na estrutura do polígono. Todas as coordenadas do eixo z são substituídas pelas coordenadas do eixo y e vice-versa. Isto ocorre para que a imagem possua uma visão lateral. Se não fosse realizada esta modificação, a visão da imagem seria dentro da parte interna da artéria, dificultando a visualização da mesma. O software realiza os processos acima para toda a lista de imagens de artéria que foram devolvidas pela rede neural. Ao final, o software possui uma lista de polígonos que formam a artéria.

Os métodos responsáveis pela execução destes processos são: *createPolygons*, *getEdgePoints* e *modifyZbyY*, além do método de inicialização da classe *Polygon3D*.

Outra possibilidade é a escolha do nível de "suavidade" do polígono, ou seja, do número de pontos da imagem original que formarão o polígono. Um polígono com mais pontos produzirá uma reconstrução com uma qualidade sensivelmente maior. O ponto negativo é que quanto mais detalhado for um polígono maior será o tempo necessário para processamento, maior será o arquivo VRML gerado e também mais lenta será a manipulação desse arquivo no browser.

3.8 - Correção manual do polígono

Existem casos em que não é possível a correta segmentação da luz arterial ou do tecido aneurismático, pois os mesmos praticamente se fundem com outras estruturas adjacentes tais como outras artérias, veias, etc. Nesses casos é possível a edição manual do polígono gerado pela segmentação. Pode-se criar mais pontos, alterar a sua posição ou unir dois pontos eliminando-se os pontos intermediários. Existe também a possibilidade de se copiar o polígono gerado por outro corte, caso a segmentação ou a identificação pela rede neural venham a falhar totalmente.

Quando um ponto é arrastado os pontos pertos do mesmo também o seguem. O grau de movimentação e o número de pontos é controlado através de uma função gaussiana, onde o parâmetro sigma, que controla o "chapéu" de Gauss pode ser alterado.

No exemplo na Fig.53 tem-se uma segmentação defeituosa, a luz arterial é fundida com outra estrutura. Como não foi possível encontrar um método de segmentação que separasse corretamente o segmento pode-se editar manualmente o polígono resultando na Fig.54.

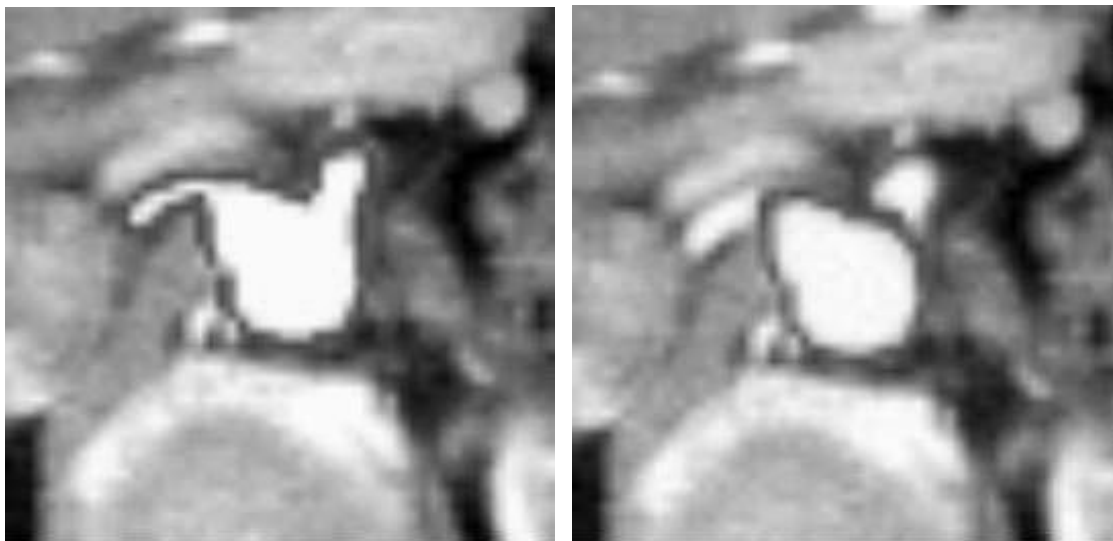


FIGURA 53: Exemplo mostrando uma segmentação defeituosa e posterior correção

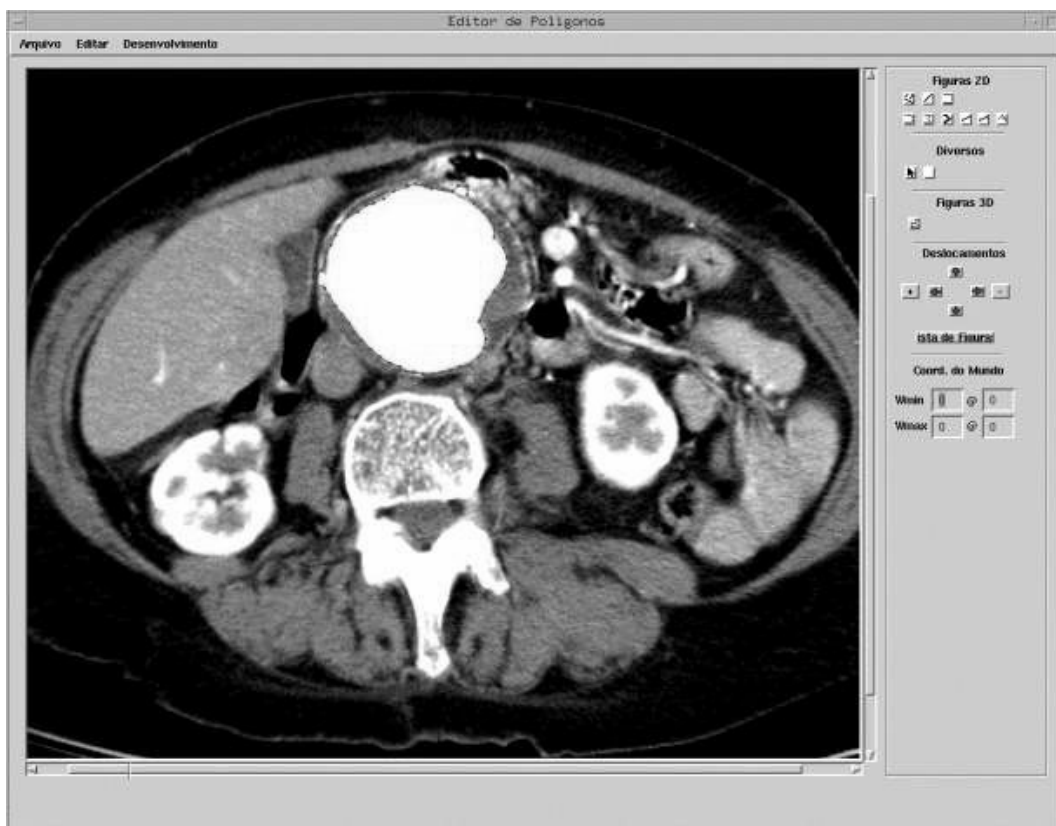


FIGURA 54: Editor para correção do polígono

3.9 - Mensuração da artéria

A principal finalidade do sistema é a mensuração do aneurisma, ou seja, através da imagem tridimensional fornecer ao usuário do programa as dimensões precisas e a localização do aneurisma.

Para conseguir com precisão as dimensões do aneurisma e da aorta não basta observar a imagem bidimensional e estimar o comprimento baseando-se na distância entre um corte e outro da tomografia, nem simplesmente calcular o diâmetro baseando-se nas bordas encontradas. Tais cálculos trariam um erro considerável, visto que a aorta descreve uma trajetória irregular, com deslocamentos simultâneos nos eixos x, y e z. Assim, se a artéria deslocar-se numa direção diagonal, a tomografia mostrará uma modificação no diâmetro da artéria que inexistente na realidade. O comprimento da aorta também é alterado, já que a distância na direção diagonal é maior que a distância vertical entre os cortes. Isto pode ser facilmente demonstrado pelo teorema de Pitágoras.

Um artifício que pode ser utilizado para que a mensuração não possua erros é construir a linha baricêntrica da artéria e utilizá-la para representar a mesma. Assim, medindo a linha tem-se o tamanho da artéria. Para a construção da linha, deve-se calcular o centro de massa de todos os cortes da tomografia. Centro de massa é definido como o ponto que representa a posição principal da matéria em um corpo. Considerando-se a artéria como uma estrutura maciça e homogênea, ele pode ser determinado utilizando-se os pontos obtidos para a determinação da borda.

3.9.1 - Localização do centro do segmento

Calculados os pontos que formam a borda da artéria, tem-se condições de determinar o centro de massa de cada segmento que forma a artéria. Isto é fundamental para a obtenção da extensão da artéria pois é através do cálculo da distância entre os centros que se consegue um valor para o comprimento com relativa precisão.

O problema de encontrar o centro de um objeto irregular não é simples. Para se resolver este problema está se utilizando o seguinte método:

- 1) Encontra-se o centro de gravidade (centróide).
- 2) Encontra-se a orientação do segmento.
- 3) Caso o centro de gravidade localize-se fora do segmento o algoritmo realiza uma busca pelo segmento partindo do centro de gravidade em uma direção perpendicular à orientação. Se mesmo assim não for encontrado o segmento repete-se a busca a partir do centro e com a direção fornecida pela orientação da figura. Se forem encontrados vários segmentos a partir dessa busca será considerado o maior segmento encontrado.
- 4) Caso o centro localize-se dentro do segmento é gerado uma linha que, a partir do centro de gravidade e da orientação, alcance as extremidades do segmento.
- 5) A partir do centro desta linha gerada no passo (3) ou (4) é gerada uma linha perpendicular que alcance as extremidades da figura. Caso a linha passe por vários segmentos será considerado o maior encontrado.
- 6) O centro desta última linha será o centro do segmento.

As coordenadas do centro de massa de um objeto podem ser obtidas pelas três integrais abaixo:

$$x_{cm} = \frac{1}{M} \int x \, dm \quad (21)$$

$$y_{cm} = \frac{1}{M} \int y \, dm \quad (22)$$

$$z_{cm} = \frac{1}{M} \int z \, dm \quad (23)$$

Em princípio, as integrais devem ser calculadas para todos os elementos de massa do objeto. Na prática, porém, existe uma forma mais simples de calculá-las. Se o objeto tem uma densidade (massa por unidade de volume ou massa específica) uniforme, pode-se escrever:

$$\frac{dm}{dV} = \frac{M}{V} \quad (24)$$

onde dV é o volume ocupado por um elemento de massa e V é o volume total do objeto. Substituindo dm , dado pela Eq.x, na Eq.y, temos:

$$x_{cm} = \frac{1}{V} \int x \, dV \quad (25)$$

$$y_{cm} = \frac{1}{V} \int y \, dV \quad (26)$$

$$z_{cm} = \frac{1}{V} \int z \, dV \quad (27)$$

Muitos objetos possuem um ponto, eixo ou plano de simetria. Nesse caso, o centro de massa do objeto está sobre esse ponto, eixo ou plano. Por exemplo: o centro de massa de uma esfera homogênea (que possui um centro de simetria) fica no centro da esfera. O centro de massa de um cone homogêneo fica no meio do cone. Mas o centro de massa não está necessariamente localizado no interior do objeto. Esse é o caso de objetos como anéis ou em forma de ferradura.

Para o cálculo do centro de gravidade existem dois métodos: *schwerpunktSilhouette* e *schwerpunktGrauwert*, já desenvolvidos anteriormente neste projeto. O primeiro baseia-se somente na posição dos pixels não pretos. O segundo utiliza o tom de cinza de

cada pixel como uma "massa" extra para cada pixel, resultando num centro de gravidade que reflete a distribuição de brilho dos pixels de um segmento.

Os métodos *haupttraegheitsAchsenGrauwert* e *haupttraegheitsAchsenSilhouette* calculam a versão bidimensional do centro de gravidade: o eixo principal de inércia, o qual fornece a linha principal ao redor da qual a "massa" de uma segmento de imagem está distribuída. Isso é utilizado para determinar a orientação de um segmento alongado. Esse eixo é sempre passa através do centro de gravidade.

Na Fig.55 a linha vermelha corresponde ao passo (3-4) do algoritmo. A linha azul corresponde ao passo (6) e o resultado final é o centro do segmento representado pelo círculo em verde.

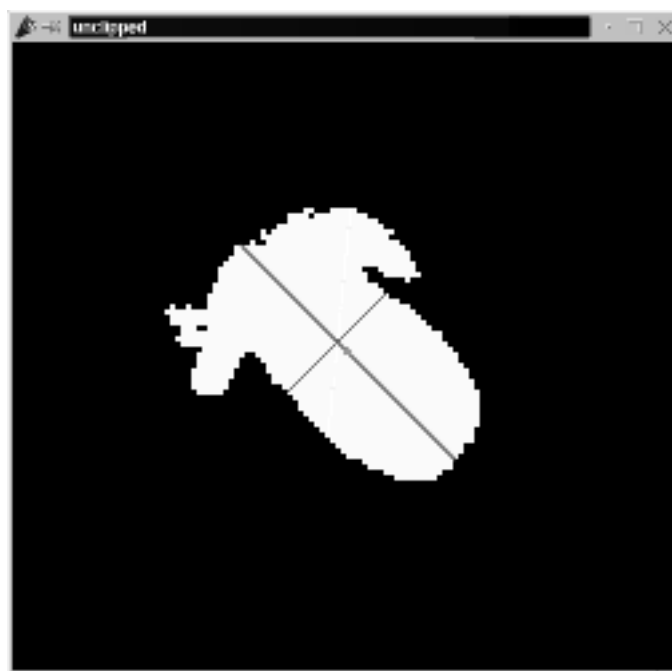


FIGURA 55: Determinação do centro do segmento

3.9.2 - Linha baricêntrica

Localizado o centro da artéria em cada uma das imagens da série, a mensuração é um processo simples. Para ter a distância entre dois pontos, basta traçar uma reta entre estes pontos e verificar o tamanho dela, ou seja, é calculada a distância euclidiana entre os centros de dois polígonos de cortes adjacentes. Como os pontos possuem coordenadas

em três dimensões, a reta desloca-se nos três eixos, dando a medida exata entre os dois pontos.

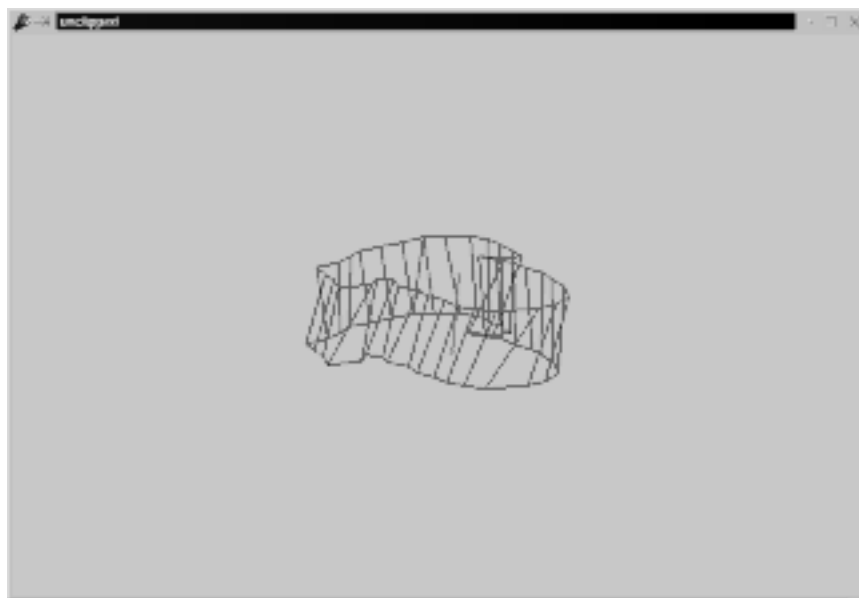


FIGURA 56: Junção dos centros em cada um dos cortes

O cálculo do tamanho da reta é simples. O ponto da imagem superior é projetado sobre a imagem inferior, gerando um terceiro ponto. Este terceiro ponto terá uma das coordenadas igual a dos centros, ou seja, terá o mesmo valor da coordenada z do centro da imagem superior e o mesmo valor da coordenada y do centro da imagem inferior. Assim, calcula-se a distância deste ponto ao centro da imagem superior, gerando uma reta, e ao centro da imagem inferior, gerando outra reta. Estas duas retas possuem um ângulo de 90° . Logo a distância entre os centros é a reta que completa este triângulo. O tamanho desta reta é calculado pelo teorema de Pitágoras ($a^2 = b^2 + c^2$). Na Fig.57 vê-se uma ilustração do que foi descrito.

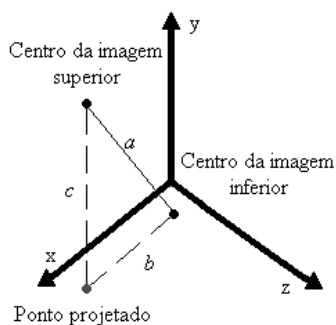


FIGURA 57: Representação do cálculo do centro de massa

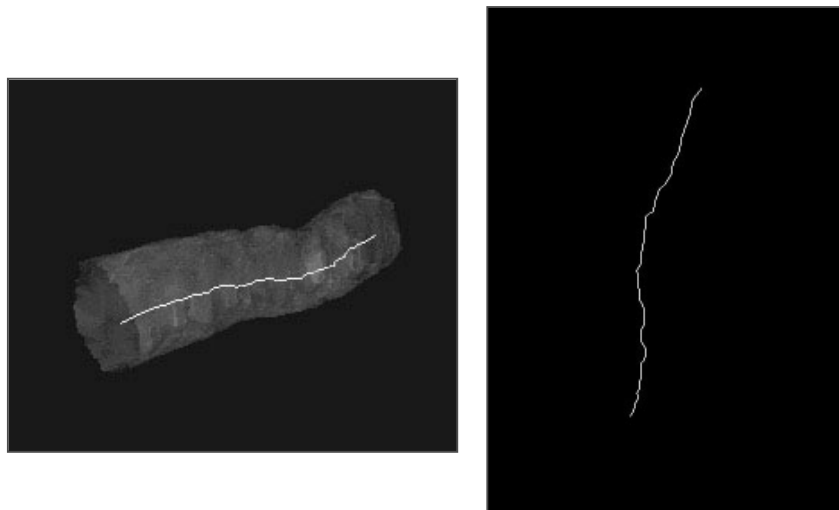


FIGURA 58: Geração da linha baricêntrica

3.9.3 - Geração dos planos perpendiculares

Para uma correta mensuração do diâmetro dos segmentos arteriais deve-se realizar a mensuração a partir de planos perpendiculares à orientação da artéria.

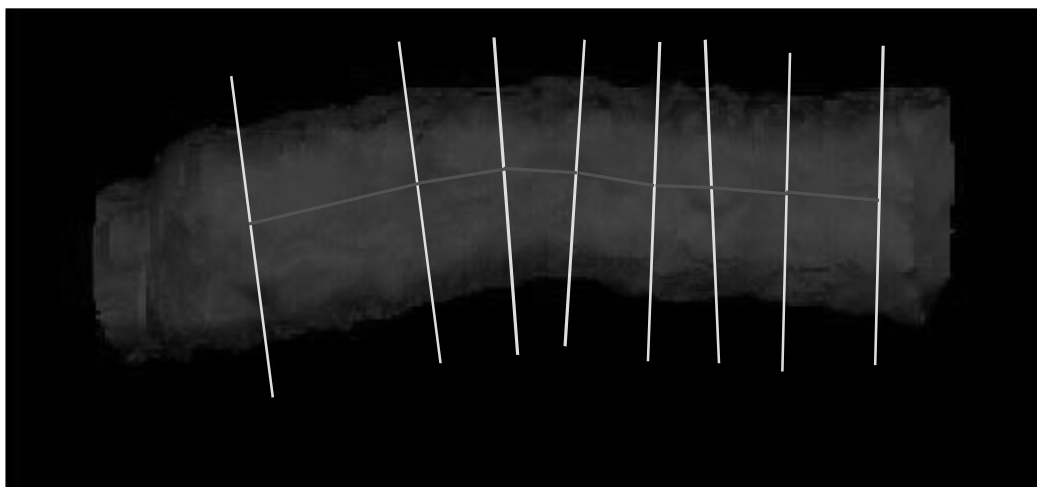


FIGURA 59: Planos perpendiculares

Através da definição de planos perpendiculares é possível gerar dados para a reconstrução exata e real da artéria. Essa reconstrução é feita a partir da intersecção dos

triângulos gerados entre dois cortes adjacentes e o plano perpendicular à orientação desses dois cortes.

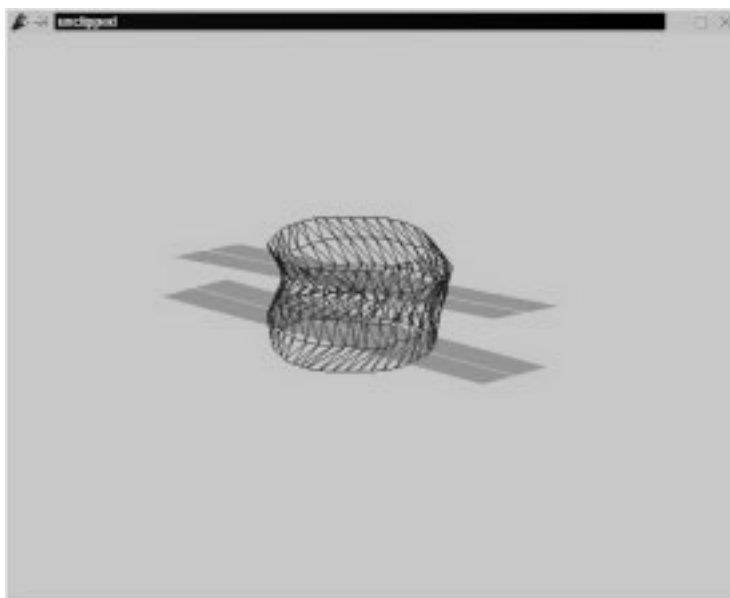


FIGURA 60: Geração dos planos perpendiculares à orientação da figura

3.9.4 - Geração do "piloto artificial"

O volume tomográfico é apresentado em cortes no sentido axial usando as coordenadas xy , tomando o valor de z fixo (posição onde corta cada slice do volume). Para ter condições de visualizar o volume em 3D é necessária a existência de cortes no sentido sagital e coronal, quando o volume não possui o piloto (no caso das tomografias) é necessário gerar um piloto artificial.

O piloto sagital corta todo o volume no sentido yz , tomando o valor de x fixo (posição onde o corte é realizado). Para representação na interface que é 2D, onde x é ignorado e o valor de y é tomado como x enquanto z toma o valor de y . A Fig.61 mostra o resultado obtido com o algoritmo.

No piloto coronal o corte é feito no volume pelo sentido xz , tomando desta vez o valor de y como fixo (corte do piloto). Para a representação na interface, y é ignorado e o valor de x permanece e z toma o valor de y . Na Fig.62 é possível observar o resultado.



FIGURA 61: piloto sagital do volume tomográfico



FIGURA 62: piloto coronal do volume tomográfico

3.9.5 - Informações necessárias para o projeto da prótese

Para que se possa projetar e construir uma prótese que se adeque às características anatômicas do paciente é necessário a obtenção de determinadas medidas da artéria aorta.

Para próteses torácicas são necessárias as dimensões mostradas na Fig.63. São elas:

- D1 - Diâmetro superior do colo proximal
- D2 - Diâmetro inferior do colo proximal
- D3 - Maior diâmetro do aneurisma
- D4 - Diâmetro do colo distal
- L1 - Extensão do colo proximal
- L2 - Extensão do aneurisma
- L3 - Extensão do colo Distal

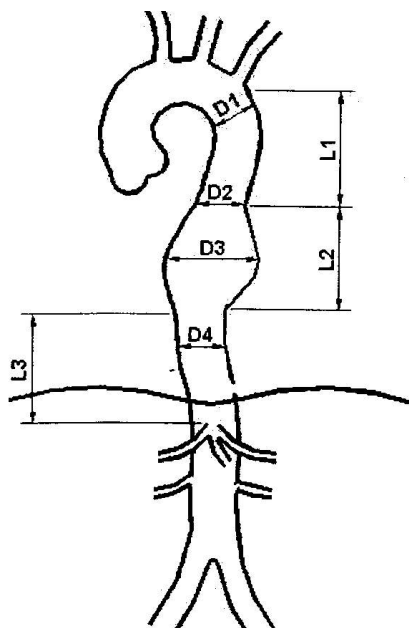


FIGURA 63: Dimensões do aneurisma para endopróteses torácicas

Para próteses aorto bi-ilíacas são necessárias as dimensões mostradas na Fig.64.

Essas medidas são mostradas a seguir:

D1 - Diâmetro superior do colo proximal

D2 - Diâmetro inferior do colo proximal

D3 - Maior diâmetro do aneurisma

D4 - Diâmetro do colo distal (se existente)

L1 - Extensão do colo proximal

L2 - Extensão do Aneurisma

L3 - Extensão do colo Distal (se existente)

D5/D6 - Diâmetro da íliaca comum - terço médio

D7/D8 - Diâmetro da íliaca comum - terço distal

L5/L6 - Extensão da bifurcação à hipogástrica

D9/D10 - Diâmetro da íliaca externa (caso o aneurisma atinja a artéria hipogástrica)

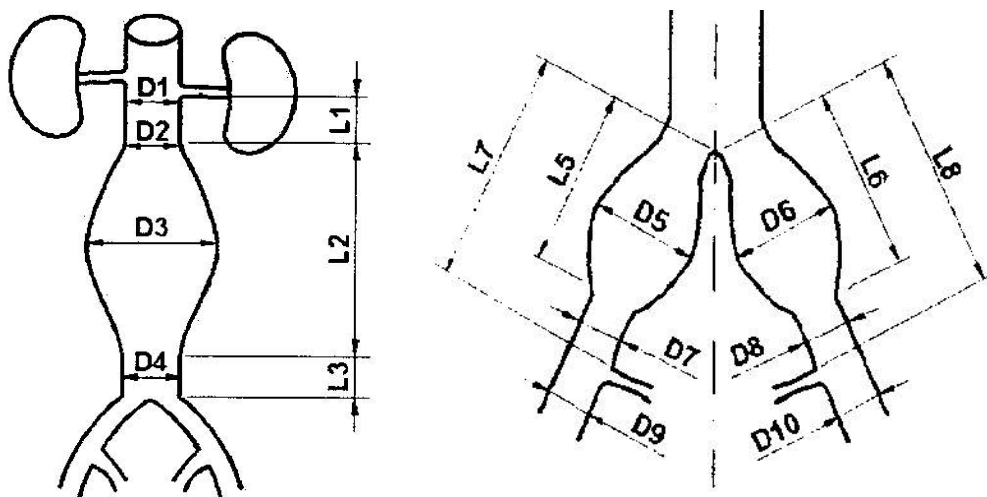


FIGURA 64: Dimensões do aneurisma para endopróteses aorto bi-iliaca

3.9.6 - Mensuração através do piloto

Além do piloto ser proporcionado para uma melhor visualização e compreensão do volume tomográfico ele também é utilizado para duas outras operações.

- 1) Escolha da faixa de cortes a ser processada para a detecção da artéria;
- 2) Determinação dos cortes a serem utilizados para a mensuração e obtenção dos dados necessários para o projeto da prótese.

A Fig.65 exibe o módulo de mensuração ativo. Nele são mostradas várias informações, na tela principal, onde são mostrados os cortes axiais é exibido o diâmetro do segmento arterial (Fig.66). Nos pilotos (Fig.67) pode-se ver a faixa de cortes selecionados na janela de mensuração (Fig.68). A determinação da faixa pode ser feita tanto através desta última tela quanto através da operação de arrastar-e-soltar nas janelas dos pilotos artificiais.

O banco de dados utilizado para armazenar os dados de mensuração é o MySQL, podendo o usuário acrescentar mais itens de mensuração quando isso for necessário. Entre os dados armazenados está também a coordenada do piloto no instante em que foi realizada a mensuração.

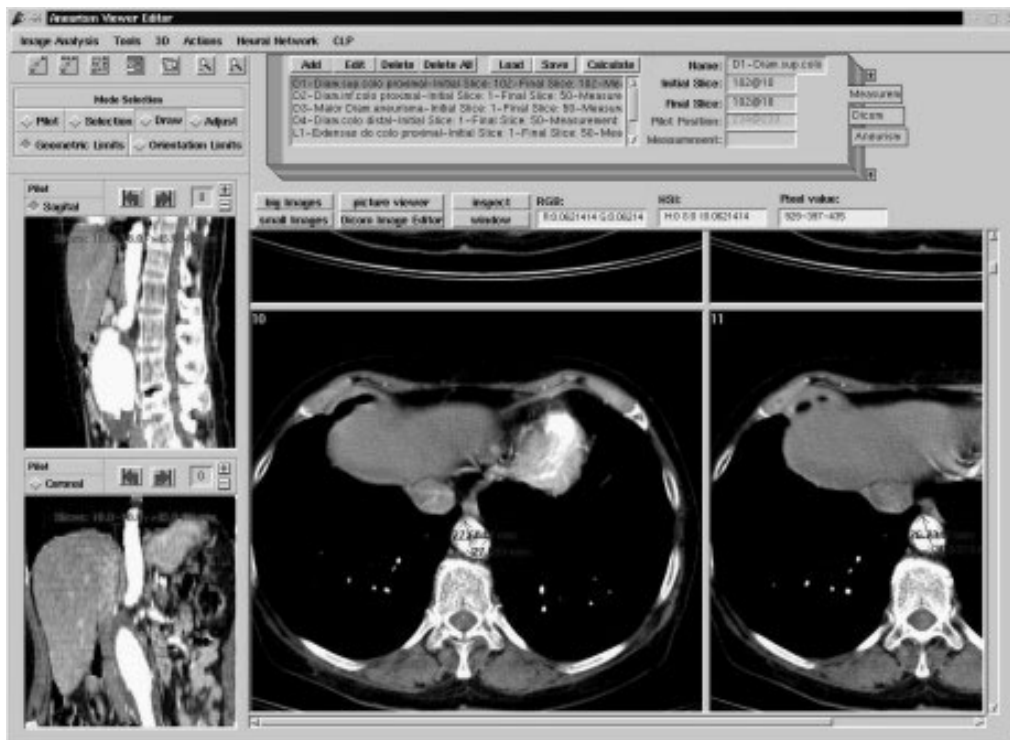


FIGURA 65: Modo de mensuração



FIGURA 66: Detalhe do módulo de visualização mostrando o diâmetro do segmento arterial

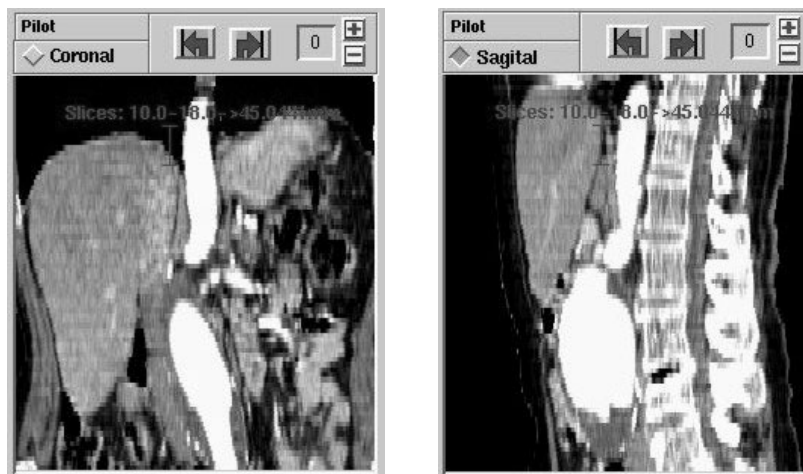


FIGURA 67: Mensuração através do piloto artificial

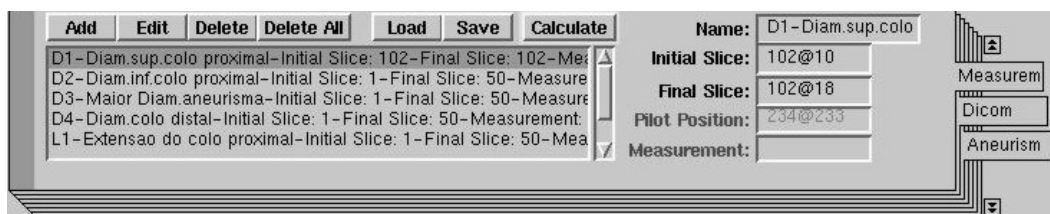


FIGURA 68: Definição dos itens a serem mensurados

3.10 - Reconstrução tridimensional da artéria

Com a série de polígonos dá-se início a fase de triangularização. Esta fase é de suma importância para a reconstrução pois esses triângulos receberão faces no momento que a reconstrução for mostrada ao usuário, e, se uma dessas faces não estiver correta, a imagem como um todo ficará prejudicada ocorrendo uma distorção do que seria uma cópia fiel de um paciente e/ou seus achados.

Como já mencionado anteriormente, o objetivo da reconstrução tridimensional é proporcionar ao usuário uma visão completa da artéria. Por ser mais tortuosa na região abdominal, a artéria não é vista corretamente através de uma imagem bidimensional. Uma visão tridimensional dela é uma boa alternativa para o usuário poder ter uma noção mais precisa de sua trajetória.

É possível a visualização da reconstrução através de um visualizador VRML ou na própria tela do software. Nesse último caso pode-se informar o nível de zoom e também alterar o ângulo de visualização e a localização da reconstrução. A visualização com texturas e possibilidade de "navegação" em tempo real só são possíveis através de um visualizador VRML.

3.10.1 - Processo de triangularização

Para a representação da artéria em três dimensões, foram utilizados algoritmos de triangularização para unir os pontos detectados na etapa anterior. Tais algoritmos são a forma mais conhecida de criação de imagens em três dimensões, partindo de listas de pontos que representam os vértices da imagem desejada.

Problemas ocorrem quando as bordas são formadas por um número diferente de pontos. Também quando as bordas possuem segmentos de reta muito longos. Isto provoca triângulos que fogem da uniformidade desejada. Para solucionar estes problemas, os segmentos de reta mais longos foram divididos em dois ou mais segmentos até que se atinja o mesmo número de pontos em ambas as bordas. Tal solução apresentou bons resultados, proporcionando triângulos de tamanho e distribuição uniforme.

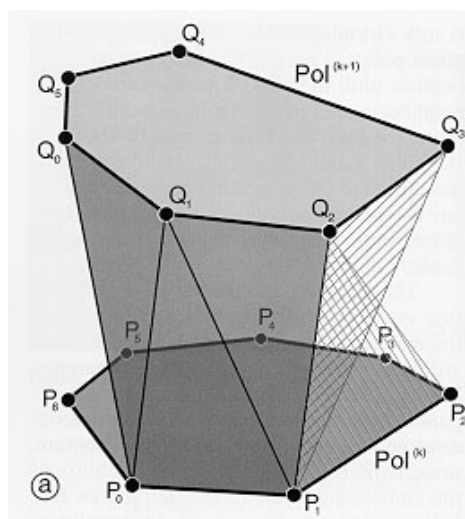


FIGURA 69: Triangularização entre os cortes

3.10.2 - Visualização da reconstrução

Neste software são proporcionadas duas opções ao usuário para a geração da reconstrução da artéria. Na primeira, as bordas da artéria possuem textura sólida, na outra elas são desenhadas em forma de grade de arame. Em ambas as forma de visualização, a área abrangida pelo tecido aneurismático é diferenciada do resto da artéria através da mudança de cor. Para tanto, foi implementada uma estrutura de métodos responsáveis pela exportação de imagens para o VRML. Ela é formada por métodos que fazem a união dos pontos das bordas das imagens e a representação dos mesmos em linguagem VRML.

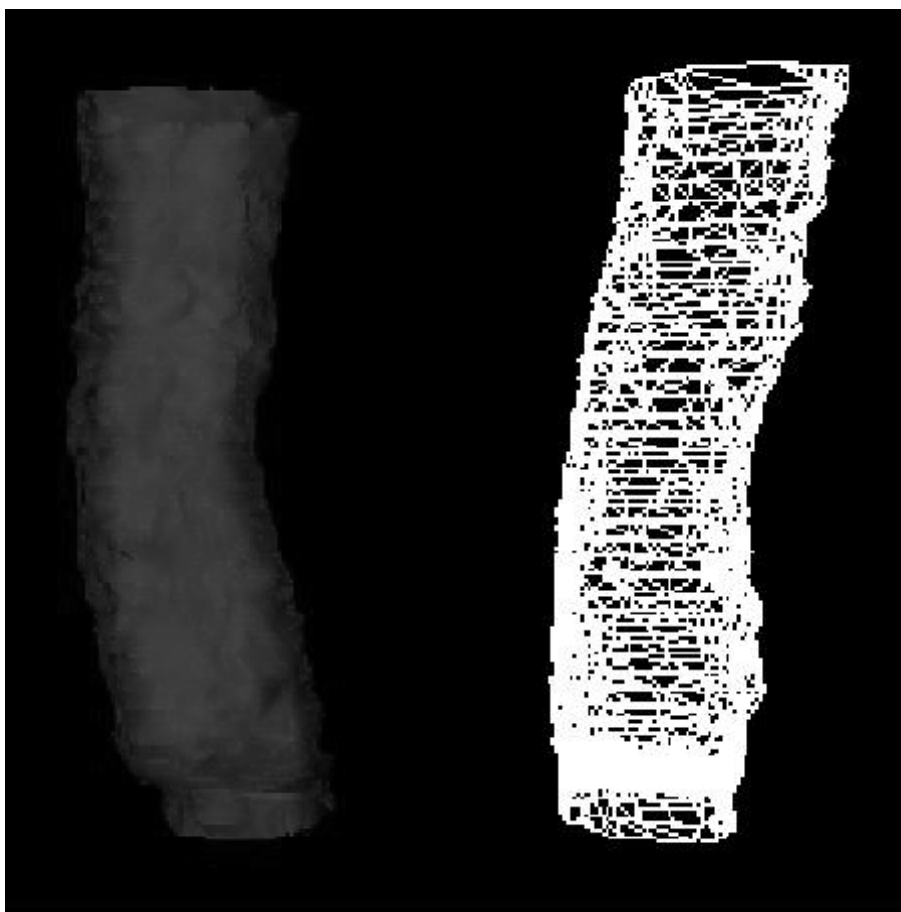


FIGURA 70: Reconstrução com texturas e com triângulos

3.10.3 - Geração da reconstrução

Finalmente os dados resultantes são disponibilizados na forma de um mundo em linguagem de realidade virtual, VRML, passível de ser visualizado e "navegado" em um browser adequado ou exportado via Internet. Nesta reconstrução pode-se visualizar de qualquer ângulo e proximidade uma reconstrução transparente do abdômen do paciente onde artérias e aneurisma foram reconstruídos com especial detalhe. Os níveis de transparência de quaisquer das anatomias mostradas podem ser alterados e os dados de mensuração, como a curva baricêntrica, podem ser visualizados ou apagados.

Encerrada a triangularização, inicia-se o processo de descrição da estrutura da artéria em linguagem VRML. Isto é feito transcrevendo-se as coordenadas 3D dos pontos e triângulos encontrados para instruções desta linguagem. A diferenciação na visualização da artéria é feita aqui, definindo-se se a artéria possuirá estrutura sólida ou possuirá estrutura sob forma de grade e quais cores cada uma assumirá.

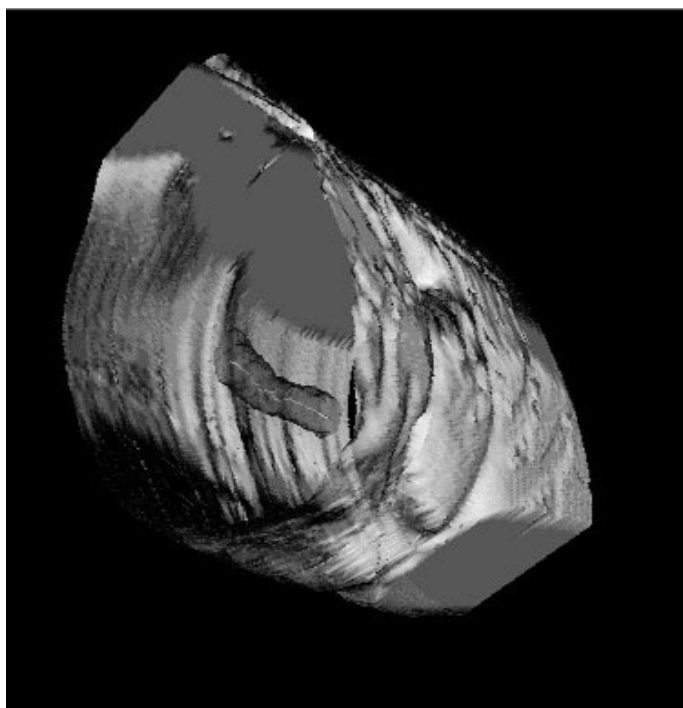


FIGURA 71: Reconstrução de um segmento do corpo humano

O algoritmo de reconstrução está sendo reimplementado utilizando superfícies spline bicúbicas de acordo com os padrões de realidade virtual para NURBs (*non-uni-*

form rational base splines). Isso fornecerá uma visualização mais perfeita e próxima da realidade.



FIGURA 72: Visualização através de um browser

4 - RESULTADOS E DISCUSSÃO

No que diz respeito as discrepâncias encontradas entre os especialistas da área na análise de imagens contendo aneurisma, com o auxílio da ferramenta de software estima-se que esta discrepância se atenua.

Em relação aos resultados obtidos do sistema de software, observou-se que o sistema detectou a luz da artéria e o tecido aneurismático e a quantificação esteve correta em todos os casos. No que se refere a reconstrução em 3D, facilitou a compreensão da disposição estruturais anatômicas como artéria aorta do paciente. Para uma maior robustez do sistema, será necessário um estudo maior de casos.

4.1 - Estudo comparativo dos resultados das mensurações

Foi realizado um estudo comparativo entre os resultados obtidos através do software desenvolvido neste trabalho e o software Advantage Workstation (AW) da GE Medical Systems. O software AW é uma avançadíssima ferramenta para análise de imagens radiológicas para a plataforma Sun, mas que no entanto, não realiza a detecção automatizada da artéria aorta, a qual deve ser feita pelo usuário através do mouse e utilizando técnicas simples de segmentação, baseadas em valores de densidade radiológica.

Essa ferramenta, juntamente com o equipamento (hardware) necessário possui um custo muito elevado, chegando, dependendo da configuração, a custar centenas de milhares de dólares o que torna a aquisição proibitiva para a grande maioria de instalações médicas, radiológicas e uso pessoal. Os testes aqui realizados foram efetuados na clínica DMI - Diagnóstico Médico por Imagem, situada em Florianópolis - SC, a qual adquiriu essa ferramenta.

Nas Fig.73 e Fig.74 são mostradas medições realizadas pelo software Advantage Workstation. O módulo de medição 3D do software AW calculou o maior diâmetro do aneurisma em 54,2 mm, o que é mostrado na Fig.. Na Fig.74 é mostrado o resultado do cálculo da extensão do aneurisma, que foi de 125,8 mm.



FIGURA 73: Mensurações 3D realizadas com o software Advantage Workstation



FIGURA 74: Mensuração 3D efetuada com o software Advantage Workstation

Nas Fig.75 e Fig.76 são mostradas medições realizadas pelo software StentPlanner, desenvolvido neste trabalho. Utilizando a linha baricêntrica os resultados foram de 54,82 mm para o diâmetro e 124,71 mm para a extensão. A medição da extensão foi realizada entre os slices de número 76 a 137. O software StentPlanner também calculou a extensão sem levar em consideração a linha baricêntrica. Nesse caso a medição da extensão resultou em 94,04 mm.

Portanto, as diferenças entre as mensuração efetuadas pelos dois softwares foram de:

Diferença no cálculo do diâmetro: 0,02 mm

Diferença no cálculo da extensão utilizando a linha baricêntrica: 1,09 mm

Diferença no cálculo da extensão não utilizando a linha baricêntrica: 31,76 mm



FIGURA 75: Mensuração 3D feita através do software StentPlanner



FIGURA 76: Mensuração 3D feita através do software StentPlanner

Deve ser feita ainda uma importante observação, no software StentPlanner são informado os slices inicial e final para o cálculo da extensão; já no AW a mensuração é feita a olho nú, sem informação dos slices que estão sendo envolvidos no processo. Portanto podem ter havido diferenças entre o número de slices utilizados para realizar o cálculo, o que causaria diferenças de vários milímetros na mensuração.

A Fig.77 mostra a reconstrução 3D da aorta, em especial do segmento aneurismático do mesmo exame utilizado anteriormente para as comparações de mensurações.

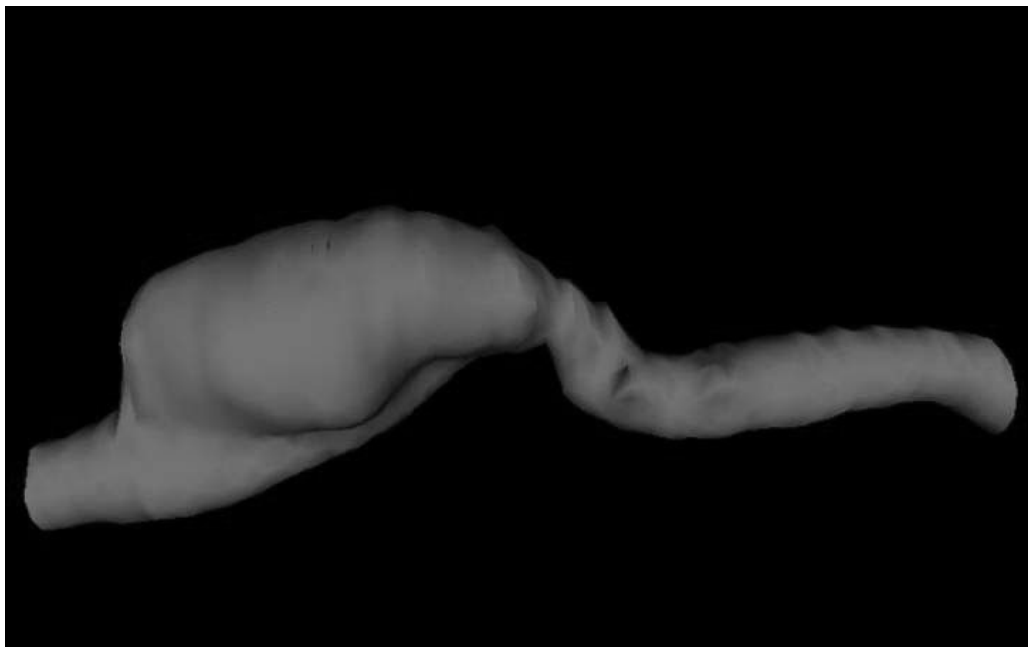


FIGURA 77: Reconstrução 3D feita através do software StentPlanner

4.2 - Trabalhos futuros

Várias melhorias devem ser implementadas para o aperfeiçoamento deste trabalho, entre elas:

- 1) Melhorar as rotinas de reconstrução resolvendo problemas tais como ligar dois segmentos de um corte a um segmento de um corte adjacente. Isso poderia ser feito através de triangularização de Delaunay.
- 2) Implementar o módulo de reconstrução utilizando o padrão NURB's, para fornecer resultados mais próximos da realidade.
- 3) Produzir a reconstrução em 3D baseada na interseção dos planos perpendiculares aos segmentos arteriais.
- 4) Estender o software para a) detecção de aneurismas ilíacos, femurais, torácicos e cerebrais e b) mensuração de todos vasos sanguíneos e não só o da artéria aorta.
- 5) Realizar um estudo estatístico com uma grande quantidade de pacientes com aneurisma com a finalidade de validação do software.

5 - CONCLUSÃO

O desenvolvimento de sistemas computacionais com capacidade de fornecer auxílio ao diagnóstico médico, exige antes de tudo, um suporte médico de alta capacidade. Estes sistemas não simplesmente precisam realizar determinada tarefa, é necessário que este sistema seja robusto o suficiente para proporcionar ao profissional confiabilidade e segurança.

Através deste trabalho, é possível observar e compreender a necessidade e a importância do desenvolvimento de uma ferramenta que seja capaz de prover ao profissional da área médica, auxílio na identificação e mapeamento de aneurismas de aorta abdominais de forma confiável, auxiliando no diagnóstico.

A partir dos estudos feitos neste trabalho, é percebido a existência de uma necessidade real. É importante que se desenvolvam métodos capazes de se adaptar a diferentes anatomias e configurações de aneurisma, tornando assim uma ferramenta realmente útil no auxílio a detecção e mensuração da patologia.

Também através deste trabalho, é possível entender o funcionamento de técnicas de processamento e análise de imagens como filtragens, segmentação e detecção de bordas.

O desenvolvimento da rede neural foi necessário a fim de classificar os segmentos de imagens candidatos à artéria aorta e ao aneurisma. Para a correta identificação deste último houve também a necessidade de utilizar uma técnica adicional chamada Método de Rotulação Consistente Inexato.

A localização automatizada da artéria aorta e do tecido aneurismático fornecerá ao profissional da área médica uma maior comodidade e segurança, evitando que o mesmo tenha recorrer a meios mais trabalhosos e propensos a erros para fazê-lo.

Como pretende-se que esta ferramenta possa ser utilizada em qualquer clínica, a ferramenta deve ser capaz de trabalhar com imagens geradas por equipamentos antigos ou modernos, fazendo desta uma ferramenta única independente do padrão da clínica.

Já a interface, deve proporcionar produtividade, respeitando as regras de ergonomia e interface, tornando-se de fácil uso para profissionais da área.

A possibilidade de visualização dos resultados em 3D em um browser é de muita importância para a análise por parte do corpo médico e o tamanho compacto das reconstruções possibilita a utilização em meio doméstico e sem a necessidade de obtenção de equipamentos de custo proibitivo.

E finalmente, com a mensuração da aorta e do aneurisma torna-se possível a criação de projetos de próteses de tamanho personalizado a cada paciente, evitando complicações decorrentes de uma medição errônea facilmente cometida ao se realizar uma análise visual sem o auxílio de uma ferramenta computacional.

Com isso, a continuação do desenvolvimento e aperfeiçoamento desta metodologia e das ferramentas mostra-se perfeitamente possível e necessária, com aplicação no cotidiano médico, tendo em vista o embasamento teórico/prático que o projeto Cyclops fornece e a flexibilidade que a ferramenta de desenvolvimento Smalltalk utilizada permite.

6 - GLOSSÁRIO

ACR American College of Radiology

AW Advantage Workstation

CAD Computer Aided Design

CLP Consistent Labeling Problem

DICOM Digital Image Communications on Medicine

FDA Food and Drug Administration

FTAB Future Errors Table

HU Hounsfield-Units

IE Information Entity

IOD Information Object Definition

IOM Information Object Module

LOD Level of Detail

NEMA National Electrical Manufacturers Association

NiTi Níquel-titânio

NURBs Non-Uniform Rational Base Splines

OSI Open Systems Interface

PACS Picture Archiving and Communications Systems

PTFE Politetrafluoretileno Expandido Ultra-fino

RM Ressonância Magnética

SNNS Stuttgart Neural Network Simulator

SOI Segment of Interest

SOP Service-Object Pair

SRSA Subcommittee on Reporting Standard for Arterial Aneurysms

TC Tomografia Computadorizada

TCP/IP Transfer Control Protocol/Internet Protocol

VRML Virtual Reality Modelling Language

WWW World Wide Web

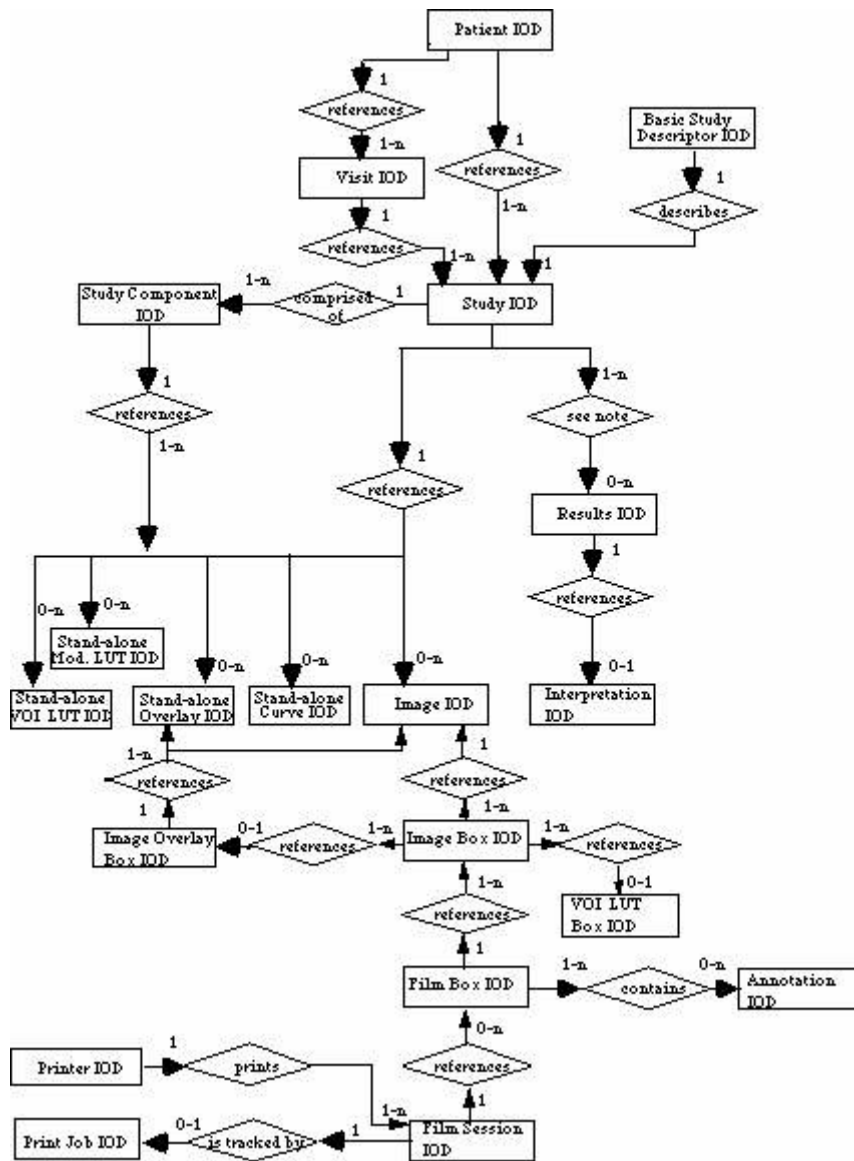
7 - REFERÊNCIAS BIBLIOGRÁFICAS

1. BARRETO, J.M. **Inteligência Artificial no Limiar do Século XXI**. Florianópolis, 1997, 1ª ed.
2. BLUM, U.; LANGER, M.; SPILLNER, G.; et al. **Abdominal aortic aneurysms; preliminary technical and clinical results with transfemoral placement of endovascular self-expanding stent-grafts**. Radiology, 198:25-31, 1996.
3. GONZALEZ, R.C.; WOODS, R.E. **Digital Image Processing**. San Francisco: Addison-Wesley, 1992.
4. HALLIDAY, D.; RESNICK, R.; WALKER, J. **Fundamentos de Física - Mecânica**. LTC, Vol. 1, 1993, 4ª ed.
5. HARALICK, R.; SHAPIRO, L. **Computer and Robot Vision**. Addison Wesley, Vol. 2, 1993, 379-426.
6. HOPKINS, T.; HORAN, B. **SMALLTALK an introduction to application development using visualworks**. Oxford: Prentice Hall, 1995.
7. JACOBOWITZ, G.R.; LEE, A. M.; RILES, S. T. et al. **Immediate and late explantation of endovascular aortic grafts: The Envascular Technologies experience**. J Vasc Surg 1999, 29:309-16
8. KLAUSMANN, P. **Variational Methods in Image Segmentation**. Diplomarbeit, Fachbereich Mathematik, Universität Kaiserslautern, Mai/1995.
9. LEZZI, G.; et al.. **Fundamentos de matemática elementar: geometria espacial**. São Paulo: Atual, 1985.
10. LEZZI, G.; et al. **Fundamentos de matemática elementar: geometria plana**. São Paulo: Atual, 1985.
11. MARRIN, C; CAMPBELL, B. **Teach Yourself VRML 2 in 21 days**. Sams Net, 1997.
12. MAY, J.; WHITE, G.H.; YU, W. et al. **Early experience with the Sidney and EVT prostheses for endoluminal treatment of abdominal aortic aneurysms**. J. Endovasc. Surg. 1995; 2:240-247.
13. MITTELHAEUSSER, G; KRUGGEL, F. **Fast Segmentation of Brain Magnetic Resonance Tomograms** in CVRMed'95: First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine, Nizza, 1995.

14. MUMFORD, D.; SHAH J. **Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems**. Comm. Pure Appl. Math., 1989
15. MOORE, S.W.; RUTHERFORD, R. B. et al. **Transfemoral endovascular repair of abdominal aortic aneurysm: Results of the North American EVT phase 1 trial**. J. Vasc. Surg. 1996; 23(4):543-553.
16. MOORE, W. S.; VESCERA, C.L. **Repair of abdominal aortic aneurysm by transfemoral graft placement**. Ann. Surg. 1994; 220:331-341.
17. PARKER, J.R. **Algorithms for Image Processing and Computer Vision**. New York: John Wiley & Sons, 1997.
18. PARODI, J.C; PALMAZ, J.C.; BAROND, H.D. et al. **Transfemoral intraluminal graft implantation for abdominal aortic aneurysms**, in Ann.Vasc.Surg. 1991, pp.491-499.
19. POPE, A.R., LOWE, D.G. **VISTA: A Software Environment for Computer Vision Research**. Internal Report, Department of Computer Science, University of British Columbia, Canada, 1994.
20. RUSS, J.C. **The image processing handbook**. Boca Raton: CRC Press, 1995, 2^a ed.
21. RUMMELHART, D.E.; HINTON, D.E.; WILLIAMS, R.J. **Learning Representations by Back-Propagating Errors** in Nature. 1996. 323:533-536.
22. SHARPI, A. **Smalltalk by Example - The Developer's Guide**. , McGraw-Hill, 1997, 1^a ed.
23. SHASTRI, L. **Semantic Networks: An evidential formalization and its Connectionist Realization**. Los Altos: Morgan Kaufmann Publ., 1998.
24. SILVEIRA, P.G.; GALEGO, G.N.; MANDELLI, M. B. et al. **Tratamento endovascular da doença aneurismática com prótese autoexpansível de NiTi**. São Paulo: XXIV Congresso do Capítulo Latino Americano da International Society of Cardiovascular Surgery e XVI Encontro Paulista de Cirurgia Vascular, fevereiro 1998, 27-29.
25. SILVEIRA, P. G.; GALEGO, G. N.; MANDELLI, M. B. et al. **Tratamento endovascular da doença aneurismática com prótese autoexpansível de NiTi. Experiência em 9 casos**. Florianópolis: II Simpósio Catarinense Catalão de Angiologia e Cirurgia Vascular, fevereiro 1999, 5-6.

26. The Khoros Group. **Khoros User's Manual, Release 1.0.5**, Vols. I - III, Albuquerque, USA: Department of Electrical Engineering & Computer Engineering, University of New Mexico, 1993.
27. v.WANGENHEIM, A.; BARRETO, J.M.; RICHTER, M.M.; KRECHEL, D. **Cyclops - Expert System Shell for the Development of Applications in the Área of medical Image Analysis**, in Jähnichen; Lucena(Eds.): Proceeding of the 4th German-Brazilian Workshop on Information Technology, Porto Alegre/Berlin, 1997.
28. v.WANGENHEIM, A. **Cyclops - Wissensbasierte Bildanalyse in der Medizin** INFIX-Verlag, Reihe, 1997.
29. VEITH, J.S. **Presidential address: Transluminally placed endovascular stented grafts and their impact on endovascular surgery**. J. Vasc. Surg. 1994; 20:855-860.
30. v.WANGENHEIM, A.; WAGNER, H.; CONRAD, P.; RICHTER, M.M.; Comunello, E.; Krechel, D. **A Knowledge-Based Approach for Failure Detection and Correction of Partially Failed Segmentations of Radiological Images** in BVM'2000: Bildverarbeitung Für Die Medizin, 2000.
31. VINCENT, L.; SOILLE, P. **Watersheds in digital spaces: An efficient algorithm based on immersion simulations**. In: IEEE Transactions on Pattern Analysis and Machine Intelligence. Pág.: 583-593. 1991.
32. WHITE GH, YU W, MAY J et al. **A new non-stented endoluminal graft for straight or bifurcatec endoluminal bypass**. J. Endovasc. Surg. 1994; 1:16-24.

Anexo 1 - Diagrama de Entidades e Relacionamentos do DICOM



Anexo 2 - Modelagem do Sistema em UML

