

UNIOESTE – Universidade Estadual do Oeste do Paraná

CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS

Colegiado de Ciência da Computação

Curso de Bacharelado em Ciência da Computação

**Implementação de um protótipo de software para reconstrução
de modelos 3D a partir de imagens no padrão DICOM**

Paolo Mautone Romera

CASCADEL

2011

Paolo Mautone Romera

Implementação de um protótipo de software para reconstrução de modelos 3D a partir de imagens no padrão DICOM

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação, do Centro de Ciências Exatas e Tecnológicas da Universidade Estadual do Oeste do Paraná - Campus de Cascavel

Orientador: Prof. Dr. Adair Santa Catarina

CASCADEL

2011

Paolo Mautone Romera

Implementação de um protótipo de software para reconstrução de modelos 3D a partir de imagens no padrão DICOM

Monografia apresentada como requisito parcial para obtenção do Título de Bacharel em Ciência da Computação, pela Universidade Estadual do Oeste do Paraná, Campus de Cascavel, aprovada pela Comissão formada pelos professores:

Prof. Dr. Adair Santa Catarina (Orientador)
Colegiado de Ciência da Computação,
UNIOESTE

Prof. Edmar André Bellorini
Colegiado de Ciência da Computação,
UNIOESTE

Prof. Dr^a Claudia Brandelero Rizzi
Colegiado de Ciência da Computação,
UNIOESTE

Cascavel, 17 de Novembro de 2011.

"Lembrar que estarei morto em breve é a ferramenta mais importante que já encontrei para me ajudar a tomar grandes decisões. Porque quase tudo - expectativas externas, orgulho, medo de passar vergonha ou falhar - caem diante da morte, deixando apenas o que é apenas importante. Não há razão para não seguir o seu coração. Lembrar que você vai morrer é a melhor maneira que eu conheço para evitar a armadilha de pensar que você tem algo a perder. Você já está nu. Não há razão para não seguir seu coração."

Steve Jobs

AGRADECIMENTOS

Agradeço ao Professor Adair Santa Catarina, pela ajuda e paciência na orientação deste trabalho. E a Bruno Eduardo Soares, pelo incentivo e ajuda na conclusão deste e de muitos outros trabalhos, dentro e fora da vida acadêmica.

Lista de Figuras

Figura 2.1	Exemplo de VR.....	7
Figura 2.2	Exemplo de <i>data set</i>	8
Figura 2.3	Exemplo de etiqueta.....	9
Figura 3.1	Estrutura interna de um Tomógrafo Computadorizado.....	13
Figura 3.2	Exemplo de Ressonância Magnética: a) Reconstrução tridimensional, b) Axial, c) Coronal T2[8].....	14
Figura 4.1	Tela 3D VR do software OsiriX.....	16
Figura 4.2	Tela 2D Viewer do software OsiriX.....	16
Figura 4.3	Menu principal do software ImageJ.....	17
Figura 4.4	MicroDicom.....	18
Figura 4.5	InVesalius, reprodução 3D de uma tomografia	19
Figura 5.1	Tela do protótipo em desenvolvimento.....	21
Figura 5.2	Sobreposição e Interpolação realizada pelo protótipo: a) Camadas sobrepostas; b) Camadas sobrepostas e interpoladas.....	23
Figura 5.3	Sobreposição de camadas	24
Figura 5.4	Rotações sem interpolação: a) no eixo Y, b) no eixo X.....	27
Figura 5.5	Rotação com interpolação no eixo X.....	27
Figura 5.6	Zoom Out obtido pelo protótipo: a) sem rotações, b) com rotação no eixo X.....	27
Figura 5.7	Projeções Tipos de projeção: a) Projeção perspectiva, b) Projeção ortogonal	28
Figura 5.8	Regiões de pontos considerados escuros, regiões A, B, C, D, E.....	29
Figura 5.9	a) Rotação da imagem sem interpolação b) Rotação da imagem com interpolação	30
Figura 5.10	<i>Sliders</i> de rotação da imagem.....	32

Lista de Abreviaturas e Siglas

DICOM – *Digital Imaging and COmmunications in Medicine*

NEMA – *National Electrical Manufacturers Association*

ACR – *American College of Radiology*

RSNA – *Radiological Society of North America*

FDA – *Food and Drug Administration*

PAC's – *Picture Archiving and Communication Systems*

TC – Tomografia Computadorizada

CR – Ressonância Computadorizada

VR – *Value Representation*

DR – Radiografia Digital

RM – Ressonância Magnética

CTI – Centro de Tecnologia da informação Renato Archer

Sumário

Lista de Figuras	vi
Lista de Abreviaturas e Siglas	vii
Sumário	viii
Resumo	x
1. INTRODUÇÃO	1
2. O PADRÃO DICOM	3
2.1. O Histórico.....	3
2.1.1. Volumes DICOM	4
2.2. Estrutura da Imagem.....	6
2.2.1. Tipos de Dados.....	7
2.2.2. <i>Header</i>	7
2.2.3. Campo de Dados.....	9
3. Equipamentos que utilizam o padrão DICOM	11
3.1. Aquisição de imagens.....	11
3.2. Equipamentos.....	12
3.2.1. Tomografia Computadorizada (TC).....	12
3.2.2. Ressonância Magnética (RM).....	13
3.2.3. Ultra-Sonografia.....	14
4. Implementações	15
4.1. OsiriX.....	15
4.2. ImageJ.....	17
4.3. <i>Photoshop Extended</i>	17
4.4. MicroDicom.....	18
4.5. InVesalius	19
5. Implementação do Software Proposto	20
5.1. 2D Convencional	20
5.2. Visualização 2D em Slideshow	21
5.3. Visualização 3D	21
5.3.1. Sobreposição	22
5.3.2. Interpolação	23
5.3.3. Manipulação	25
5.3.4. Projeção	26

5.4. Soluções Alternativas.....	29
5.4.1. Não Interpolar o Objeto	30
5.4.2. Rotacionar o Observador	31
5.4.3. Otimizações	31
5.5. Avaliação dos usuários	32
6 Conclusão	33
6.1 Trabalhos Futuros	34
Referências Bibliográficas	35

Resumo

O padrão DICOM é um protocolo de padronização utilizado nos sistemas médicos; este padrão estabelece como uma imagem deve ser armazenada e transmitida dentre os diversos sistemas existentes. Uma imagem deste padrão pode conter várias camadas, possibilitando sua representação em outros formatos dimensionais como, por exemplo, o 3D e o 3D a cores, dependendo do tipo de equipamento utilizado.

O presente trabalho apresenta a implementação de um sistema gratuito que possa ser utilizado por estudantes que necessitem entender imagens médicas computadorizadas; o sistema é simples, não necessita de um computador de última geração para sua execução e é totalmente portátil; foi desenvolvido em Java, necessitando apenas da máquina virtual Java instalada no computador para a execução do mesmo. Os resultados obtidos foram satisfatórios porém, o tempo de execução do sistema deixa a desejar. O desempenho do sistema foi comprometido em prol da portabilidade; somente as funcionalidades padrão da linguagem foram utilizadas.

Palavras-chave: DICOM, Reconstrução 3D, Java

Capítulo 1

Introdução

Um grande problema da medicina moderna é a interpretação de imagens médicas. Como sua representação é bidimensional, o responsável por interpretá-la deve tentar abstraí-la de forma tridimensional [1]. A tomografia computadorizada (TC) e a Ressonância Magnética (RM) são alguns dos exames empregados na realização de diagnósticos. Através destes exames é possível obter imagens bidimensionais do corpo, possibilitando a observação de tecidos moles e tecidos duros.

Independentemente do tipo de exame, os profissionais encontram problemas na interpretação correta dos modelos visuais destas imagens [1]. Como são imagens bidimensionais, equivalentes à visão de um corte transversal ou longitudinal, não é possível visualizar um órgão em sua totalidade. O que se veria em cada imagem é tão somente a vista de um corte deste órgão. A visualização completa das estruturas é possível através da modelagem das camadas bidimensionais em um modelo tridimensional.

O presente trabalho tem como objetivos compreender a estrutura de um arquivo no padrão DICOM e desenvolver um software que interprete o seu conteúdo, reconstruindo um modelo 3D a partir de imagens tomográficas. Embora softwares como este já existam, em sua maioria apresentam problemas como custo, dependência de plataforma, restrições funcionais ou dificuldade de manuseio por parte do usuário, alguns destes softwares serão apresentados no capítulo 4.

Como contribuição social pretende-se disponibilizar gratuitamente o software proposto aos estudantes de medicina, para que estes iniciem o entendimento dos exames computadorizados feitos atualmente, enriquecendo sua formação profissional. Este software

foi desenvolvido na linguagem Java visando à independência de plataforma, portabilidade e praticidade ao usuário.

O software proposto tem como entrada arquivos de tomografias computadorizadas Axiais (TC), restringindo seu funcionamento a este único tipo de exame. Convém ressaltar que ele também não poderá ser utilizado para diagnósticos, já que não possui o certificado de Boas Práticas emitido pela ANVISA – Agência Nacional de Vigilância Sanitária [2]. A ANVISA é o órgão da administração pública federal que tem por finalidade institucional promover a proteção da saúde da população por intermédio do controle sanitário da produção e da comercialização de produtos e serviços submetidos à vigilância sanitária, inclusive dos ambientes, dos processos, dos insumos e das tecnologias a eles relacionados [2].

O entendimento e manipulação de imagens DICOM, juntamente com a liberação da ferramenta, podem servir como incentivo para as clínicas da região, que se utilizam deste tipo de imagens, solicitarem o desenvolvimento de sistemas semelhantes. Atualmente qualquer tipo de necessidade médica/computacional obriga estas clínicas a buscarem soluções com fornecedores de outras regiões.

Capítulo 2

O PADRÃO DICOM

O DICOM é um padrão extremamente amplo, com muitos protocolos de transmissão de imagens e recursos, os quais não estão compreendidos nos objetivos deste trabalho. Portanto, neste capítulo será apresentado brevemente o padrão DICOM: sua história, funcionamento e estrutura.

2.1. Histórico

O padrão DICOM (*Digital Imaging and COmmunications in Medicine*) teve sua primeira versão desenvolvida em 1983, pelo *American College of Radiology* (ACR) e a *National Electrical Manufacturers Association* (NEMA). Porém, desde então, muitas outras instituições se uniram no aperfeiçoamento do padrão.

O objetivo inicial do padrão era a independência dos equipamentos imageadores utilizados na medicina. Para este fim foi criado o *ACR-NEMA Digital Imaging and Communications Standards Committee*, cujo esforço inicial foi o estudo dos diversos padrões existentes na época. Em 1985 foi publicado, na reunião anual do *Radiological Society of North America* (RSNA), a primeira versão do padrão DICOM, conhecida na época como *ACR-NEMA 300-1985* ou *ACR- NEMA 1.0*.

Esta primeira versão apresentava vários erros e imperfeições. O comitê *ACR-NEMA* estabeleceu então os grupos de trabalho, formando subcomitês e dividindo o trabalho. Em 1988 uma nova revisão do formato, chamada *ACR-NEMA 300-1988* ou *ACR-NEMA 2.0*, foi liberada. Esta revisão lentamente foi sendo adotada pelos fabricantes de equipamentos médicos e ainda hoje pode ser encontrada em aparelhos antigos.

Infelizmente a capacidade limitada de transmissão de dados do ACR-NEMA 2.0 e o número crescente de equipamentos de imagens digitais desenvolvidos motivaram uma nova revisão do padrão.

A terceira versão do padrão foi apresentada ainda em seu estágio inicial em 1992 e, em setembro de 1993, as primeiras nove partes de um formato totalmente novo estavam prontas. Esta versão foi chamada de ACR-NEMA DICOM ou DICOM 3.0. Como não existiram as versões DICOM 1.0 e 2.0, este padrão é comumente chamado apenas de DICOM [3].

Apesar do padrão não ter sido alterado, conta com um acréscimo de 23 suplementos, necessários em consequência das mudanças e necessidades que surgiram desde então. Hoje o padrão conta com 18 volumes (1 ao 20, os volumes 9 e 13 foram removidos do padrão) [3], contendo informações sobre os protocolos a serem seguidos, a estrutura da imagem e muitas outras especificações que serão apresentadas adiante.

As empresas empregam o DICOM em PAC's (*Picture Archiving and Communication Systems*); estes são sistemas médicos consistindo, necessariamente, de *Hardware* e *Software* usados na aquisição, armazenamento, representação visual das imagens e na transferência de imagens radiológicas e outras informações médicas entre computadores. Atualmente é empregado em praticamente todos os equipamentos médicos que trabalham com imagens [3].

2.1.1 Volumes DICOM

Os 18 volumes DICOM existentes, sendo que os volumes 9 e 13 foram removidos do padrão, estão disponíveis na página do NEMA, seu download é gratuito e feito por conexão ftp.

Os volumes existentes e seu conteúdo, resumidamente, são os seguintes:

Volume 1: Um *Overview* do padrão e seu histórico, funcionamento e apresentação da transmissão de imagens.

Volume 2: Estrutura de dados do padrão, fluxo de informações e segurança, regras de tipagem, especificações e modelo de implementação, interface de rede e transferência entre aparelhos e modelo de implementação de rede.

Volume 3: Continuação do volume 2. Armazenamento dos dados, protocolos, extensão dos dados no mundo real, codificação dos dados, definição de informação dos objetos, alguns exames que utilizam o padrão.

Volume 4: Comportamento, normativas de armazenamento, gerenciamento de dados, protocolo de serviço de armazenamento, paleta de cores suportada, definição de informação de cor.

Volume 5: Tipos de dados, codificação de valores, *Value Representation* (VR), elementos de dados, data sets e serviço de fornecimento de dados, sintaxes de transferência de imagens para outros formatos com e sem perda.

Volume 6: Cores, definição geral de cores, utilização de outros idiomas (alemão, francês) em seus descritores.

Volume 7: Modelo de representação em camadas, estrutura de comandos, estabelecimento de associações, serviços DIMSE, Protocolo DIMSE, parâmetros *C-Find*, serviço de requisição, sequenciamento, protocolo de armazenamento *C-STORE*, registro de aplicação para codificação DICOM, operações assíncronas.

Volume 8: Suporte para comunicação de imagens via web, protocolos de transporte, regras de empacotamento e codificação.

Volume 10: Modelo de comunicação geral DICOM, formato de arquivo DICOM, gerenciamento de informação.

Volume 11: Armazenamento, transmissão e distribuição de imagens DICOM em mídias, CD-ROM, DVD, etc.

Volume 12: Mapeamento do computador para formatos de mídia, formatos lógicos, sistemas de arquivo.

Volume 14: Padrão de armazenamento, apresentação de imagens em escala de cinza.

Volume 15: Segurança e sistemas de gerenciamento, conexões seguras, assinatura digital.

Volume 16: Mapeamento de recursos, *templates*, tabelas.

Volume 17: Modelos de curva, casos de uso, domínio da aplicação, sincronização.

Volume 18: Acesso Web para persistência de objetos DICOM (WADO), requisições html, resposta html.

Volume 19: Aplicação como Host, troca de mensagem, *MimeType*, interface de aplicação.

Volume 20: Transformações de DICOM para e através do padrão HL7.

Lembrando que os volumes 9 e 13 foram retirados do padrão, totalizando 18 volumes, e 23 suplementos que são atualizados constantemente; os dados apresentados acima correspondem aos volumes e suplementos até o início de 2011.

2.2 Estrutura da Imagem

O DICOM é um padrão de imagens que só pode ser criado a partir de equipamentos médicos especializados. A sua visualização e manipulação são livres em software, mas sua criação e edição são proibidas, visto que é um padrão muito amplo e complexo, possuindo diversas restrições funcionais; somente fabricantes autorizados podem, de fato, distribuir ferramentas para criação e edição de arquivos no formato DICOM[4].

Todos os *bytes* deste padrão são representados em *little endian*, devendo a transformação ser feita pela aplicação de visualização, se necessário. Como a representação depende da plataforma utilizada para criar as imagens, o padrão DICOM adotou o *little endian*, como forma de representação básica.

A representação *little endian* é a utilizada por vários processadores, dentre eles o intel x86, para armazenamento de dados em seus registradores. Sua característica é ter o *byte* menos significativo à esquerda, o que acarreta uma inversão no armazenamento. Por exemplo, os *bytes* F15D seriam armazenados como 5DF1 em *little endian*.

2.2.1 Tipos de Dados

Existem 27 tipos básicos de dados no padrão DICOM, conhecidos como *Representation Value* (VR). Estes foram projetados para encapsular e representar qualquer tipo de dado médico possível. Dentre estes dados estão os dados do equipamento utilizado para gerar a imagem, data, dados sobre o paciente, dentre outros [3]. A figura 2.1 apresenta exemplos de tipos básicos de dados (VR).

VR	Definição	Tipo de Caracteres	Tamanho
AE Application Entity	Caracteres que definem a entidade de aplicação	Alfanumérico	16 bytes máximo
DT	Data e hora	“0”-“9”, “.”	26 bytes máximo
OW	Texto cuja a codificação é especificada na sintaxe de transferência	N/A	Depende da Sintaxe de transferência
UID	Identificador único	“0” – “9”	64 Bytes máximo

Figura 2.1: Exemplo de VR [5]

2.2.2 Header

O *Header* contém obrigatoriamente um preâmbulo de 128 bytes, utilizado apenas para informações adicionais quanto à compatibilidade dos equipamentos. Caso não sejam usados devem ser preenchido com 00_H e, neste caso, podem ser ignorados.

Seguido ao preâmbulo existe um campo de 4 *bytes* chamado prefixo. Este campo deve conter os caracteres “DICM”, em letras maiúsculas, para indicar o formato do arquivo. Este campo não faz parte da estrutura DICOM.

No *Header* os dados da imagem são divididos em *data sets*, conforme apresentado na figura 2.2.

Etiqueta		VR		Tamanho	Valor
Número do Grupo	Número do Elemento	Caractere de 2 bytes estabelecendo a representação do valor (tipo do dado)	Reservado	Tamanho em bytes do campo valor	
2 bytes	2 bytes	2 bytes	2 bytes	4 bytes	Variável

Figura 2.2: Exemplo de *data set* [5]

O *data set* contém de 3 a 4 campos: a *tag* ou Etiqueta, o VR, o tamanho dos dados e o valor da variável, sendo que o campo VR pode ficar implícito em algumas implementações. Caso seja explícito a representação dos dados está descrita no volume 5 do padrão. A *tag* identifica quais os dados que estarão presentes em seguida e a variável é o dado propriamente dito. É importante ressaltar que o tamanho do *Header* é variável e o número de *tags* pode ser de alguns milhares, podendo crescer a cada nova revisão. Porém, apenas as *tags* importantes para cada sistema necessitam ser acessadas, as demais podem ser ignoradas.

Cada *tag* é dividida em duas partes, por exemplo, a *tag* (0008, 0008). Os primeiros 2 bytes hexadecimais representam o grupo, onde serão agrupados os dados semelhantes. Os dois bytes posteriores representam o elemento. A *tag* em questão representa “*ImageType*”, ou tipo de imagem, referente a codificação do *data set* e que será explicada posteriormente. Os grupos pares estão representados no volume 6 do padrão e fazem parte do DICOM, os grupos ímpares são para uso privado[5].

O *header* termina com a *tag* (7FE0, 0010), que indica o início da área de dados da imagem. Neste ponto dados da imagem já foram lidos do cabeçalho, como o tamanho da imagem, o número de bits por *pixel*, podendo ser de 8, 12, 16 bits, a distância entre os *layers* ou fatias do exame e o tipo de equipamento, dentre outros.

Todas são informações que podem ajudar na representação da imagem, porém após a *tag* de término de *header* e o campo de dados, que indica onde se encontram os dados dos *pixels*, devem ser processado de acordo com a *tag ImageType*.

A tag *ImageType* indica se a imagem está codificada com compressão, nos formatos JPEG, JPEG2000, JPEG-LS, compressão RLE (*Run-Lenght Encoding*) e ZIP, ou sem compressão, nos formatos TIFF e BMP, ou ainda formatos de vídeo como o AVI[3], podendo variar de acordo com a qualidade de imagem desejada pelo utilizador.

Na figura 2.3 são apresentados os campos de algumas *tags*, seguidas por seus títulos e descrições das mesmas.

Grupo	Elemento	Título	Descrição
0002	0010	Transfer Index UID	Define a sintaxe de transferência
0002	0016	AE Fonte	Título da entidade de aplicação.
0008	0016	SOP Class UID	Define a classe par serviço-objeto (SOP)
0008	0060	Modalidade	Modalidade da imagem (MR, CT, XR)
0008	0070	Fabricante	GE, Siemens, Toshiba
0018	0050	Espessura do Corte	Parâmetro de aquisição da imagem
7FE0	0010	Pixel Data	Imagem

Figura 2.3: Exemplo de etiqueta[5]

2.2.3 Campo de Dados

A área de dados propriamente dita ou campo de dados, onde se encontra a imagem codificada, segue a *tag* (7FE0, 0010). Em uma TC, um campo de dados ocupa cerca de 524 KB e, na maioria dos casos, é monocromática. O campo de dados ocupa o restante do arquivo; toda a extensão posterior representa os *pixels* da imagem.

Um dos maiores problemas na representação deste tipo de imagem em um computador normal é a memória. A quantidade de memória para estudar este tipo de imagens é enorme. Considerando uma TC com 500 camadas, cada uma com tamanho igual a 524 KB, totaliza no disco rígido 250 MB.

A capacidade de armazenamento vem crescendo como nunca, logo, isto poderia sanar o problema para estudar as tomografias. Entretanto exames com imagens mais complexas,

como a reconstrução 3D a cores e a mamografia digital, que ocupam cerca de 6 e 117 vezes o espaço de uma tomografia, respectivamente, conduzem ao uso da compressão dos pixels da imagem, reduzindo significativamente o espaço ocupado no disco. Porém, a reconstrução 3D destes tipos de imagens é difícil, pois ocupa grande parte ou toda a memória principal disponível [3].

Capítulo 3

Equipamentos que utilizam o padrão DICOM

Neste capítulo serão apresentados alguns dos equipamentos que utilizam o padrão DICOM, os métodos de aquisição de imagens utilizados e algumas diferenças entre os exames realizados com estes aparelhos.

3.1. Aquisição de imagens

Apesar do foco deste trabalho ser a TC Axial é importante mencionar alguns equipamentos que utilizam o DICOM e algumas de suas características. Os equipamentos estão especificados nas *tags* (0008, 0060) que identifica qual o tipo de modalidade de exame, e a *tag* (0008, 0070) que identifica o fabricante do equipamento.

A maioria destes equipamentos produz imagens no formato digital. O Raio-X ou radiografia é o principal meio de diagnóstico por imagens no Brasil e, em sua grande maioria, utiliza aquisição analógica (por filme radiográfico) [6].

Entretanto, a aquisição de imagens pode ocorrer através da Radiografia Digital (DR). Neste caso as imagens são obtidas por aparelhos de Raios-X que possuem uma placa de circuitos sensíveis a este tipo de radiação. O aparelho, através desta placa de circuitos, gera uma imagem digital e a envia diretamente para o computador na forma de sinais elétricos.

As imagens podem também ser obtidas pela Radiografia Computadorizada (CR). As imagens são obtidas utilizando-se aparelhos de radiologia convencional (os mesmos utilizados para produzir filmes radiográficos), porém seus “chassis” com filmes radiológicos são substituídos por “chassis” com placas de fósforo.

Os sistemas de aquisição de imagens digitais oferecem maior versatilidade na manipulação das imagens, além de se mostrarem muito mais precisas e homogêneas entre si, em relação ao brilho e contraste das imagens.

Algumas das vantagens de se utilizar radiografias digitais são: facilidade de exibição da imagem gerada, redução da dose de Raios-X, facilidade na aquisição, processamento, armazenamento e transmissão das imagens.

As imagens geradas podem ser visualizadas em monitores e não em filmes contra a luz, usando negatoscópios. Quanto à dose de Raios-X, ajusta-se esta para que a imagem tenha o mínimo de ruído, conseqüentemente o paciente absorve uma quantidade menor de radiação.

Operações de Processamento de Imagem como o aumento do contraste, equalização por histograma e subtração de imagens, podem ser utilizadas para melhorar a visualização das características importantes da radiografia. Com imagens digitais estas operações são facilmente codificadas em software.

Por fim, a facilidade de aquisição, armazenamento e recuperação de imagens, permite a transmissão de dados para longas distâncias, usando rotas de comunicação de dados.

3.2. Equipamentos

Existem muitos outros equipamentos que utilizam o DICOM, porém seu funcionamento é similar, alterando-se somente o método de aquisição dos sinais que formarão os pixels da imagem final. A seguir três exames que explicam o funcionamento da aquisição de imagem.

3.2.1. Tomografia Computadorizada (TC)

Um dos exames mais utilizados e foco deste trabalho é a Tomografia Computadorizada.

Em um tomógrafo, uma série de Raios-X são emparelhados em uma espécie de tubo, denominado *gantry*. Na extremidade oposta encontra-se o receptor. O tubo se move pelo corpo do paciente, podendo ser Linearmente (Tomografia Computadorizada Axial) ou Circularmente ao redor do paciente (Tomografia Computadorizada Helicoidal). Este último torna possível a visualização 3D de forma automática.

A imagem é obtida através da movimentação do gantry, onde o receptor capta a radiação e transmite as informações para o computador ao qual está conectado. Neste computador a imagem pode ser codificada e visualizada.

Na figura 3.1, retirada da página em [7], as setas vermelhas indicam os aparelhos de Raio-X, as setas verdes indicam o tubo de Raios-X (D), e o receptor (T), por fim a seta preta (R) indica a direção dos dados[6].

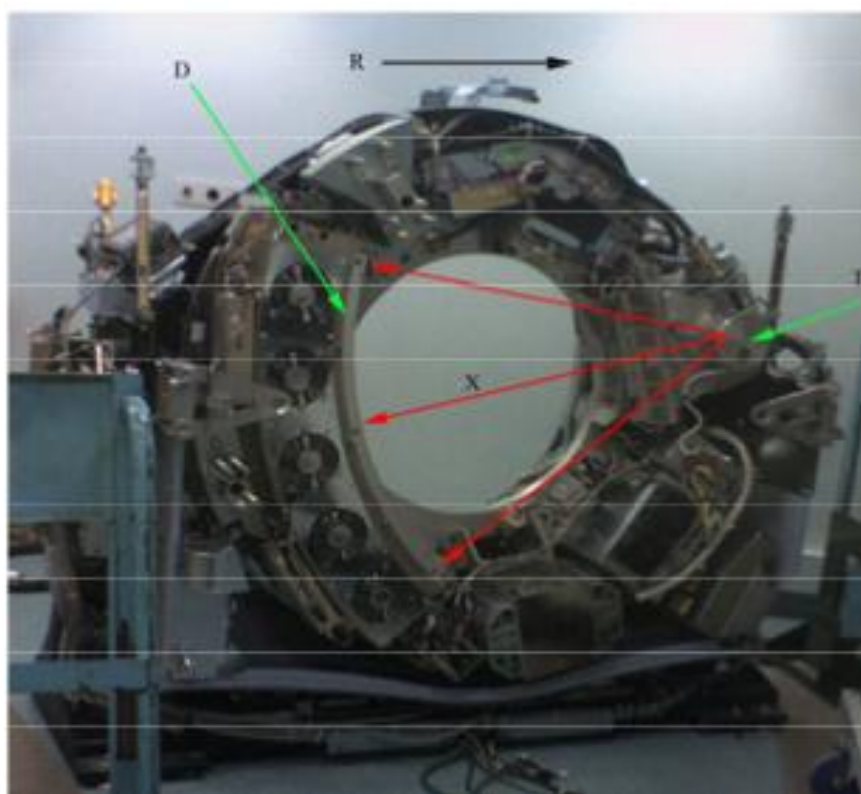


Figura 3.1: Estrutura interna de um Tomógrafo Computadorizado [7].

3.2.2. Ressonância Magnética (RM)

A ressonância magnética, antigamente chamada de Ressonância Nuclear Magnética, é um método de obtenção de imagens computadorizadas que não utiliza radiação ionizante.

Esta técnica se divide em três etapas: alinhamento, excitação e detecção de radiofrequência. O alinhamento apresenta o comportamento dos átomos quando submetidos a um campo magnético.

Na excitação cada núcleo de hidrogênio "vibra" numa determinada frequência proporcional ao campo magnético aplicado. O aparelho emite uma onda eletromagnética na mesma frequência. Neste momento acontece uma troca de energia da onda emitida pelo equipamento para os átomos de hidrogênio, este fenômeno é conhecido como ressonância.

Na detecção de radiofrequência, os núcleos de hidrogênio que receberam a energia tornaram-se instáveis. Ao retornar ao estado normal, eles emitem ondas na mesma frequência. Então o equipamento detecta essas ondas e determina a posição no espaço e a intensidade da energia.

As imagens de RM, como as apresentadas na figura 3.2, são capazes de demonstrar diferentes estruturas no cérebro e demonstrar alterações mínimas na maioria das doenças[8].

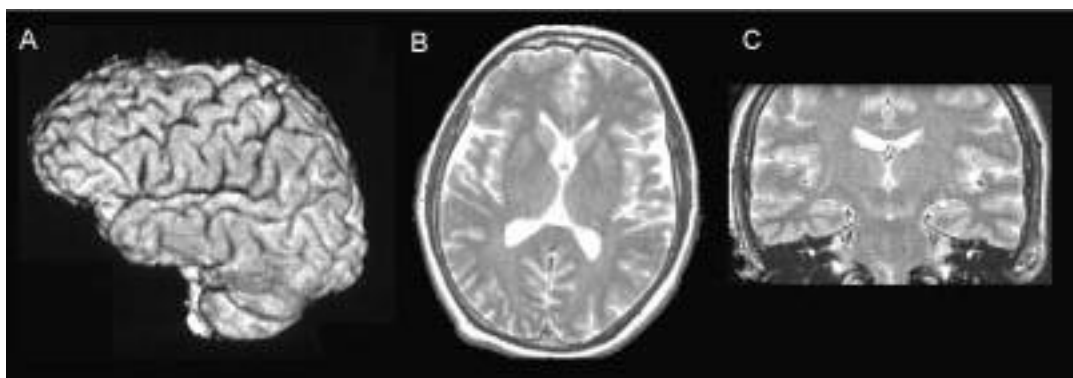


Figura 3.2: Exemplo de Ressonância Magnética: a) Reconstrução tridimensional, b) Axial, c) Coronal T2[8]

3.2.3. Ultra-Sonografia

A ultra-sonografia consiste de vibrações mecânicas periódicas, ou seja, uma onda que se propaga através do meio material. É gerada pela aplicação de um sinal elétrico a um dispositivo transdutor. Diferenças de pressão em determinado meio propagam-se constituindo as ondas sonoras, sem que haja perturbação nas partículas deste meio durante o processo.

A ultra-sonografia utiliza alguns parâmetros como o período, comprimento de onda, frequência, intensidade, amplitude e velocidade de propagação destas ondas para encontrar uma medida a partir do eco gerado na amostra analisada (músculos e gorduras). É um método em que o pulso ultra-sônico é transmitido por meio de amostra e os ecos, gerados pelas diversas interfaces na amostra, são detectados e processados na forma de medidas [9].

Capítulo 4

Implementações

Este capítulo tem como objetivo apresentar as implementações existentes, seus pontos positivos e negativos.

4.1. OsiriX

O Software OsiriX [10], desenvolvido para a plataforma Mac OSX na linguagem Objective-C, é capaz de modelar as imagens DICOM em diversas representações, permitindo visualização das camadas em 2D e visualização em 3D através da união das mesmas, além de exportar as imagens DICOM para os formatos de imagem convencionais ou para formatos de vídeo.

O OsiriX é apresentado em duas versões (32-bit e 64-bit). A versão 32-bits é gratuita e de código aberto, porém sem a aprovação do FDA (*Food and Drug Administration*), por isso não pode ser utilizada para diagnósticos. Já a 64-bits, também chamada de OsiriX MD, é uma versão comercial e aprovada pelo FDA, que usa como base o OsiriX 32-bits. Além destas versões ele também oferece versões para dispositivos móveis como iPad e iPhone. Para utilizar este software é necessária a utilização de equipamentos *Apple* (*Macintosh, iPhone, iPad*) e que possuam sistemas operacionais Mac OSX, ou nas versões móveis o IOS.

Possui uma vasta gama de funcionalidades, sendo capaz de manipular imagens de muitos equipamentos diferentes, com ótima qualidade. Mesmo na sua versão gratuita possui recursos suficientes para visualização de imagens monocromáticas e coloridas, oferecendo também a possibilidade de exportá-las para diversos formatos de imagem e vídeo. Apresenta também um banco de dados onde os estudos podem ser armazenados, ou abertura destes por

CD. As figuras 4.1 e 4.2 mostram exemplos de imagens em 3D e 2D no OsiriX, respectivamente.

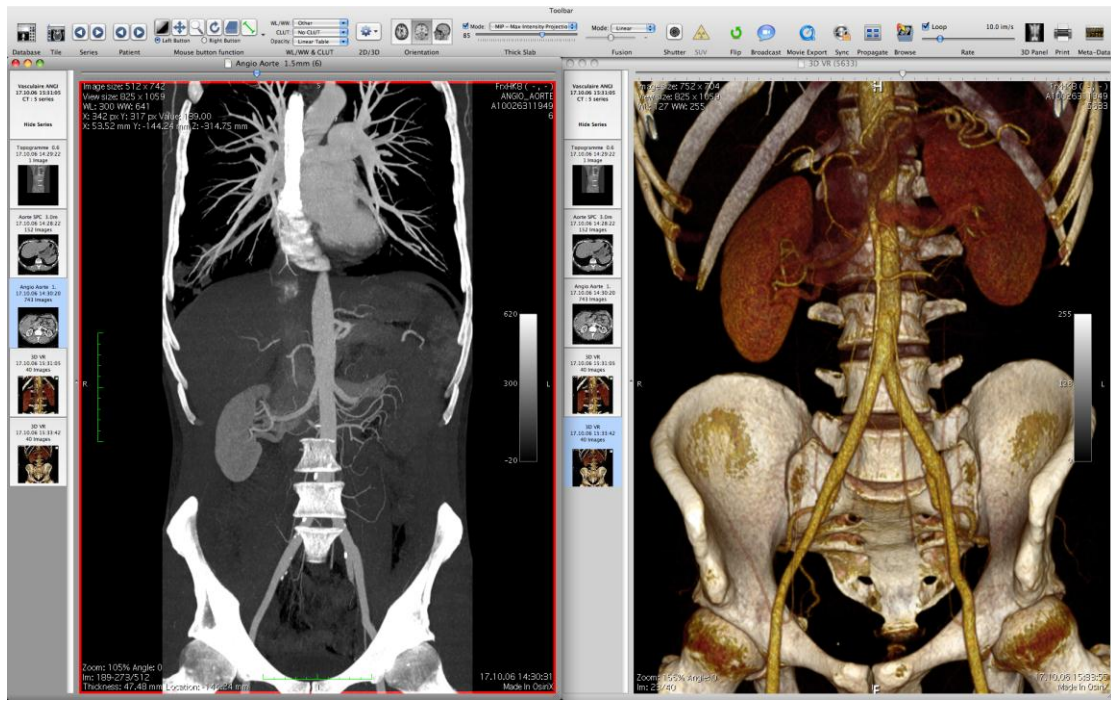


Figura 4.1: Tela 3D VR do software OsiriX[11]

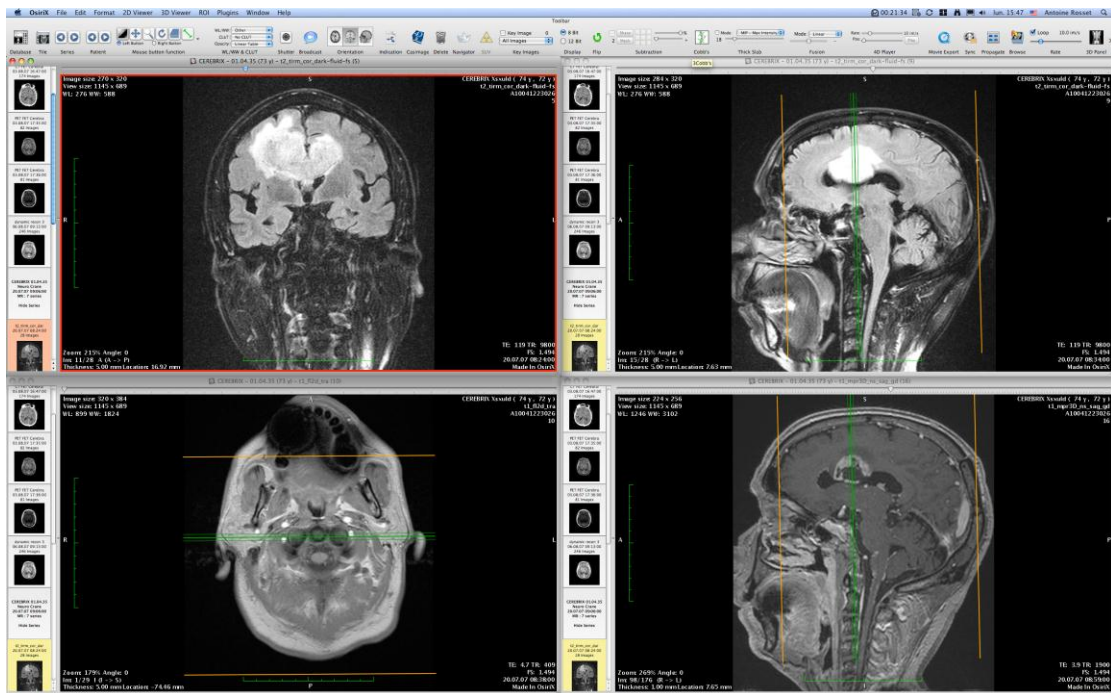


Figura 4.2: Tela 2D Viewer do software OsiriX [11]

4.2. ImageJ

O ImageJ é um software de código aberto, desenvolvido em Java, capaz de abrir imagens DICOM. Embora seja muito mais simples que o OsiriX, oferece vários plugins que efetuam algumas operações semelhantes como a visualização 2D e 3D, seleção e exportação de imagens.

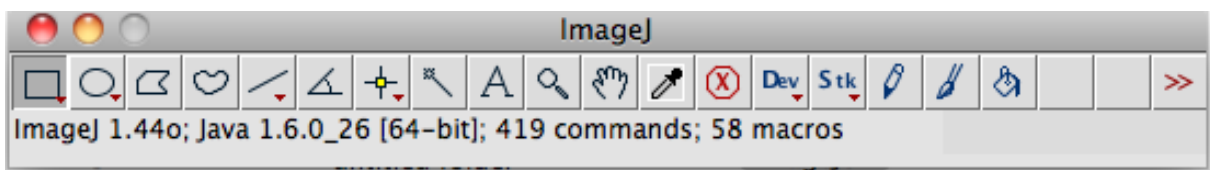


Figura 4.3: Menu principal do software ImageJ

Todas as funções estão disponíveis no menu apresentado na figura 4.3. Também é possível automatizar algumas funções através de macros.

Devido a seu desenvolvimento ser em Java, o software é facilmente portátil para qualquer plataforma com máquina virtual Java.

Uma das vantagens de sua utilização é a possibilidade de elaborar novos *plugins* que atendam as necessidades dos usuários. Porém, o ImageJ é um software de utilização complexa e suas funções DICOM necessitam conhecimento prévio de anatomia para utilização [12].

4.3. Photoshop Extended

O *Photoshop* pode ler os quadros de um arquivo DICOM e convertê-los em camadas. Também pode colocar os quadros DICOM em uma grade ou abri-los como um volume 3D. Os arquivos suportados são de 8, 10, 12 ou 16 bits por *pixel*. Onde os arquivos de 10 e 12 bits são convertidos em arquivos de 16 bits.

Qualquer ferramenta *Photoshop* pode ser utilizada para ajustar, marcar ou fazer comentários *Photoshop*. Os demais devem ser salvos como DICOM, PSB (Formato de

Documento Grande), *Photoshop*, *Photoshop PDF*, *Photoshop Raw*, PNG ou TIFF. O *Photoshop* pode salvar imagens DICOM, porém todos os filtros e máscaras são descartados [13].

4.4. MicroDicom

Embora muito menos conhecido que os demais, o MicroDicom é outra ferramenta gratuita desenvolvida para a plataforma Windows. Equipada com as ferramentas básicas de manipulação de imagens DICOM como zoom, rotação, seleção, etc. Também é capaz de exportar as imagens para os formatos mais conhecidos [14].

Sua interface é simples e intuitiva assim como sua utilização, porém seus recursos visuais são escassos, apresentando somente a visualização 2D de imagens individuais ou em slides. A figura 4.4 mostra um exemplo de imagem vista no MicroDicom.

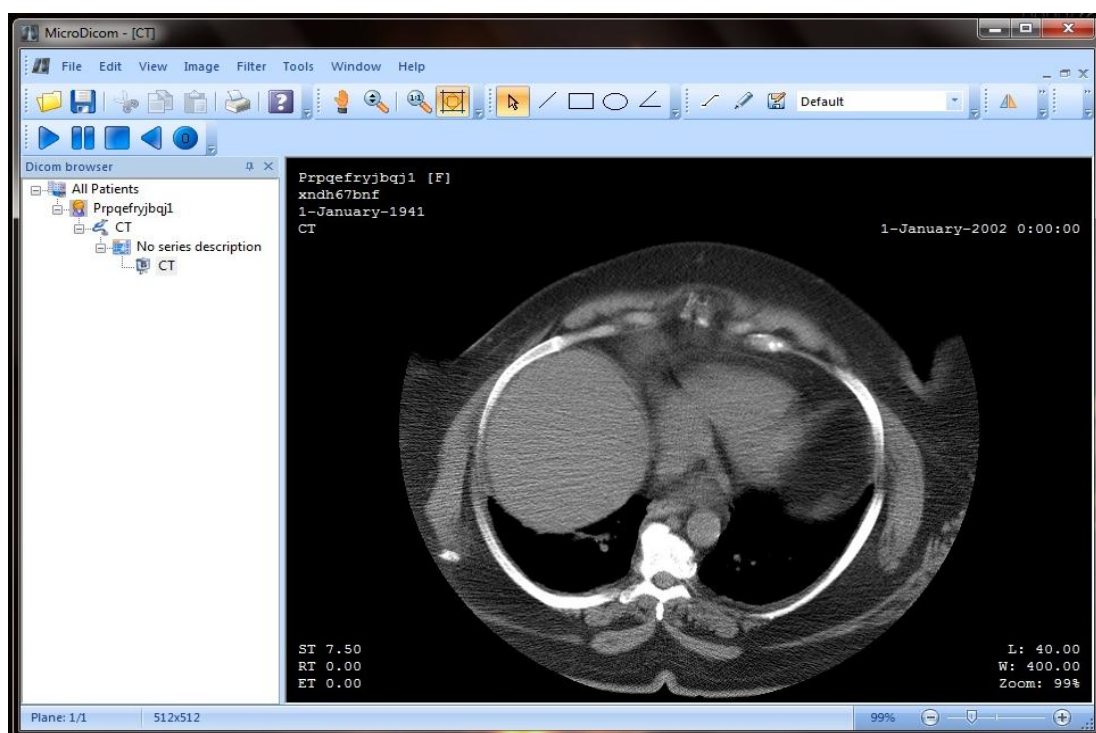


Figura 4.4: MicroDicom [14]

4.5. InVesalius

Este software foi desenvolvido pelo antigo CenPRA atual CTI(Centro de Tecnologia da informação Renato Archer), unidade do Ministério de Ciência e Tecnologia, nas linguagens Python e C++, sua distribuição é OpenSource e atualmente funciona nas plataformas Windows e GNU/Linux.

O InVesalius iniciou-se em 2001 no contexto do projeto Prototipagem Rápida na Medicina e vem sendo amplamente utilizado por hospitais e cirurgiões desde 2004. Atualmente está em sua versão 3.0 Beta 2, oferecendo uma enorme gama de funcionalidades e manipulação das imagens apresentadas. Basicamente este software, faz as transformações das imagens de Tomografias Computadorizadas e Ressonâncias Magnéticas para os planos 2D e 3D, em uma qualidade considerável. Sua comunidade vem crescendo largamente nos últimos anos [16].

A figura 4.5 mostra a utilização do InVesalius em uma reprodução 3D.

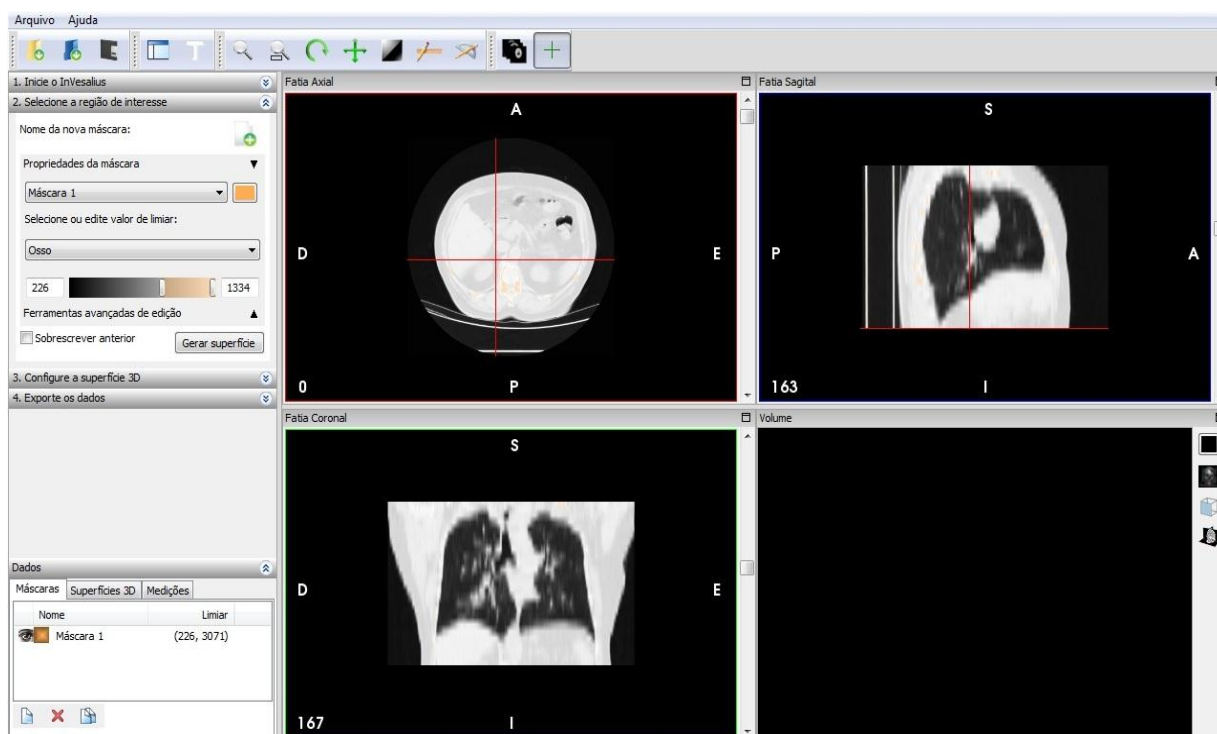


Figura 4.5: InVesalius, reprodução 3D de uma tomografia.

Capítulo 5

Implementação do Protótipo Proposto

Este capítulo tem como objetivo apresentar o protótipo proposto e as técnicas utilizadas em seu desenvolvimento. Também serão relatadas as dificuldades encontradas, e outras possíveis soluções para estas dificuldades.

5.1. 2D Convencional

O sistema proposto tem como objetivo a reconstrução das imagens DICOM em 3D. Para implementar a reconstrução 3D foi necessário implementar as funções convencionais 2D. As funcionalidades em 2D possibilitam a visualização dos exames em imagens individuais ou em *slideshow*.

O decoder utilizado na implementação do protótipo é distribuído pela *Sun Microsystem*. Ele é capaz de ler imagens DICOM de 8, 12 e 16 bits, dando importância somente aos VR significantes para a abertura da imagem em um computador pessoal. Juntamente com o pacote Jai, torna-se possível abrir e representar a área de dados da imagem em uma instância *RendererRaster*, que nada mais é que uma imagem preparada para ser processada e apresentada na tela do computador.

Após o *raster* ser obtido, uma auto-escala é realizada para que seja possível plotar a imagem caso o *raster* tenha mais de 8 bits. Desta forma os valores de cada *pixel* ficam no intervalo abrangido por um byte (0-255). Neste ponto a imagem já pode ser plotada.

A tela, mostrada na figura 5.1, apresenta a imagem ampliada à direita, enquanto que na esquerda são apresentadas as miniaturas da imagem visualizada.

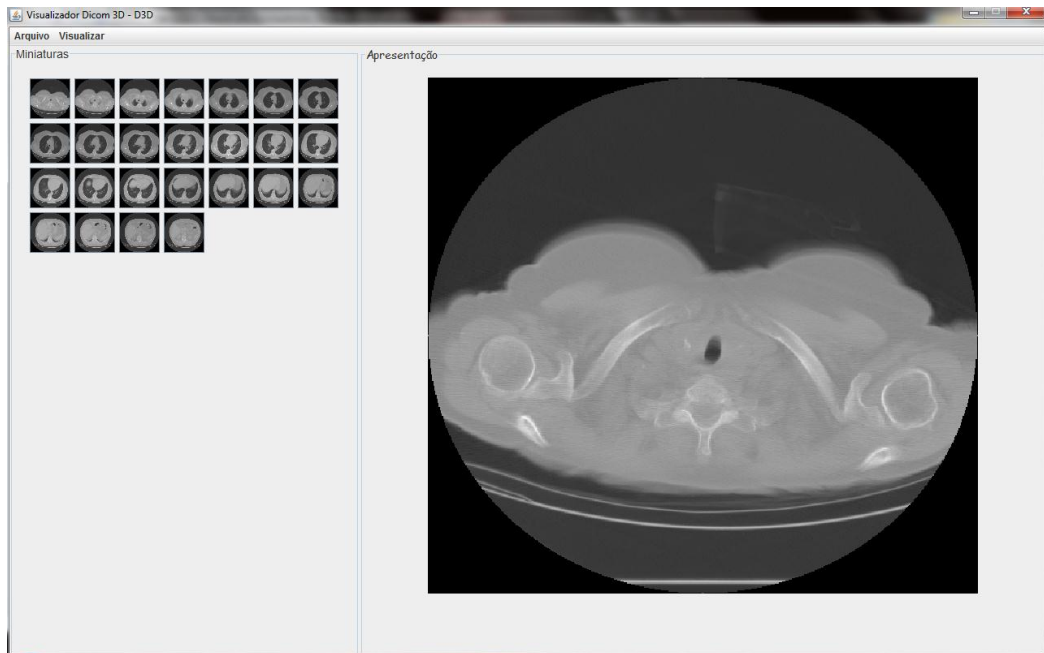


Figura 5.1: Tela do protótipo desenvolvido

5.2. Visualização 2D em *Slideshow*

A implementação da visualização em *slideshow* pode ser considerada simples. Uma *thread* é lançada quando o recurso é acionado. Esta *thread* invoca a próxima imagem da sequência em um determinado espaço de tempo; quando a sequência chega ao fim a apresentação é reiniciada, sendo também possível parar a visualização.

Justifica-se o uso de uma *thread* pelo fluxo contínuo de processamento realizado pelo sistema. Caso não se utilizasse uma *thread* o sistema ficaria parado e o *slideshow* somente seria interrompido encerrando o protótipo. A imagem a ser exibida num espaço de tempo é selecionada por esta *thread*.

5.3. Visualização 3D

A visualização tridimensional ocorre por meio de quatro etapas igualmente importantes; cada uma delas é responsável por uma parte do processo de manipulação e

visualização das imagens como um volume 3D. Estas etapas, apresentadas a seguir, são: sobreposição, interpolação, manipulação e, por fim, projeção.

Todo o processo é aplicado sobre uma matriz tridimensional de elementos do tipo *byte*. Nesta matriz estão armazenadas todas as imagens contidas no exame tomográfico e sobre ela todas as etapas de processamento são realizadas.

Esta matriz terá as dimensões altura x largura x profundidade. A altura é igual a altura das camadas, a largura é igual a largura das camadas, e a profundidade é calculada pela equação 5.1. Cada imagem interpolada representará um milímetro da realidade, ou seja se o corte entre duas camadas da tomografia tiver como espaçamentos 10mm, serão interpoladas nove imagens entre elas.

$$profundidade = ((NC - 1) \times E) + 1 \quad (\text{eq. 5.1})$$

Onde NC é o número de camadas originais do exame, ou seja, todas as camadas DICOM abertas pelo sistema, E é o espaçamento entre as camadas (dado em milímetros e obtido no cabeçalho do arquivo).

A figura 5.2 mostra como as camadas são sobrepostas e interpoladas. Na esquerda temos as imagens sobrepostas no espaço, separadas pela largura de corte obtido na imagem, ou seja, a distância entre as camadas originais. Na direita é apresentada a imagem interpolada, onde as camadas interpoladas passam a representar uma aproximação do original, com a espessura de 1 mm cada. O processo completo será explicado posteriormente.

5.3.1. Sobreposição

Inicialmente organizam-se as imagens originais (aquelas armazenadas no arquivo) nas camadas da matriz tridimensional; assim, cada camada da imagem original será armazenada em sua posição multiplicada por $(E * (\text{Número da camada} - 1))$, formando *gaps* de tamanho E ao longo da profundidade da matriz. Após esta etapa as imagens estão sobrepostas.

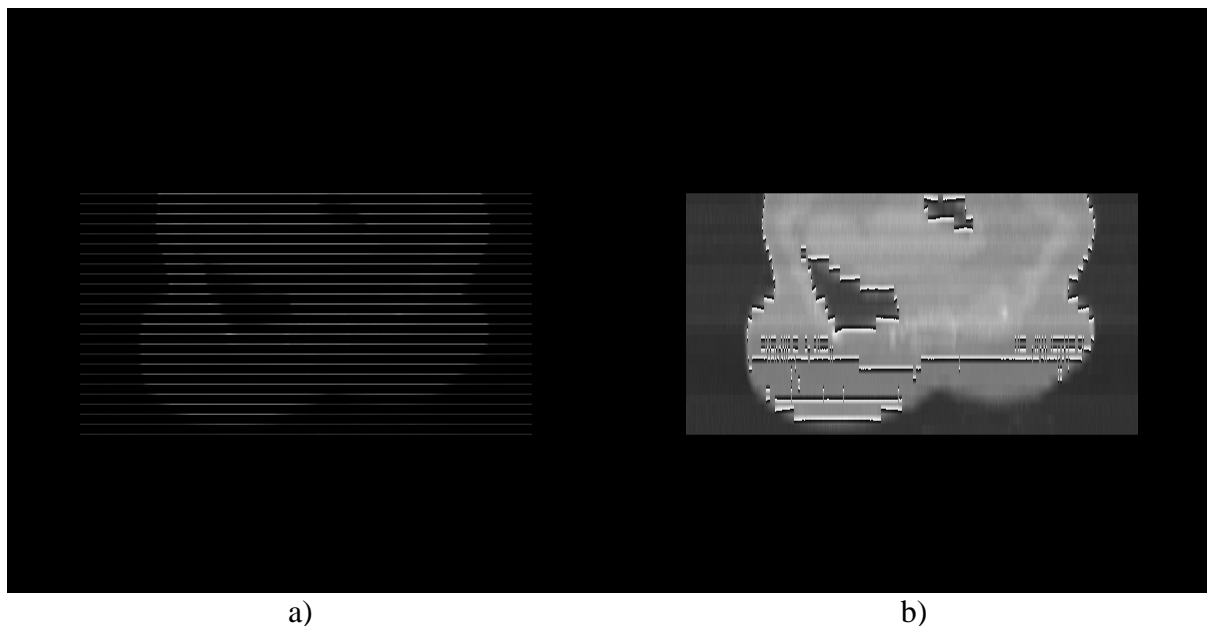


Figura 5. 2: Sobreposição e Interpolação realizada pelo protótipo: a) Camadas sobrepostas; b) Camadas sobrepostas e interpoladas

A figura 5.3 mostra como é feita a sobreposição das camadas originais na memória. A matriz resultante terá uma profundidade proporcional ao número de camadas e ao espaçamento entre elas. Antes de serem sobrepostas elas passam por um processo de autoescala, para que seus *pixels* estejam no intervalo aceito para que possam posteriormente ser projetadas posteriormente.

5.3.2. Interpolação

Após concluída a etapa de sobreposição existem gaps na imagem, ou seja, espaços vazios sem informação útil. Para cada camada preenchida na matriz, existem várias vazias. A etapa de interpolação visa preencher estas camadas vazias, eliminando os *gaps*.

Interpolar imagens é um processo computacionalmente caro, mas não utiliza as mesmas áreas de dados. Essa característica possibilita a divisão do processo de interpolação em threads, acelerando sua execução. O número de *threads* é igual ao número de camadas

originais menos um. O processo de interpolação é realizado em aritmética incremental, evitando o uso repetitivo de operações matemáticas caras, como a divisão.

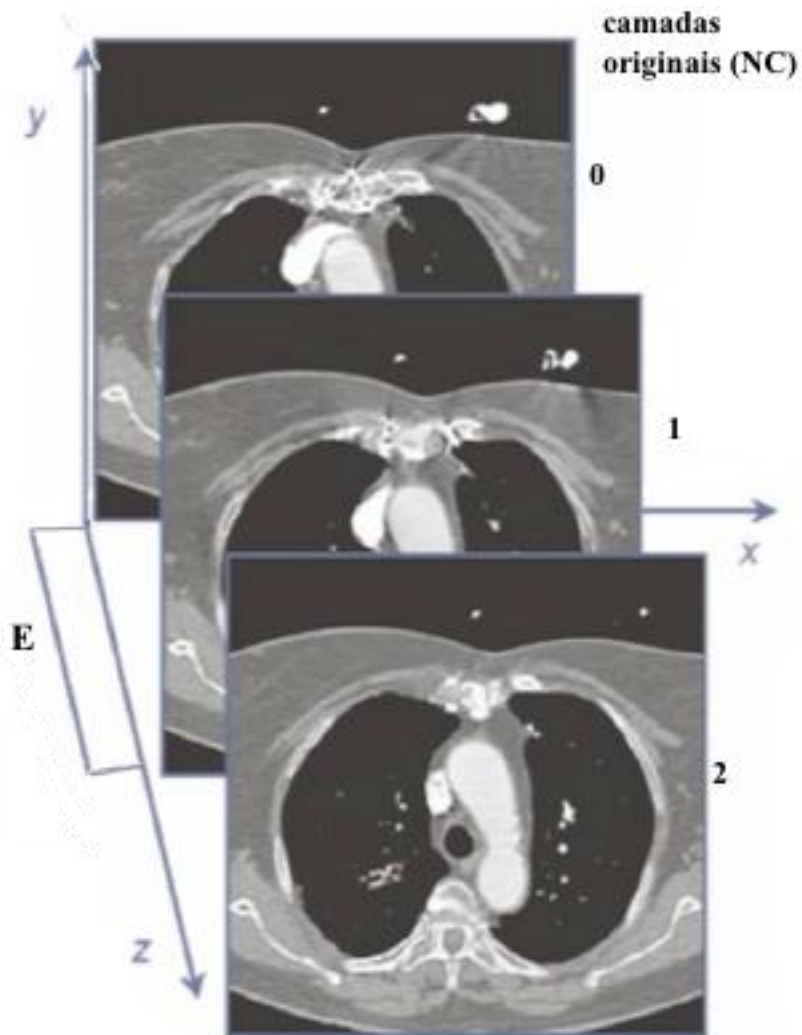


Figura 5.3: Sobreposição de camadas [3].

Os processo de interpolação em aritmética incremental consiste em pré-calcular uma matriz, de mesmo tamanho que as camadas originais. Os *pixels* desta matriz serão calculados pela equação 5.3.

$$I(x, y) = \frac{(f(x, y) - g(x, y))}{E} \quad (\text{eq. 5.2})$$

onde:

$f(x, y)$: imagem no início de um *gap* $g(x, y)$: imagem no final de um *gap*
 E : distância em mm entre duas imagens do exame, correspondente ao *gap* $I(x, y)$: Matriz incremento pré-calculado

Assim, fazendo $f_{i+1}(x, y) = f(x, y)$, pode-se obter a imagem da camada $i+1$ pela equação 5.3

$$f_{i+1}(x, y) = f_i(x, y) + I(x, y) \quad (\text{eq. 5.3})$$

Após a execução das *threads* a totalidade da matriz tridimensional estará preenchida com as imagens interpoladas.

5.3.3. Manipulação

Através de transformações geométricas é possível rotacionar o objeto contido na matriz tridimensional. As matrizes de transformação geométrica são aplicadas diretamente sobre essa matriz 3D.

A rotação é importante para se visualizar as diferentes faces do objeto, ela modifica a localização de cada voxel, ou seja, representação tridimensional de cada *pixel* da imagem. Desta forma torna-se possível apresentar no monitor partes da imagem que não seriam perceptíveis.

O processo de rotação ocorre sobre o eixo definido na escolha do usuário, aplicando-se a equação 5.8, o usuário escolhe se irá rotacionar a imagem horizontalmente (ao redor do eixo Y), verticalmente (ao redor do eixo X) ou sobre si mesma (ao redor do eixo Z). Já o processo de zoom ocorre de maneira mais simples; o plano de projeção é aproximado ou afastado do objeto, aumentando ou diminuindo o objeto projetado, respectivamente [16].

As equações 5.4 a 5.7 apresentam as transformações geométricas. Para obter o efeito desejado é necessário concatenar, através de multiplicações, as matrizes necessárias.

$$T(dx, dy, dz) = \begin{vmatrix} 1 & 0 & 0 & dx \\ 0 & 1 & 0 & dy \\ 0 & 0 & 1 & dz \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (\text{eq: 5.4})$$

$$R_x(\theta) = \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\text{sen} \theta & 0 \\ 0 & \text{sen} \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (\text{eq: 5.5})$$

$$R_y(\theta) = \begin{vmatrix} \cos \theta & 0 & -\text{sen} \theta & 0 \\ 0 & 1 & 0 & 0 \\ \text{sen} \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (\text{eq: 5.6})$$

$$R_z(\theta) = \begin{vmatrix} \cos \theta & -\text{sen} \theta & 0 & 0 \\ \text{sen} \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad (\text{eq: 5.7})$$

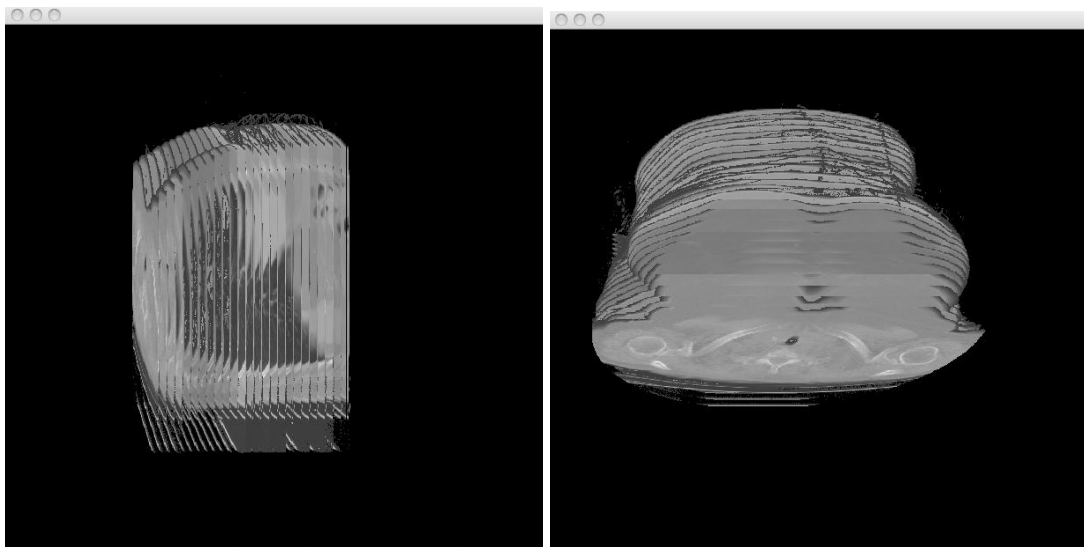
$$MTR = T(x, y, z) \times R(\theta) \times T(-x, -y, -z) \quad (\text{eq: 5.8})$$

Onde θ representa um ângulo qualquer.

As figuras 5.4 e 5.5 apresentam exemplos de rotações obtidas pelo protótipo, com e sem interpolação. A figura 5.6 mostra o resultado do zoom.

5.3.4. Projeção

A projeção é a etapa que possibilita exibir o objeto na tela do computador. Para realizar a projeção opera-se também com matrizes, assim como as transformações geométricas. Para reduzir o tempo de processamento optou-se por realizar a transformação perspectiva através da equação 5.9 e 5.10, onde devemos apenas definir a posição do observador e a distância ao plano de projeção[16].



a)

b)

Figura 5.4 Rotações sem interpolação: a) no eixo Y, b) no eixo X

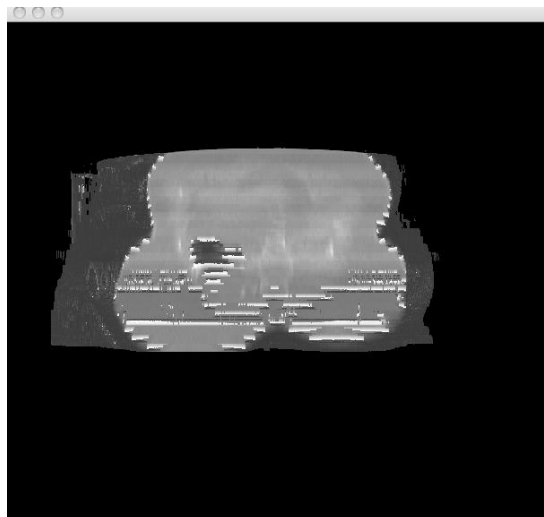
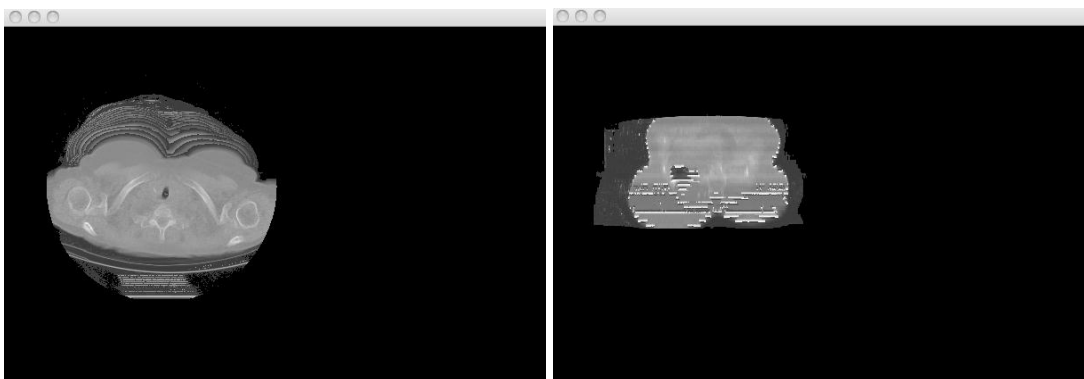


Figura 5.5 Rotação com interpolação no eixo X



a)

b)

Figura 5.6 Zoom Out obtido pelo protótipo: a) sem rotações, b) com rotação no eixo X

Adotou-se este modelo de projeção pela melhor representação computacional do mundo real; seu realismo é superior a projeção ortogonal[16]. Além disso os resultados obtidos foram considerados bons.

A figura 5.7 mostra a diferença fundamental entre os tipos de projeção ortogonal, e perspectiva ou cônica.

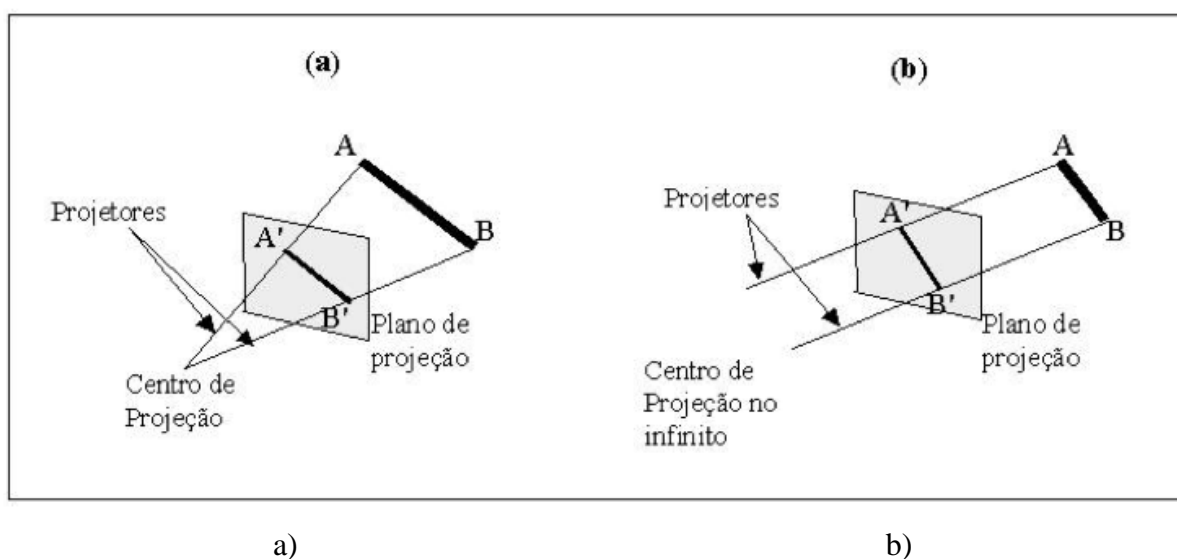


Figura 5.7 Tipos de projeção: a) Projeção perspectiva, b) Projeção ortogonal [16]

A projeção transforma os pontos do objeto de três dimensões (x, y, z) para pontos projetados em duas dimensões (x, y), sendo possível então desenhá-los na tela do computador.

$$x_p = x \left(\frac{z_{prp} - z_{vp}}{z - z_{prp}} \right) = x \left(\frac{d_p}{z - z_{prp}} \right) \quad (\text{eq. 5.9})$$

$$y_p = y \left(\frac{z_{prp} - z_{vp}}{z - z_{prp}} \right) = y \left(\frac{d_p}{z - z_{prp}} \right) \quad (\text{eq 5.10})$$

Onde Z_{prp} é o ponto no eixo Z em que o observador está localizado, Z_{vp} é o ponto no eixo Z onde o plano de projeção está localizado, d_p representa a distância do observador até o plano de projeção e as variáveis x, y, z representam as coordenadas de cada pixel do objeto.

No momento em que os cálculos de projeção são efetuados, os pixels muito escuros, são ignorados, de forma a não ocultarem as camadas inferiores, desta forma, apenas a informação importante será retratada no plano de projeção. Estes pontos escuros são os pontos que não fazem parte do corpo que se deseja ver na imagem, ou seja, os pontos que se encontram ao redor dos pontos relevantes, conforme apresentados na figura 5.8.

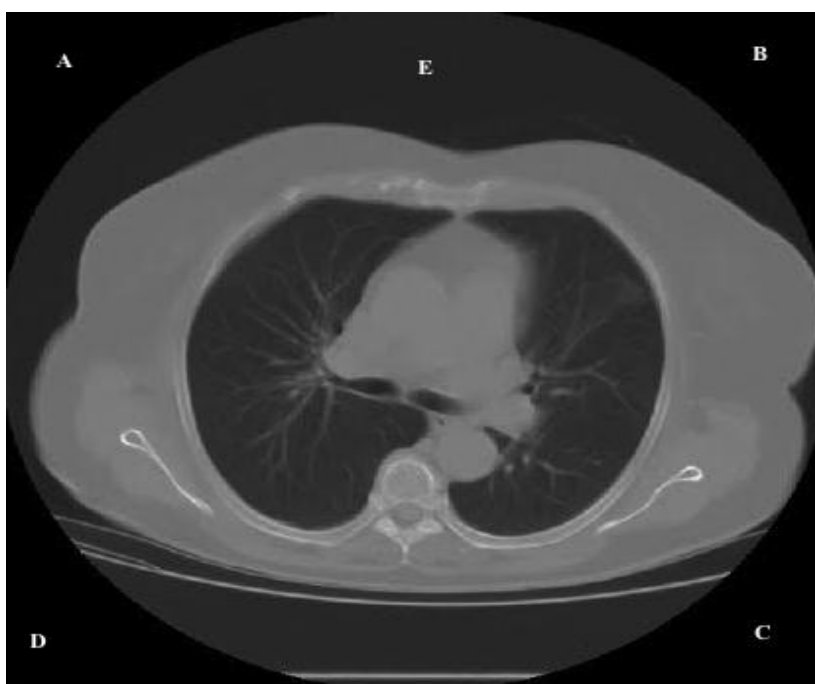


Figura 5.8: Regiões de pontos considerados escuros, regiões A, B, C, D e E são ignoradas.

5.4. Soluções Alternativas

Outras soluções alternativas foram avaliadas; algumas foram mantidas no protótipo e podem ser utilizadas caso o usuário desejar. Estas soluções apresentaram alguns problemas funcionais, temporais ou sua complexidade eram maiores que o necessário. As soluções avaliadas são apresentadas a seguir.

5.4.1. Não Interpolar o Objeto

A manipulação do objeto, sem interpolá-lo, forneceu resultados interessantes; é possível visualizar as camadas da imagem de maneira descontínua e sua execução é mais rápida que a interpolada, porém não tão precisa. Embora a interpolação traga alguns erros de arredondamento, proveniente das diferenças cromáticas significativas de acordo com o espaçamento entre camadas, sua aproximação do real é mais efetiva.

Esta solução foi mantida para o usuário que preferir esta visualização. A figura 5.9 compara os resultados da rotação sem interpolação (à esquerda) e com interpolação (à direita).

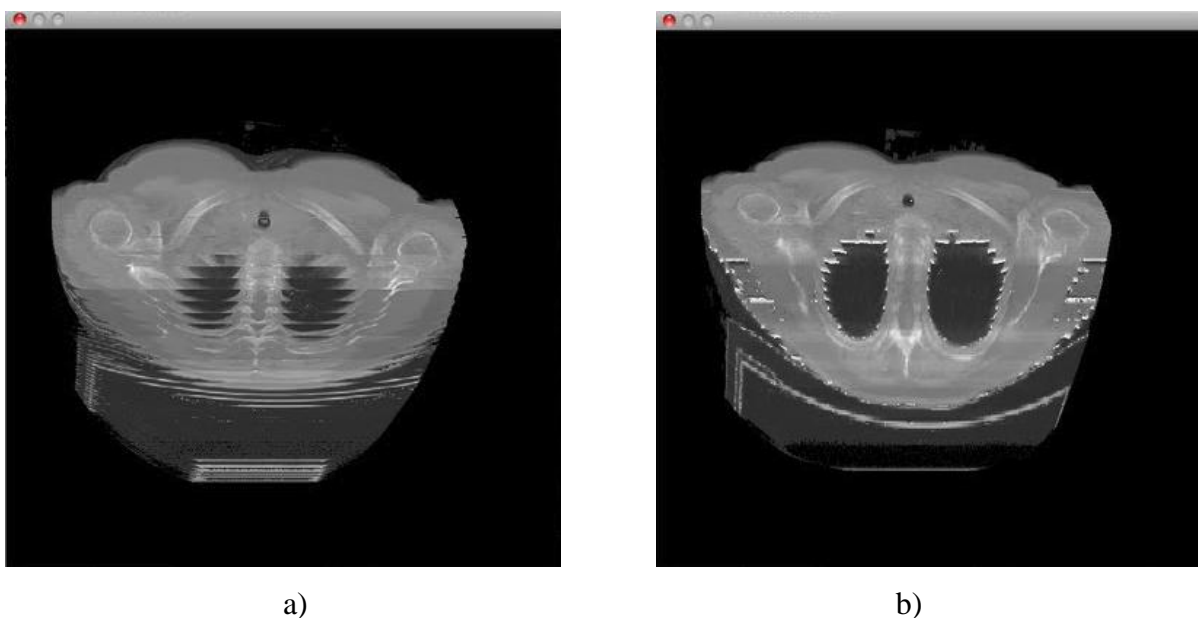


Figura 5.9 Comparação do volume 3D gerado pelo protótipo: a) sem interpolação, b) com interpolação

Nota-se que, apesar de não estar interpolada, a imagem da esquerda provê um melhor detalhamento em algumas regiões do objeto. Porém, na visão geral, a imagem da direita é a que mais se aproxima do objeto no mundo real.

5.4.2. Rotacionar o Observador

Embora rotacionar o observador ao redor do centro geométrico do objeto pareça ser mais simples do que rotacionar o próprio objeto, a complexidade é muito maior que rotacionar o objeto todo. No primeiro caso torna-se necessário transformar as coordenadas do objeto para coordenadas de câmera, concatenando diversas matrizes. Embora os cálculos possam ser feitos de uma única vez, estes tornam-se muito mais complexos. Seu desempenho é semelhante ao método onde se rotaciona o objeto.

5.4.3. Otimizações

O uso de *threads* em alguma etapas da execução, como a interpolação e a visualização em slides, melhorou significativamente o desempenho. O uso de *threads* em processadores multi-core, devido ao acesso concorrente à memória, pode-se sobrepor os dados armazenados. Neste trabalho não foi necessário controlar acesso concorrente pois as threads acessam regiões de memória exclusivas.

Como um dos objetivos deste trabalho é a portabilidade, certas técnicas não puderam ser utilizadas; as mais importantes e significativas envolvem o processamento das matrizes e equações matriciais utilizando-se as GPU (*Graphic Processing Unit*), ou seja, os processadores gráficos presentes em placas de vídeo, que possuem uma otimização estrutural para este tipo de cálculos. Porém eles se restringem a certos tipos de placas de vídeo. Atualmente as GPUs começam a ser empregadas em processadores convencionais, mas com o desempenho substancialmente menor.

Se a portabilidade não fosse essencial, seria possível otimizar o tempo de execução com APIs gráficas como o OpenGL, OpenCL ou até mesmo a utilização do CUDA (*Compute Unified Device Architecture*), que funciona apenas em placas de vídeo NVIDIA e permite a utilização dos núcleos da GPU para processamento paralelo.

5.5 Avaliação dos usuários

Três estudantes da área da saúde, dos cursos de medicina e farmácia, utilizaram o protótipo a fim de testar sua usabilidade e o resultado das imagens obtidas. Os resultados obtidos foram os seguintes.

O protótipo apresenta fácil adaptação e boa usabilidade. Os *sliders* (barras de rolagem para selecionar a rotação), apresentados na figura 5.10, são práticos e fáceis.

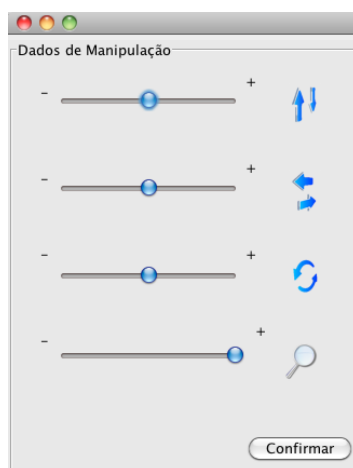


Figura 5.10: *Sliders* de rotação da imagem

O tempo de execução não foi visto como um problema sério, embora tenha sido mencionado.

Os resultados obtidos com o modelo foram satisfatórios. A preferência de visualização ficou com as imagens sem interpolação, pois oferecem certa facilidade na conexão do objeto 3D com os exames em chapas tomográficas. As imagens com interpolação foram preferidas por aqueles que já entendem os exames e gostariam de ver o objeto como um corpo real.

Capítulo 6

Conclusões

O padrão DICOM é muito mais que um formato de imagem. Trata-se de um protocolo rígido criado para padronizar todo e qualquer sistema médico, evitando a incompatibilidade entre eles. Embora empresas privadas decidam sobre o funcionamento de grande parte de seu armazenamento e transmissão, ao atender o padrão DICOM, garante-se a compatibilidade.

Com o sistema proposto estudantes e interessados em utilizar imagens tomográficas poderão, gratuitamente e em qualquer computador, visualizá-las de forma fácil e prática.

As *threads* proporcionam redução significativa no tempo consumido na interpolação das imagens. O processo realizado em 4 segundos passou a consumir menos de 1 segundo utilizando os dois núcleos e 24 threads. Com as condições propostas, e mantendo a portabilidade, os resultados foram satisfatórios, porém uma melhora no tempo de execução seria desejável o software ser utilizado extensivamente.

Os resultados obtidos foram semelhantes aos sistemas computacionais existentes, porém a portabilidade foi comprovada. A execução das operações foi mais lenta que nos softwares testados, visto que não foram utilizados aceleradores gráficos.

De modo geral, o software atendeu as expectativas e funcionalidades propostas. Levando em consideração a memória principal, realmente é necessário uma quantidade considerável de memória para armazenar e manipular o objeto, visto que não existe o que fazer além de armazená-lo completamente em memória principal para manipulá-lo. Nos testes executados o consumo de memória atingiu o pico de cerca de 300MB para um exame de apenas 25 camadas. Para os computadores atuais esta quantidade não inviabiliza a utilização do protótipo desenvolvido.

6.1 Trabalhos Futuros

Como trabalhos futuros têm-se as seguintes sugestões:

- a) otimizar e expandir este sistema para outros tipos de exames;

- b) Utilizar tecnologias, como o CUDA, para melhorar o desempenho e, com algum incremento na complexidade da programação, manter a portabilidade, ainda que parcial do sistema.

- c) Otimizar o sistema visual e funcionalmente, a fim de produzir uma versão com aprovação da ANVISA e que possa ser utilizada comercialmente;

- d) Auxiliar no desenvolvimento de trabalhos semelhantes, que venham a ser realizados por estudantes de Ciência da Computação, de forma a enriquecer o campo de pesquisa em imagens médicas, que é muito pouco difundido em nossa região.

- e) Desenvolver o modelo completo de visualização proposto por Alvy-Ray Smith. Este modelo para visualização de objetos 3D, apesar de realizar cálculos mais complexos pode ser implementado utilizando mecanismos de aceleração gráfica.

Referências Bibliográficas

- [1] BANDEIRA, Rafael; OLIVEIRA, Lucas. Desenvolvimento de uma ferramenta para reconstrução tridimensional de imagens de SPECT cardíaco In. XVIII CIC - XI ENPOS - I MOSTRA CIENTÍFICA. 2009.
- [2] ANVISA – Agência Nacional de Vigilância Sanitária. Certificado de Boas Práticas: Disponível em: <http://portal.anvisa.gov.br/wps/portal/anvisa/home/produtossaude?cat=Boas+Praticas&cat1=com.ibm.workplace.wcm.api.WCM_Category/Boas+Praticas/a7236f004f6bfd2baf3bfc894994279/PUBLISHED&con=com.ibm.workplace.wcm.api.WCM_Content/Boas+Praticas/a6aad9804098da2ca6c7ee7232b7a538/PUBLISHED&showForm=no&siteArea=Produtos+para+a+Saude&WCM_GLOBAL_CONTEXT=/wps/wcm/connect/anvisa/Anvisa/Inicio/Produtos+para+a+Saude/Publicacao+Produtos+para+a+Saude/Boas+Praticas> Acessado em: Julho de 2011.
- [3] PIANYKH, Oleg. Digital Imaging and Communications in Medicine (DICOM). Boston Ed. Springer, 2008, 383p
- [4] LEADTOOLS - The World Leader in Imaging SDKs Disponível em: <<http://www.leadtools.com/SDK/medical/dicom-spec1.htm>>. Acessado em: Julho de 2011.
- [5] SILVA, Ronaldo; SAMPAIO, Milton. Padrão de comunicação de imagens médicas DICOM v3.0. Disponível em: <<http://pt.scribd.com/doc/9692832/Padrao-DICOM>>. Acessado em: Julho de 2011.
- [6] FICEL, Marcelo. SAIBA O QUE É PACS E DICOM: Disponível em: <http://www.tecnologiaradiologica.com/materia_pacs_dicom.htm>. Acessado em: Julho de 2011.
- [7] Prof. Maurício. Diagnóstico por Imagem Guia de Auto - Tomografia Computadorizada – Histórico, aspectos físicos e princípios da formação da imagem - Uni Sant'Anna. Disponível em: <<http://igor.triunfus.com.br/santanna/ddi/aula06/aula06.pdf>>. Acessado em: Julho de 2011
- [8] JUNIOR, Edson; YAMASHITA Helio. Aspectos básicos de tomografia computadorizada e ressonância magnética. Revista Brasileira de Psiquiatria, volume.23 suppl.1 São Paulo. Maio, 2001. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1516-44462001000500002>. Acessado em: Julho de 2011

- [9] KALASHE, Lucas. Sistema de Medição da Espessura da Camada Muscular e Adiposa Via Ultra-Som (Trabalho de Conclusão de Curso Engenharia da Computação) - UNICENP/NCET, Curitiba, 2007, Monografia.
- [10] OsiriX Imaging Software, Advanced Open-Source PACS Workstation DICOM Viewer About OsiriX..., Disponível em: <<http://www.osirix-viewer.com/AboutOsiriX.html>>. Acessado em: Julho de 2011
- [11] OsiriX Imaging Software, Advanced Open-Source PACS Workstation DICOM Viewer OsiriX Graphic User Interface. Disponível em: <<http://www.osirix-viewer.com/Snapshots.html>>. Acessado em: Julho de 2011.
- [12] ImageJ - Documentation, Disponível em: <<http://rsb.info.nih.gov/ij/docs/index.html>>. Acessado em: Julho de 2011.
- [13] Adobe Photoshop, Sobre os arquivos DICOM (Photoshop Extended): Disponível em: <http://help.adobe.com/pt_BR/Photoshop/11.0/WSC66C38A0-928A-46c7-9236-41F1A0136128a.html>. Acessado em: Julho de 2011.
- [14] MicroDicom, MicroDicom Viewer. Disponível em: <<http://www.microdicom.com/dicom-viewer/overview.html>>. Acessado em: Julho de 2011.
- [15] InVesalius, Sobre o Software. Disponível em: <http://www.softwarepublico.gov.br/dotlrn/clubs/invesalius/one-community?page_num=0>. Acessado em: Outubro de 2011.
- [16] AZEVEDO, E.; CONCI, A. Computação gráfica: teoria e prática. Rio de Janeiro : Campus, 2003. v.1, 384 p.