

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

**Reconstrução Tridimensional de Objetos
a partir das Projeções de suas Secções**

por

PATRÍCIA NOLL DE MATTOS

Dissertação submetida à avaliação,
como requisito parcial para a obtenção do grau de Mestre
em Ciência da Computação

Prof. M.Sc. Anatólio Laschuk
Orientador

Porto Alegre, janeiro de 1996.



UFRGS
Instituto de Informática
Biblioteca

CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Mattos, Patrícia Noll de

Reconstrução Tridimensional de Objetos a partir das Projeções de suas Secções / por Patrícia Noll de Mattos. - Porto Alegre: CPGCC da UFRGS, 1996.

78 f.: il.

Dissertação (mestrado) - Universidade Federal do Rio Grande do Sul, Curso de Pós-Graduação em Ciência da Computação, Porto Alegre, BR-RS, 1996. Orientador: Laschuk, Anatólio.

1. Processamento gráfico. 2. Tomografia computadorizada. 3. Reconstrução tridimensional. 4. Projeções. I. Laschuk, Anatólio. II. Título.

UFRGS			
INSTITUTO DE INFORMÁTICA			
BIBLIOTECA			
Nº CHAMADA:		Nº REE:	
61:621.32(043)		41930	
M435n		DATA:	
		08/08/03	
ORIGEM:	DATA:	VALOR:	
1	20/07/03	R\$ 30,00	
FUNDO:	FORN.:		
II	II		

Aplicação dos computadores em Informática médica
Processamento: Image
Diagnósticos médicos
Tomografia; Informática médica

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
 Reitor: Prof. Hélgio Trindade
 Pró-Reitor de Pesquisa e Pós-Graduação: Prof. Cláudio Scherer
 Diretor do Instituto de Informática: Prof. Roberto Tom Price
 Coordenador do CPGCC: Prof. José Antônio Palazzo de Oliveira
 Bibliotecária – Chefe do Instituto de Informática: Zita Prates de Oliveira

CNPq 1.03.04.00-2

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Sistema de Bibliotecas da UFRGS

INF 41830
61:681.32(043) M435r

05315540

[0373341] Mattos, Patrícia Noll de.
Reconstrução tridimensional de objetos a partir
das projeções de suas secções. 1996. 78 f. :
il.

Eduquem os meninos e não será preciso castigar os homens!

Pitágoras

Agradecimentos

Gostaria de agradecer a todos que, de alguma forma, contribuíram para a realização deste trabalho. Em especial, ao professor Laschuk, por sua disponibilidade, apoio e confiança, e pelas várias explicações matemáticas que, muitas vezes, se fizeram necessárias.

Gostaria de agradecer ao Luciano Castro, pelo apoio, companheirismo e disponibilidade que me dedicou durante este trabalho.

Sumário

Lista de Abreviaturas	7
Lista de Figuras	8
Resumo	9
Abstract	10
1 Introdução	11
2 Tomografia Computadorizada	12
2.1 Estado da Arte	12
2.1.1 Tipos de Tomografia Computadorizada	13
2.1.1.1 Tomografia Computadorizada por Transmissão (TCT)	13
2.1.1.2 Tomografia Computadorizada por Emissão (TCE)	16
2.1.2 Projeções	17
2.1.3 Reconstrução Bidimensional	21
2.1.3.1 Técnica de Fourier	22
2.1.3.2 Método da Convolução	24
2.1.3.3 Método das Projeções	25
2.2 Resumo	27
3 Proposta de Trabalho 2D	28
3.1 Modelagem dos Objetos	28
3.2 Simulação do Processo de Tomografia Computadorizada	29
3.2.1 Método de Reconstrução Bidimensional Utilizado por Este Trabalho	29
3.2.1.1 O Método	30
3.2.1.2 Refinamento	35
3.2.2 Algoritmo de Intersecção	37
3.2.3 Algoritmo de Recorte	39
3.2.4 Aquisição das Imagens Projetadas	42
3.2.5 Algoritmo de Reconstrução	47
3.3 Resumo	50
4 Proposta de Trabalho 3D	52
4.1 Modelagem dos Objetos	52
4.2 Simulação do Processo de Tomografia Computadorizada	53
4.2.1 Algoritmo de Secção Tridimensional	53
4.2.2 Aquisição das Imagens Projetadas	56
4.2.3 Algoritmo de Reconstrução	59
4.3 Resumo	62
5 Análise dos Resultados	64
5.1 Etapa Bidimensional	64
5.2 Etapa Tridimensional	66
6 Conclusões	67
Anexo 1 Geração de Projeções	69
Anexo 2 Teorema do Método das Projeções	71

Anexo 3 Resultados Obtidos	74
Bibliografía Referenciada	76
Bibliografía Consultada	77

Lista de Abreviaturas

- AIP_2D Aquisição das Imagens Bidimensionais Projetadas.
- AIP_3D Aquisição das Imagens Tridimensionais Projetadas.
- TCE Tomografia Computadorizada por Emissão.
- FFT Fast Fourier Transform.
- patch Porção delimitada de uma superfície.
- PET Positron-Emission Tomography.
- pixel Picture element (cada um dos elementos da matriz de pontos de um monitor de vídeo).
- SPECT Single-Photon Emission Tomography.
- SPTC Simulação do Processo de Tomografia Computadorizada.
- TC Tomografia Computadorizada.
- TCT Tomografia Computadorizada por Transmissão
- voxel Volume element (análogo tridimensional do pixel).

Lista de Figuras

FIGURA 2.1	- Sombras de raio-x, em diferentes ângulos	13
FIGURA 2.2	- Propagação de um feixe paralelo de raio-x através de uma secção do corpo humano [ROS 82]	15
FIGURA 2.3	- Secção do corpo humano com uma fonte de raios-gama distribuída [ROS 82]	16
FIGURA 2.4	- Determinação da atenuação dos raios do feixe	18
FIGURA 2.5	- Projeções paralelas [ROS 82]	19
FIGURA 2.6	- Projeções <i>fan-beam</i> [ROS 82]	20
FIGURA 2.7	- Processo de reconstrução bidimensional através de projeções	21
FIGURA 2.8	- Teorema da Fatia de Fourier [ROS 82]	23
FIGURA 2.9	- Projeções levadas a diferentes ângulos [ROS 82]	23
FIGURA 3.1	- Organograma da Proposta 2D	28
FIGURA 3.2	- Representação matricial do objeto	29
FIGURA 3.3	- Método de reconstrução bidimensional [ROS 82]	30
FIGURA 3.4	- Método de Kackzmarz [ROS 82]	33
FIGURA 3.5	- Processo incremental de intersecção	37
FIGURA 3.6	- Intersecção da aresta do polígono com as arestas limites	40
FIGURA 3.7	- Recorte do polígono	41
FIGURA 3.8	- Grade de raios	43
FIGURA 3.9	- Raio-soma	44
FIGURA 3.10	- Peso da porção da face para a célula	45
FIGURA 3.11	- Absorção da célula	45
FIGURA 3.12	- Cálculo da área do polígono interno à célula	46
FIGURA 3.13	- Seleção de porções das faces	48
FIGURA 4.1	- Organograma da Proposta 3D	52
FIGURA 4.2	- Objeto tridimensional modelado	53
FIGURA 4.3	- Tamanho da fonte de raio-x	54
FIGURA 4.4	- Recorte das porções das faces interceptadas pelo plano em z	54
FIGURA 4.5	- Plano de corte	56
FIGURA 4.6	- Inteseccção das porções das faces com o plano de corte	57
FIGURA 4.7	- Porções das faces da fatia interceptadas pelo raio 1	57
FIGURA 4.8	- Estrutura para o armazenamento das projeções	58
FIGURA 4.9	- Grade de raios sobre fatia do objeto	60
FIGURA 5.1	- Resultados	66
FIGURA A1.1	- Projeção $P_{\theta}(t)$ de $f(x,y)$ no ângulo θ [ROS 82]	70
FIGURA A2.1	- Hiperplano $\vec{W}_1 \vec{f} = P_1$ [ROS 82]	71
FIGURA A3.1	- Imagem 5x5, onde foram executados cinco ciclos	74
FIGURA A3.2	- Imagem 5x5, onde foram executados oito ciclos. A imagem foi alterada por ruído	74
FIGURA A3.3	- Imagem 5x5 reconstruída no primeiro ciclo, devido a algum conhecimento prévio da mesma	75
FIGURA A3.4	- Imagem 5x5, onde foram executados cinco ciclos	75
FIGURA A3.5	- Imagem 5x5, onde foram executados cinco ciclos. Densidades variadas	75

Resumo

Este trabalho trata da reconstrução tomográfica bidimensional e tridimensional, a partir de projeções, visando sua utilização em qualquer aplicação que necessite do volume completo do objeto em questão ou que necessite das imagens adquiridas dos objetos projetados.

Propõe-se utilizar um dos métodos pelo qual a tomografia computadorizada gera as fatias bidimensionais do corpo exposto, expandindo-o para a terceira dimensão. Para tornar isto possível, é necessário o estudo de métodos de reconstrução bidimensional, bem como, a implementação de um deles.

É desenvolvido, então, um estudo sobre os conceitos da tomografia computadorizada, seus diferentes tipos, contendo, os métodos de reconstrução bidimensional utilizado por elas, além, dos métodos de aquisição das projeções dos objetos a serem reconstruídos.

É apresentada uma proposta de trabalho bidimensional, onde será implementado, além do algoritmo de reconstrução, o processo de aquisição das projeções do objeto em questão, bem como, a modelagem do mesmo. Além, do desenvolvimento de uma proposta de trabalho tridimensional, onde o método de reconstrução bidimensional desenvolvido será expandido para a terceira dimensão e implementado, juntamente com o processo de aquisição dos objetos tridimensionais projetados, bem como, a modelagem dos mesmos.

Um protótipo para cada uma destas propostas é gerado, contendo, cada um, a etapa de aquisição das imagens projetadas, e a etapa de reconstrução do objeto. Tanto os resultados da imagem sendo reconstruída, como as imagens adquiridas dos objetos projetados, são gravados em arquivos.

Por fim, é apresentada uma análise dos resultados gerados por estes protótipos, no que diz respeito à convergência de seus algoritmos, e a melhor maneira de utilizá-los, a fim de se obter um melhor resultado.

Palavras-chave: Tomografia Computadorizada, Reconstrução Tridimensional, Projeções.

TITLE: "TRIDIMENSIONAL OBJECTS RECONSTRUCTION FROM THE PROJECTIONS OF THEIRS SECTIONS"

Abstract

This paper is about bidimensional and tridimensional tomographic reconstruction working with projections in order to use in any application that needs complete volume of the object used or that needs images of the projected objects acquired.

It is used one of the methods by which the computerized tomography generates bidimensional slices of the exposed body expanding it to third dimension. In order to make it possible it was necessary to study bidimensional reconstruction methods, as well as implementing one of them.

It is developed a study on concepts of computerized tomography, its different types, including bidimensional reconstruction methods used by them, besides the methods used to acquire projections of the objects to be reconstructed.

It is presented a proposal of the bidimensional work, where it will be implemented, besides the reconstruction algorithm, the aquisition process of the projection of the meant objects, as well as modelling them. It is also developed a proposal of a tridimensional work, where the bidimensional reconstruction method developed will be implemented and expanded to third dimension as well as the process of aquisition of the tridimensionals projected objects as well as modelling them.

A prototype of these proposals is generated , containing each one the aquisition step of the projected images and the reconstruction step of the object. The results of the reconstructed image as well as the acquired images of the projected objects are saved in files.

Finally it is presented an analysis of the generated results of the prototypes as to the convergence of these algorithms and the best way to use them in order to obtain a better result.

Keywords: Computerized Tomography, Tridimensional Reconstruction, Projections.

1 Introdução

Esta dissertação trata da reconstrução de imagens tomográficas, tanto bidimensionais, quanto tridimensionais.

A imagem gerada por um tomógrafo é de natureza bidimensional, porém, as estruturas anatômicas dos órgãos são inerentemente tridimensionais, o que sugere a reconstrução do volume completo do objeto. Várias técnicas têm sido usadas para se visualizar um objeto tridimensional, a partir de várias de suas secções bidimensionais.

Porém, pode-se utilizar o mesmo processo que a tomografia computadorizada utiliza, para gerar uma coleção de imagens bidimensionais do corpo humano, a fim de se gerar o volume como um todo. Dessa forma, não se estará utilizando uma representação visual tridimensional do objeto, como nas técnicas comentadas acima, mas se estará gerando o objeto tridimensional de fato. Serão necessárias, porém, técnicas de visualização, para que se explore, da melhor maneira possível, o volume gerado.

O capítulo dois trata dos conceitos da tomografia computadorizada bidimensional, contendo: tipos de tomografia computadorizada, geração de projeções e os métodos de reconstrução bidimensional

O capítulo três descreve a proposta de trabalho bidimensional, incluindo a modelagem dos objetos bidimensionais e a simulação do processo de tomografia computadorizada, contendo a aquisição das projeções e suas posteriores reconstruções. São descritos também alguns algoritmos utilizados por esta proposta.

O capítulo quatro trata da proposta de trabalho tridimensional, onde uma dos métodos de reconstrução tomográfica bidimensional é expandido para a terceira dimensão. É descrito o processo adotado para a implementação do mesmo, bem como, para a simulação da tomografia computadorizada tridimensional, a qual é composta pelo processo de aquisição de imagens projetadas, bem como, pelo processo de reconstrução tridimensional.

O capítulo cinco trata da análise dos resultados, onde são feitas considerações importantes a respeito dos algoritmos desenvolvidos, e dos seus resultados.

As conclusões se encontram no capítulo seis, contendo alguns comentários a respeito do trabalho realizado, e apresentando algumas sugestões de procedimentos a serem realizados no futuro.

No capítulo sete, são apresentadas as referências bibliográficas, e, por fim, o anexo A-1 apresenta um aprofundamento matemático na geração das projeções, o anexo A-2 apresenta um teorema matemático que prova o métodos de reconstrução bidimensional utilizado por este trabalho, apresentado no capítulo três e o anexo A-3 apresenta alguns resultados gerados.

2 Tomografia Computadorizada

2.1 Estado da Arte

A Tomografia Computadorizada (TC) veio reduzir acentuadamente a deficiência que os diagnósticos radiológicos convencionais possuíam para distinguir estruturas com valores de absorção muito próximos entre si, como os tecidos leves (músculos, órgãos, sangue e outros).

A essência dos diagnósticos radiológicos convencionais é explorar a variação da transmissão de raio-x por diferentes substâncias, na criação de imagens que refletem o formato das estruturas dentro do corpo. Um tecido é diferenciado dos demais através de sua absorção, isto é, da intensidade de absorção que possui.

Neste processo, apenas uma única sombra de raio-x da parte de interesse é projetada no filme, o que limita a capacidade de distinguir diferentes tecidos. Tecidos adjacentes, cujas absorções são significativamente diferentes, são claramente distinguíveis. Deste modo, estruturas como ossos, que absorvem mais raio-x que outros tecidos, assim como estruturas normais ou anormais do pulmão, cercadas por ar, são facilmente reconhecidas. Porém, as regiões cujas absorções de raio-x são similares, tais como a maioria dos tecidos leves do corpo, não são bem distinguíveis por este processo. Outra limitação destes diagnósticos radiológicos convencionais é a superposição das imagens de estruturas encobertas, diminuindo a capacidade de identificar estruturas normais e anormais.

Para melhorar este processo, foram desenvolvidas várias substâncias de contraste, contendo um alto número atômico. Elas são utilizadas para aumentar artificialmente a diferença de absorção de certas estruturas e dos tecidos que os cercam. Porém, estes agentes podem ser arriscados para o paciente, além de produzirem, muitas vezes, imagens difíceis de serem interpretadas, para muitos tipos de doenças.

Com a introdução da Tomografia Computadorizada, foi possível a distinção de regiões adjacentes, com poucas diferenças na absorção de raio-x, o que a torna um notável avanço em relação aos diagnósticos radiológicos convencionais, os quais podem apenas distinguir três diferentes tipos de absorção: ar, tecidos leves e osso [AXE 83].

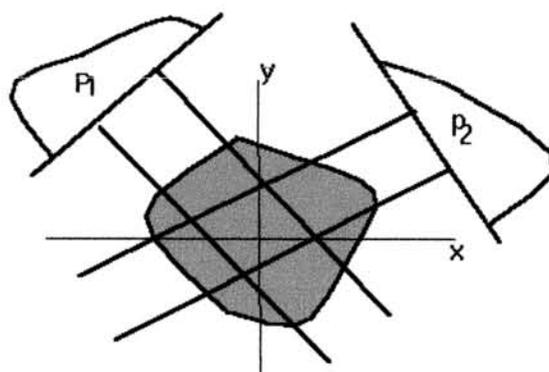


FIGURA 2.1 - Sombras de raio-x, em diferentes ângulos.

Este avanço tornou-se possível porque obtêm-se não apenas uma sombra de raio-x, mas várias, em diversos ângulos (no caso de TC por transmissão), como ilustrado na figura 2.1, o que permite também a separação de estruturas encobertas, possibilitando que muitas doenças que, até então, eram difíceis de diagnosticar, passem a ser demonstradas claramente por TC.

Dessa forma, os pacientes não necessitam mais passar por procedimentos que lhes poderiam ser perigosos ou desconfortáveis para a obtenção de um diagnóstico. A imagem da região doente pode ser facilmente obtida ao expô-la à fonte de raio-x ou gama (de acordo com o tipo de TC).

A utilização da Tomografia Computadorizada não se limita à Medicina, sua utilização se estende a outras áreas como a Engenharia, a Astronomia, ou qualquer outra que necessite da visualização total de um objeto [POG 77].

2.1.1 Tipos de Tomografia Computadorizada

Uma imagem bidimensional é obtida por Tomografia Computadorizada através da exposição da região de interesse a raios direcionados diversamente, cujas informações representam a soma das absorções das estruturas atravessadas por eles. Estas informações são chamadas de raios-soma [VIA 90], ou projeções. Diversas projeções, em diversas direções, são levadas a um algoritmo de reconstrução, a fim de se obter a imagem bidimensional correspondente da região de interesse.

O processo descrito acima pode ocorrer através da transmissão de raio-x ou por emissão de raios-gama, além de outros.

2.1.1.1 Tomografia Computadorizada por Transmissão (TCT)

Os diversos raios direcionais comentados acima são levados a diferentes ângulos, gerando-se, em cada um, a projeção do objeto no ângulo atravessado. Também, como dito anteriormente, cada raio projetado representa a atenuação total de

raio-x nas estruturas percorridas por ele. Logo, cada projeção representa a atenuação total de raio-x no ângulo em questão.

Assim, pode-se dizer que a obtenção de uma imagem bidimensional por TC dá-se a partir de várias de suas projeções unidimensionais. Cada projeção pode ser vista como a atenuação total dos raios-x, em determinado ângulo.

Para se obter a imagem de uma secção do corpo humano, por esta técnica, é necessária uma série de varreduras com um feixe de raio-x. Este feixe atravessa o plano de corte (fatia) do paciente em diferentes ângulos, gerando várias projeções unidimensionais, a partir da detecção das diferentes intensidades de raio-x em cada ponto. Após tais projeções serem geradas, um algoritmo de reconstrução bidimensional é utilizado para gerar a imagem do plano do paciente [VIA 90].

O feixe de raio-x é projetado no corpo do paciente, nos diferentes ângulos, dependendo da posição de sua fonte. Detectores de raio-x percebem a existência de raios na região de interesse do paciente. A saída do detector é digitalizada, e levada ao computador para que a imagem seja gerada. Enquanto o paciente permanecer parado, a fonte de raio-x e o detector se movem gradativamente ao redor do paciente, de modo a obter um conjunto de projeções em diferentes direções. Segundo Axel *et al.* [AXE 83], este movimento dá-se em cerca de 180 passos, correspondendo a 180 graus. O conjunto completo de projeções é então utilizado pelo computador, com um algoritmo de reconstrução, para gerar uma imagem digital da fatia irradiada. Esta imagem digital consiste em uma matriz de números. Os números computados dependem do total de absorção de raio-x pelos tecidos, correspondendo, em cada localização, à intensidade de absorção de raio-x.

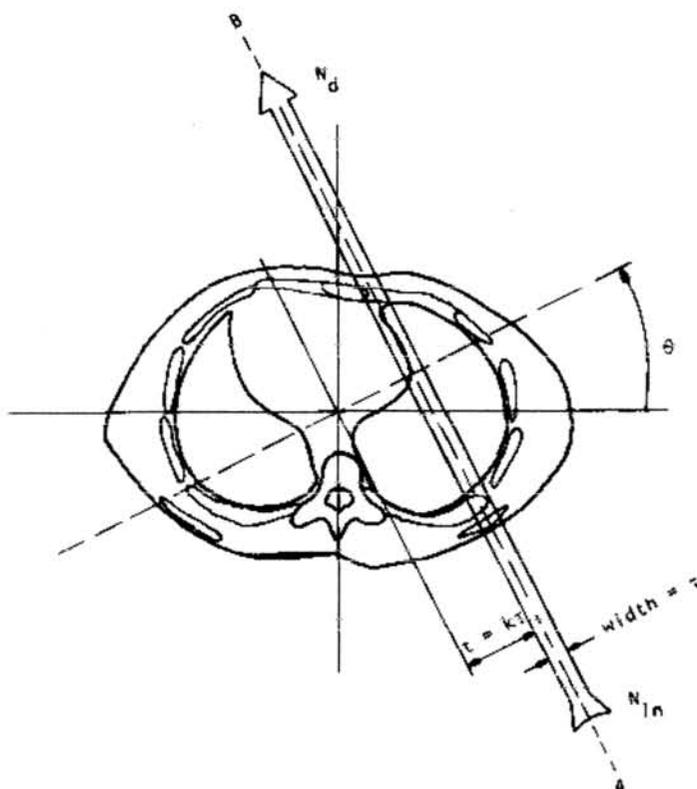


FIGURA 2.2 - Propagação de um feixe paralelo de raio-x através de uma secção do corpo humano [ROS82].

Na figura 2.2 é ilustrado um feixe paralelo de raio-x, atravessando a imagem de um plano do paciente. Cada raio do feixe é composto por uma série de fótons de raio-x, os quais podem ser perdidos, por espalhamento ou absorção, sendo, portanto, considerados apenas os fótons que são propagados na direção representada pela flecha. Esta perda pode ser representada pelo relacionamento entre o número de fótons incidentes à medição e o número deles que atravessa a região, chegando ao outro lado da imagem, sendo percebidos pelo detector. Esta atenuação nos fótons representa a absorção de raio-x nesta região. A cada ponto, de cada raio, ambas as perdas são consideradas por uma constante. A integração destas constantes, para todos os raios do feixe ao longo do plano de corte, representa a projeção da sombra de raio-x no mesmo, na direção em questão. Mais detalhes podem ser encontrados em Mattos [MAT 93] ou em Rosenfeld *et al.* [ROS 82].

As varreduras de raio-x podem ser feitas através de um sistema de fonte paralela (colimado) ou por um sistema de feixe divergente e mais de um detector [VIA 90].

2.1.1.2 Tomografia Computadorizada por Emissão (TCE)

Nesta técnica, a geração da imagem de uma secção do corpo é feita a partir de um isótopo radioativo que foi previamente distribuído e absorvido por um dado órgão. Um isótopo, específico para cada tipo de órgão, pode ser administrado a um paciente em forma de radiofármacos, tanto por injeção, como por inalação. Os isótopos radioativos são caracterizados pela emissão de raios-gama, através do seu decaimento nuclear. Do ponto de vista prático, os fótons de raio-gama são indistinguíveis dos de raio-x, suas origens é que lhes dão distinção de nomes [ROS 82]. A concentração de um isótopo, em qualquer secção, muda com o tempo, devido ao decaimento radioativo e à cinética bioquímica do corpo, o que implica que todos os dados para uma imagem seccional sejam coletados em um tempo constante, normalmente pequeno, associado às mudanças de concentração. Apesar disso, este aspecto dá à Tomografia Computadorizada por Emissão maior potencial e utilidade em diagnósticos médicos, uma vez que, ao analisar diversas vezes uma imagem da mesma secção, pode-se determinar o estado funcional de diversos órgãos do corpo do paciente.

As técnicas de Tomografia Computadorizada por Emissão são divididas em PET (Positron-Emission Tomography) e SPECT (Single-Photon Emission Tomography) [KNO 83].

Todos os três tipos de tomografia (PET, SPECT e TC) armazenam dados provenientes de raios direcionais utilizando-os em algoritmos de reconstrução para gerar a imagem da fatia em questão.

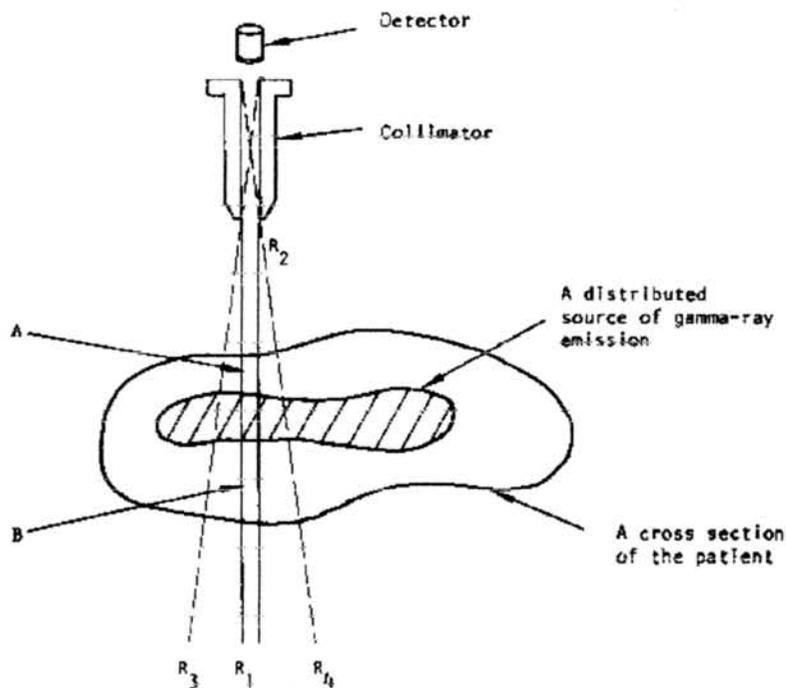


FIGURA 2.3 - Secção do corpo humano com uma fonte de raios-gama distribuída [ROS 82].

A figura 2.3 apresenta a imagem de uma secção do corpo sendo atravessada por raios de uma fonte de raios-gama distribuída. Segundo Rosenfeld *et al.* [ROS 82], o número de fótons de raios-gama emitidos por segundo, por cada elemento da fonte, é proporcional à concentração da fonte em cada ponto.

O número de fótons vindos na direção R1 e R2, detectados pelo detector, significa a integral ao longo deste raio. Ao movimentar o colimador-detector para uma posição adjacente na lateral, pode-se determinar a integral ao longo de outro raio paralelo a R1R2. Após serem feitas estas varreduras, uma projeção é gerada. Para que outras projeções sejam geradas, basta rodar o paciente ou o colimador-detector.

Um outro tipo de tomografia é a por ultrassom. Seu objetivo é similar ao da tomografia por raio-x, ou seja, constrói uma secção do corpo descrevendo seus detalhes morfológicos. O problema é a refração do ultrassom ao propagar-se através dos tecidos. Esta técnica é melhor abordada por Rosenfeld *et al.* [ROS 82].

Os dados tratados nesta dissertação são do tipo usado em Tomografia Computadorizada por Transmissão de raio-x.

2.1.2 Projeções

Como já dito anteriormente, as projeções relacionam-se com a atenuação total dos raios-x que atravessaram a região de interesse, em determinado ângulo.

De um lado do feixe está a fonte, e do outro, o detector. Cada raio que parte da fonte possui uma concentração de fótons de raio-x diferente da percebida pelo detector. Esta diferença ocorre, uma vez que os tecidos atravessados pelo raio absorvem uma determinada quantidade destes fótons, além de alguns serem perdidos. Esta diminuição na quantidade de fótons representa então a soma das diferentes quantidades de fótons absorvidas por cada tecido, dependendo de sua absorção (capacidade de absorção). Esta perda de fótons é chamada de atenuação de raio-x, para o raio em questão. Após todos os raios do feixe serem considerados, a informação que se tem é a atenuação total de raio-x do feixe naquela direção, ou seja, a soma das absorções das estruturas atravessadas por ele.

A atenuação total de raio-x, em determinada direção, é chamada de projeção, a qual é obtida computando a atenuação de cada raio do feixe. Uma maneira de se computar este valor é integrando-se as absorções, que representam a atenuação de fótons de raio-x, ao longo do raio, o que é chamado de integral do raio ou raio-soma, segundo Viana *et al.* [VIA 90]. Esta técnica é explicada no anexo A-1.

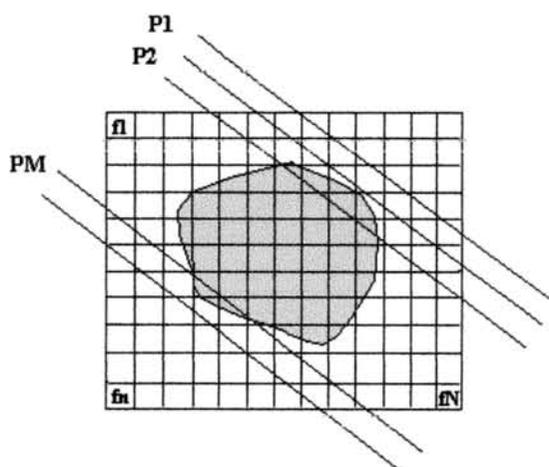


FIGURA 2.4 - Determinação da atenuação dos raios do feixe.

Outra forma de se calcular a atenuação de cada raio (raio-soma) é através do somatório das absorções das estruturas atravessadas por cada um, como ilustrado na figura 2.4. A região a ser estudada foi dividida em N (n^2) células, onde f_k é o valor indicativo da absorção da célula k . Assim sendo, a imagem passa a ser um vetor $\vec{f} = (f_1, \dots, f_k, \dots, f_N)$. Ao computar-se este valor, para todos os M raios, define-se um conjunto de M equações, a saber:

$$\begin{aligned}
 P_1 &= \sum_{k=1}^N W_{1k} f_k, \\
 P_2 &= \sum_{k=1}^N W_{2k} f_k, \\
 &\dots, \\
 P_j &= \sum_{k=1}^N W_{jk} f_k \\
 &\dots, \\
 P_M &= \sum_{k=1}^N W_{Mk} f_k
 \end{aligned} \tag{1}$$

onde M é o número de raios-soma P_j , com $j=1, \dots, M$; N é o número de células, com $k=1, \dots, N$ e W_{jk} é o peso da célula k para o raio j , ou seja, indica se o raio j passa por toda a célula k , por uma parte, ou não a atravessa.

Na aquisição das imagens projetadas, em 3.2.4, ocorre a aquisição dos M raios-soma, segundo o método expresso acima.

As projeções podem ser levadas ao longo de um conjunto de raios paralelos, chamadas de projeções paralelas. É possível também levá-las ao longo de raios que emanam de um ponto origem, chamadas de projeções com feixe em leque (*fan-beam*). As figuras 2.5 e 2.6 expressam, respectivamente, estes dois tipos de projeções.

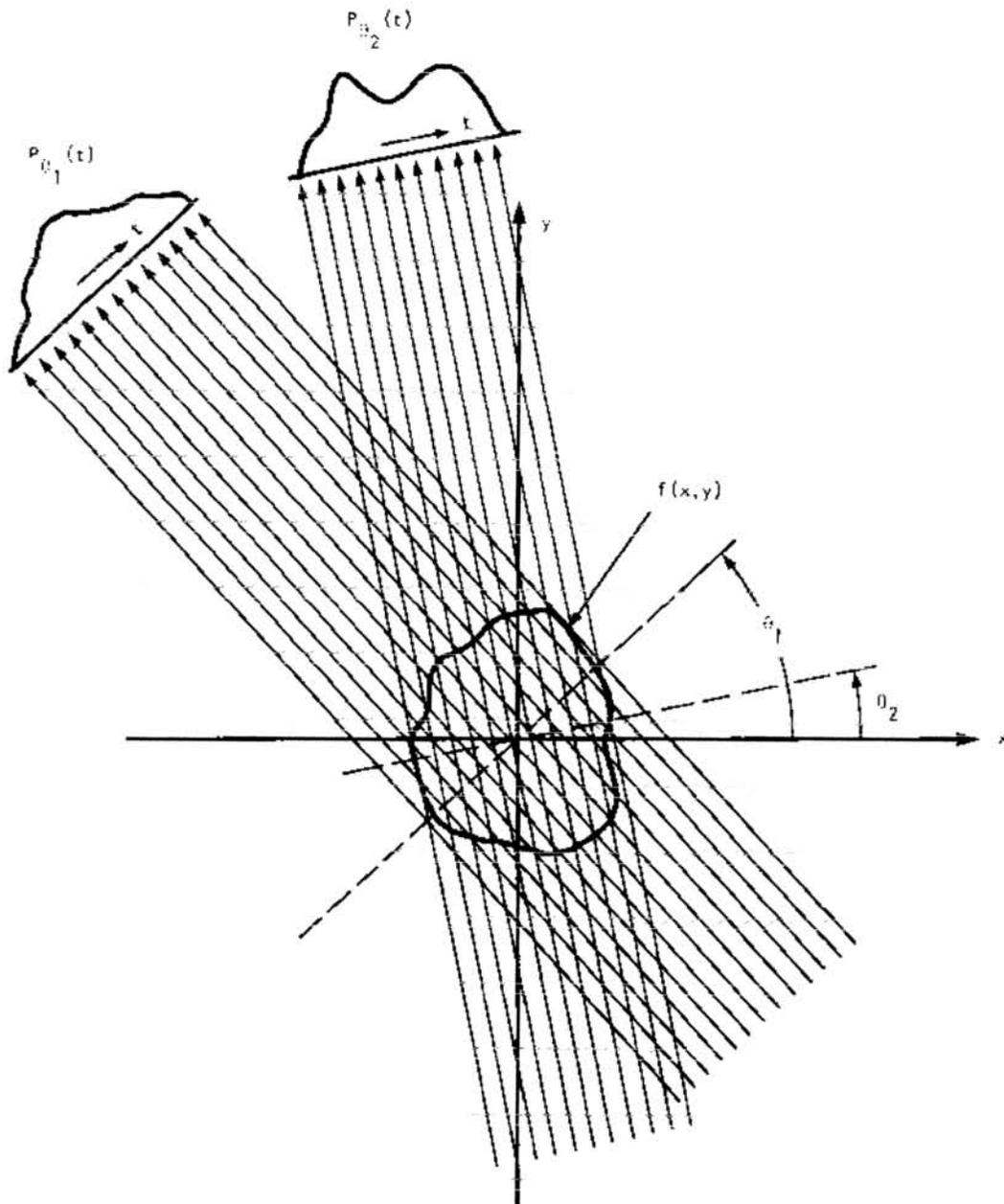


FIGURA 2.5 - Projeções paralelas [ROS 82].

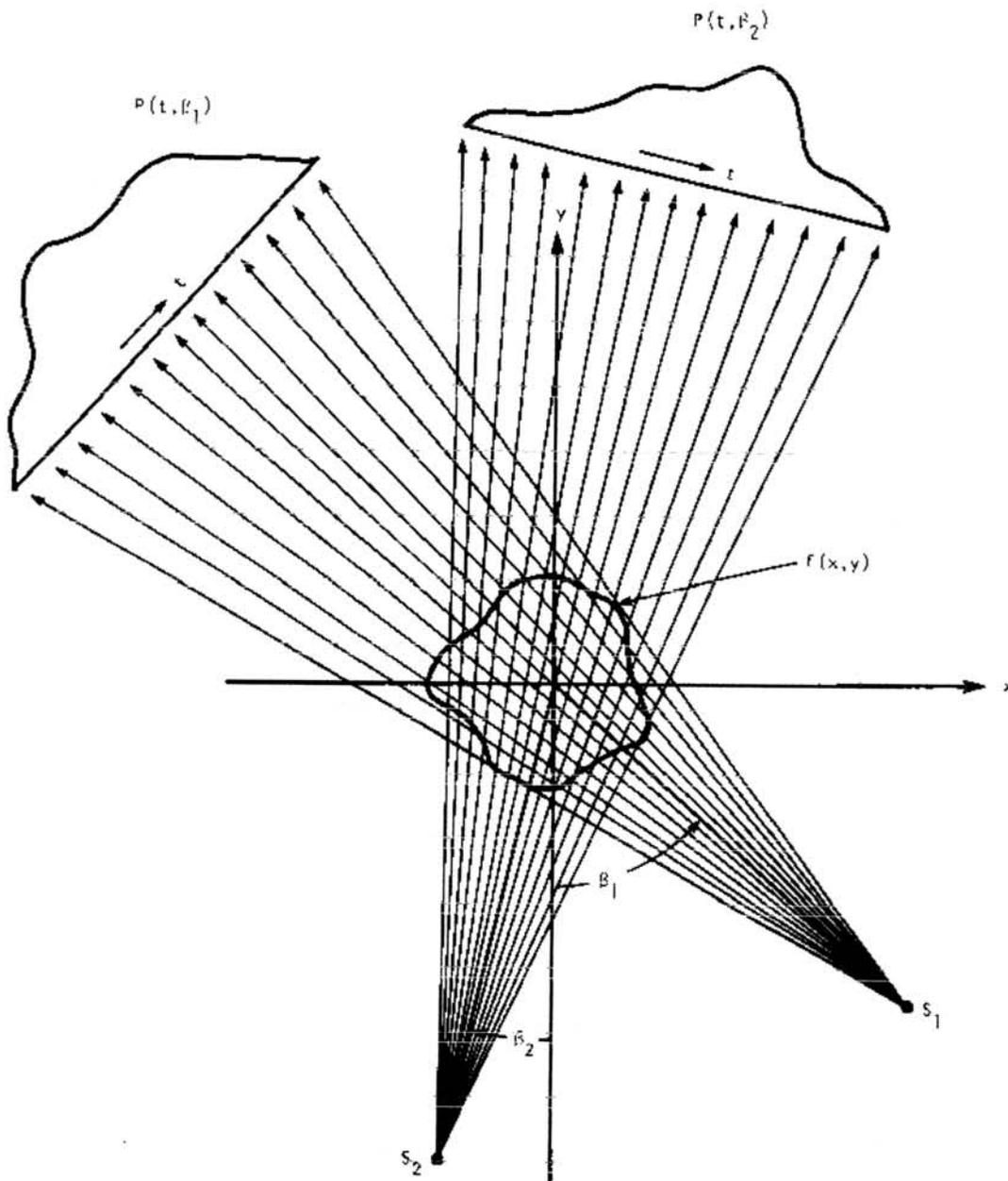


FIGURA 2.6 - Projeções *fan-beam* [ROS 82].

Neste trabalho, foram consideradas apenas projeções paralelas.

2.1.3 Reconstrução Bidimensional

Uma vez geradas as projeções, elas devem ser combinadas a fim de se obter a imagem original. Como dito anteriormente, cada projeção corresponde à soma da absorção de raio-x pelas estruturas, na direção em questão. É necessário então que estas projeções sejam combinadas de modo a gerar as formas individuais de cada objeto.

A intersecção das várias projeções dá a cada célula informações de vários ângulos, individualizando os valores totais das projeções para cada uma delas, permitindo que as formas dos objetos se formem de maneira incremental.

De um modo geral, a reconstrução bidimensional da imagem de uma fatia dá-se pela combinação gradativa de suas várias projeções unidimensionais.

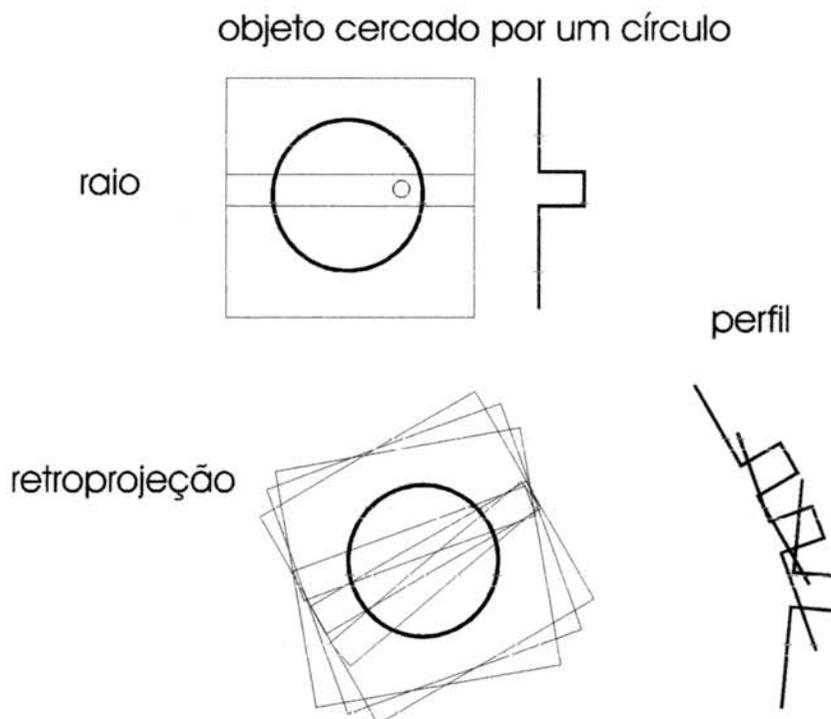


FIGURA 2.7 - Processo de reconstrução bidimensional através de projeções.

Na figura 2.7 é mostrado o processo de reconstrução da imagem de uma estrutura cercada por um círculo transparente ao raio-x, através da intersecção das várias projeções ali representadas. As projeções aparecem na forma de um pico, cuja ordenada é proporcional à intensidade de absorção dos objetos atravessados, para cada direção.

Serão apresentados, a seguir, três métodos de reconstrução bidimensional, todos baseados no processo descrito acima, a saber:

- 1) Técnica de Fourier,
- 2) Método da Convolução e

2) Método das Projeções.

A Técnica de Fourier, baseada na transformada de Fourier, utiliza técnicas matemáticas mais avançadas. Nesta etapa, o método será apresentado sem maiores aprofundamentos matemáticos ou de implementação, pois estes fogem ao escopo deste trabalho, mas podem ser encontrados em Rosenfeld *et al.* [ROS 82].

O Método da Convolução, ou Retroprojeção-Filtrada, é utilizado na maior parte dos tomógrafos, e utiliza técnicas matemáticas mais simples.

O Método das Projeções é baseado na resolução de sistemas de equações lineares, o qual expressa de uma forma mais clara o processo de reconstrução bidimensional.

2.1.3.1 Técnica de Fourier

Esta método é baseado na Transformada de Fourier, uma técnica matemática mais avançada, porém muitas vezes necessária, quando se trata de aplicações maiores e mais dispendiosas.

A Transformada de Fourier trabalha no domínio de frequência, e dá uma nova representação a uma função, normalmente mais fácil de ser utilizada. A função $f(x,y)$ que representa a imagem é transformada, bem como a função $p(t)$, que representa o raio soma. Feito isso, é possível desdobrá-las de modo a gerar funções, no domínio de frequência, para a reconstrução da imagem. Uma vez geradas, estas são anti-transformadas para gerar a imagem reconstruída [ROS 82].

Seja $F(u,v)$ a transformada de Fourier da imagem $f(x,y)$:

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy. \quad (2)$$

Seja também $S_{\theta}(w)$ a Transformada de Fourier da Projeção $P(t)$:

$$S_{\theta}(w) = \int_{-\infty}^{\infty} P_{\theta}(t) e^{-j2\pi wt} dt. \quad (3)$$

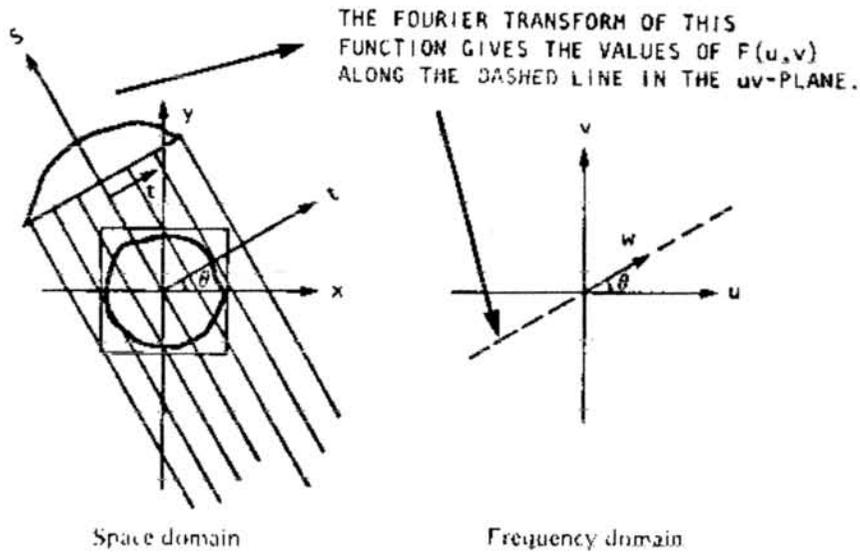


FIGURA 2.8 - Teorema da Fatia de Fourier [ROS 82].

$S_\theta(w)$ foi transformada de modo que os valores para a linha $v=0$ de $F(u,v)$, demonstrada na figura 2.8, possam ser obtidos:

$$S_\theta(w) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) e^{-j2\pi w(x \cos \theta + y \sin \theta)} dx dy \quad (4)$$

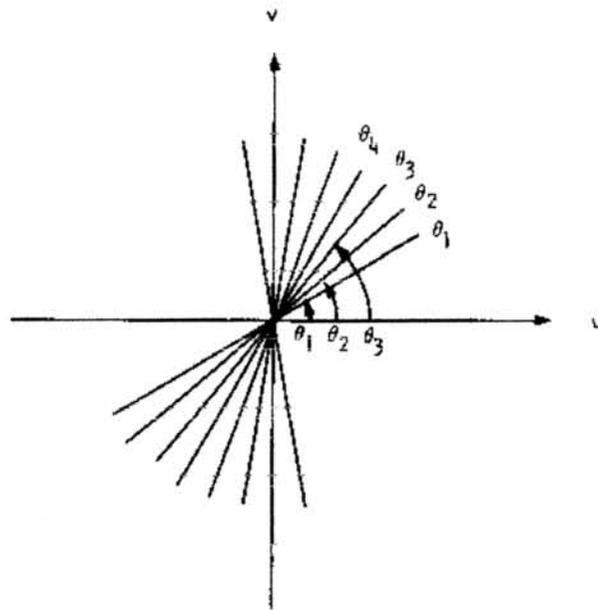


FIGURA 2.9 - Projeções levadas a diferentes ângulos [ROS 82].

Ao levar as projeções $P(t)$ da imagem a diversos ângulos $\theta_1, \theta_2, \dots, \theta_k$ e então aplicar a Transformada de Fourier S_θ a cada uma dessas projeções, pode-se determinar os valores de $F(u,v)$ nas linhas radiais, expressas na figura 2.9. Se infinitas projeções forem feitas, então $F(u,v)$ pode ser conhecido em todos os pontos do plano uv . Conhecendo $F(u,v)$, a função imagem $f(x,y)$ pode ser recuperada, utilizando a inversa da Transformada de Fourier:

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux+vy)} dudv \quad (5)$$

2.1.3.2 Método da Convolução

Este método é um dos mais utilizados na maioria das aplicações que utilizam raio-x (ou raio-gama), por ser um dos mais simples, e por ter-se mostrado bastante confiável em aplicações rápidas.

Este método, também chamado de Retroprojeção-Filtrada no espaço real, segue a forma básica de reconstrução bidimensional explicado anteriormente. O processo de distribuição adotado por este método, onde é encontrado um novo valor para cada célula, com base no seu peso e nos valores das células vizinhas, reconstrói a imagem filtrando cada projeção antes de retroprojetá-la [VIA 90].

Uma imagem é uma função bidimensional de intensidade luminosa $f(x,y)$, onde seu valor corresponde à intensidade nestas coordenadas. A imagem que se percebe consiste na luz dos objetos da cena. Cada ponto seu contém a contribuição da luminosidade de todos os outros pontos vizinhos. Um ponto (x,y) no plano de imagem não corresponde apenas ao ponto (x',y') no plano objeto, mas é também função da energia radiante dos infinitos vizinhos do mesmo [NET 89].

A imagem $g(x,y)$ corresponde à distribuição de energia de um objeto, e ao somatório das contribuições de seus vizinhos. A influência de um determinado ponto diminui nos pontos mais afastados, o que permite restringir a influência de um determinado elemento em seus vizinhos, sem perda significativa de informação para um posterior processamento. Esta restrição é feita normalmente atribuindo determinados pesos às células da imagem, na matriz convolução.

A matriz convolução define o peso da própria célula e de cada vizinho. Seus valores, bem como sua extensão, dependem do objetivo da convolução. Ela pode ser utilizada para reconstruir a imagem ou como um filtro, para ser utilizado na reconstrução da mesma.

Seu processo de distribuição, levado a diferentes direções, funciona como um algoritmo reconstrutor. Ele minimiza os efeitos produzidos por cada célula individualmente, levando em consideração as células vizinhas. Consiste em varrer toda a imagem e, para cada célula, encontrar um novo valor, considerando a própria célula e seus vizinhos, podendo atribuir diferentes pesos a cada uma delas. Somam-se os produtos das absorções das células da imagem por seus pesos e então divide-se pelo

divisor , que corresponde normalmente à soma dos termos da matriz convolução (consiste na distribuição do valor às células da imagem), e soma-se ao patamar, que pode ser o valor que a célula possui até o momento [ROX 90].

A divisão de cada célula da imagem pelo divisor ocorre a fim de minimizar o fator constante introduzido no cálculo, sendo chamado de Fator de Normalização [NET 89]. Este garante também que os valores permaneçam no intervalo de zero ao maior tom de cinza [ROX 90].

A convolução pode também ser utilizada como um filtro, na reconstrução da imagem, eliminando os efeitos indesejáveis da mesma. O processo é o mesmo descrito no método de reconstrução anterior, com a diferença que neste a imagem é filtrada..

Segundo esta abordagem, para cada projeção $P(t)$, é calculada a projeção filtrada $Q(t)$, que será utilizada para reconstruir a imagem. Assumindo-se que uma projeção, que só possui valores positivos, passe a ter uma série de valores positivos e negativos. Esta operação filtra a forma do objeto através de uma função filtro, incorporando no perfil do objeto valores positivos e negativos, o que contribui para a qualidade da imagem. Este processo é repetido para vários ângulos θ , então, os valores negativos dos perfis são subtraídos dos positivos, de modo a remover suas contribuições indesejadas, restaurando a verdadeira aparência do objeto.

As equações abaixo mostram que, para cada projeção $P\theta(t)$, primeiro calcula-se a projeção filtrada $Q\theta(t)$ por (7), para então, através de (6), reconstruir a imagem $f(x,y)$. A descrição detalhada deste teorema pode ser encontrada em Mattos [MAT 93].

$$f(x, y) = \int_0^{\pi} Q_{\theta}(x \cos \theta + y \sin \theta) d\theta \quad (6)$$

$$Q_{\theta}(t) = \int_{-\infty}^{\infty} S_{\theta}(w) |w| e^{j2\pi w t} dw \quad (7)$$

A decisão quanto a calcular a convolução discreta no domínio de frequência, onde se calcula a Transformada de Fourier, utilizando Transformada Rápida de Fourier FFT (*Fast Fourier Transform*), ou no domínio de tempo depende do tamanho da máscara de convolução (matriz convolução) a ser utilizada [NET 89].

2.1.3.3 Método das Projeções

Este método, proposto por Kaczmarz [MAT 93], utiliza a técnica de resolução de sistemas lineares. Esta técnica reconstrói a imagem de forma incremental, adicionando valores à ela até que esta seja reconstruída.

A abordagem matemática é utilizada a fim de reduzir o problema de dimensões, particionando o problema 2D em uma série de problemas lineares unidimensionais [PHE 77].

A técnica de resolução de sistemas lineares utilizada trata cada equação de (1) como uma equação do sistema, cujo ponto de intersecção das mesmas representa a solução do sistema. Tratando-se de uma imagem representada por N células, cada equação do sistema irá trabalhar com estas N variáveis. Deste modo, optou-se por trabalhar em um espaço N-dimensional, de modo que o ponto que representa a solução do sistema (intersecção das equações de (1)) corresponda à imagem resultante após todas as equações de (1) terem sido consideradas.

Considera-se cada raio-soma como um hiperplano em um espaço N-dimensional. Começa-se supondo uma solução inicial ao problema. Esta é projetada, como explicado a seguir, no hiperplano representado pelo primeiro raio-soma do feixe (primeira equação de (1)), gerando uma nova solução. Esta é projetada no hiperplano representado pelo segundo raio-soma do feixe, gerando uma segunda solução, e assim por diante até que todos os raios-soma tenham sido considerados. O processo reinicia projetando essa solução no primeiro raio-soma do novo conjunto, até que o ponto de intersecção destas equações seja encontrado.

Em termos práticos, este processo consiste em adicionar à imagem a diferença entre o raio-soma medido e o computado, para cada raio-soma. O raio-soma medido consiste no raio-soma obtido quando foram geradas as projeções, durante o processo de aquisição. O raio-soma computado consiste no raio soma obtido a partir da imagem em reconstrução, antes de cada raio-soma medido ser considerado.

Isto consiste em adicionar incrementalmente à imagem a variação que ocorreu com o novo raio-soma em relação ao anterior, para todas as projeções. Quando a variação for zero, a imagem estará reconstruída.

A equação seguinte (8) expressa o processo incremental de reconstrução, na iteração i, que ocorre com a imagem:

$$f_m^{(i)} = f_m^{(i-1)} + \left[(P_j - q_j) / \sum_{k=1}^N W_{jk}^2 \right] W_{jm}, \quad (8)$$

onde m é a célula cujo valor f_m está sendo calculado, P_j é o raio-soma medido j, W_{jk} é o peso atribuído à célula k pelo raio j e q_j é o raio-soma computado j e dado por:

$$q_j = \sum_{k=1}^N f_k^{(i-1)} W_{jk} \cdot \quad (9)$$

onde N é o número total de células, m é o número de células pertencentes ao objeto, W é o peso de cada célula, P_j é o raio-soma computado e q_j é o raio-soma medido.

No primeiro ciclo de iterações, abrangendo todas as M equações, o índice i da iteração é igual ao índice do raio-soma, ou seja, $i = j$.

Este método assemelha-se ao processo de convolução, não apresentando técnicas matemáticas muito avançadas e, muitas vezes, desnecessárias. Ele foi o adotado por este trabalho, uma vez que apresenta o processo de reconstrução sob uma nova abordagem, através do processo incremental, que melhor o representa.

2.2 Resumo

Este capítulo revisou os conceitos básicos da TC, desde a geração das projeções unidimensionais de uma imagem até sua reconstrução bidimensional. Vários métodos de reconstrução foram apresentados, bem como o processo básico de reconstrução seguido por eles.

Para que uma imagem bidimensional seja reconstruída, são necessárias várias de suas projeções unidimensionais. Para que as projeções sejam geradas, é necessária a composição de vários raios-soma, os quais representam a atenuação total de raio-x (se tratando de TCT) ao longo dos mesmos. Cada projeção representa, então, a atenuação total de raio-x, em determinada direção.

Uma vez geradas as projeções, elas devem ser levadas a um algoritmo de reconstrução para que a imagem seja gerada. Este algoritmo pode seguir diversos métodos para reconstruir a imagem, como: Técnica de Fourier, Método da Convolução ou Método das Projeções.

Independente do método utilizado, a reconstrução bidimensional da imagem se dá através da sobreposição de suas várias projeções, a fim de que as estruturas individuais dos objetos comecem a se formar.

O método adotado por este trabalho, Método das Projeções, consiste em adicionar à imagem, incrementalmente, a diferença entre o raio-soma medido na aquisição das projeções, e o raio soma computado, que consiste na soma dos produtos das absorções de todas as células por seu peso.

3 Proposta do Trabalho – 2D

Como já dito anteriormente, esta dissertação propõe o estudo das técnicas de reconstrução bidimensional, bem como a implementação de uma delas, com o objetivo de uma melhor compreensão deste processo. Além disso, é proposta, também, a expansão da técnica bidimensional implementada para a terceira dimensão.

Este capítulo trata do processo completo da reconstrução bidimensional. São apresentadas a modelagem dos objetos e a simulação do processo de tomografia computadorizada, onde é feita a aquisição das imagens projetadas, e a posterior reconstrução das mesmas.

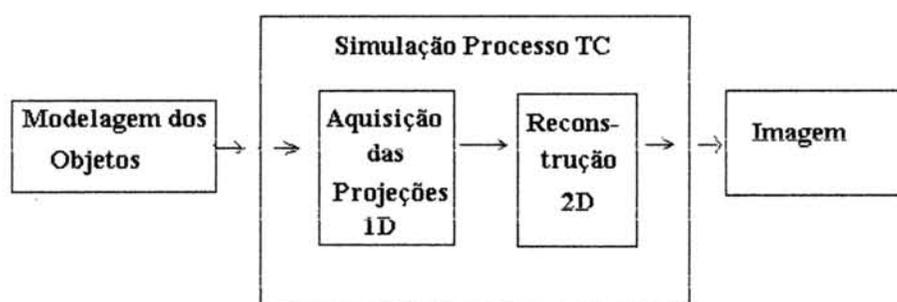


FIGURA 3.1 - Organograma da Proposta 2D.

O organograma acima expressa o processo completo da proposta de trabalho bidimensional. Primeiramente, os objetos são modelados e, a seguir, são submetidos ao processo de simulação da TC. Este processo ocorre em duas etapas: a primeira consiste na aquisição das imagens projetadas, bem como na gravação das mesmas em arquivos; a segunda consiste na reconstrução das mesmas.

Serão apresentados, neste capítulo, algoritmos e técnicas utilizados neste trabalho.

3.1 Modelagem dos Objetos

A síntese dos objetos é importante, pois possibilita uma melhor avaliação dos resultados obtidos, uma vez que já se conhece os objetos originais, tornando mais fácil sua comparação com os objetos reconstruídos, além de, muitas vezes, não se possuir, à disposição, objetos reais.

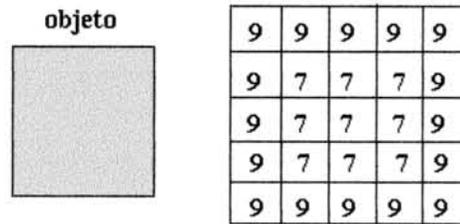


FIGURA 3.2 - Representação matricial do objeto.

Os objetos foram encarados sob a forma matricial, trabalhados de forma geométrica, como ilustrado na figura 3.2. Cada célula da matriz representa uma porção da face, portanto, o aumento no tamanho da imagem, aumenta, na mesma proporção, o número das porções das faces dos objetos sintetizados.

Esta matriz bidimensional é armazenada em um arquivo objeto, contendo vértices e porções de faces, como ilustrado abaixo:

```

arquivo OBT
VÉRTICES número de vértices
0 x y do vértice 0
1 x y do vértice 1
...
n x y do vértice n
PORÇÕES DAS FACES número de porções das faces
0 v1 v2 v3 v4 intensidade da porção da face 0
1 v1 v2 v3 v4 intensidade da porção da face 1
...
n v1 v2 v3 v4 intensidade da porção da face n

```

As porções das faces são originalmente *patches* quadrados.

3.2 Simulação do Processo de Tomografia Computadorizada

O objetivo desta etapa é a aquisição das imagens bidimensionais projetadas (AIP_2D) e suas posteriores reconstruções.

3.2.1 Método de Reconstrução Bidimensional Utilizado por Este Trabalho

Como dito anteriormente, o método de reconstrução bidimensional adotado por este trabalho foi o **Método das Projeções**, proposto por Kaczmarz [MAT 93]. A base deste método já foi comentada no capítulo anterior. Agora serão dados maiores detalhes sobre o mesmo. Primeiramente, é feito um estudo do algoritmo, passando, em seguida, ao seu refinamento.

3.2.1.1 O Método

A figura 3.3 mostra uma imagem $f(x,y)$, da região de interesse. Sobre esta imagem é colocada uma grade quadrada, com N (n^2) células, tendo-se $f(x,y) = \vec{f}(f_1, \dots, f_N)$, onde $f_1, \dots, f_m, \dots, f_N$ representam os valores das células da imagem.

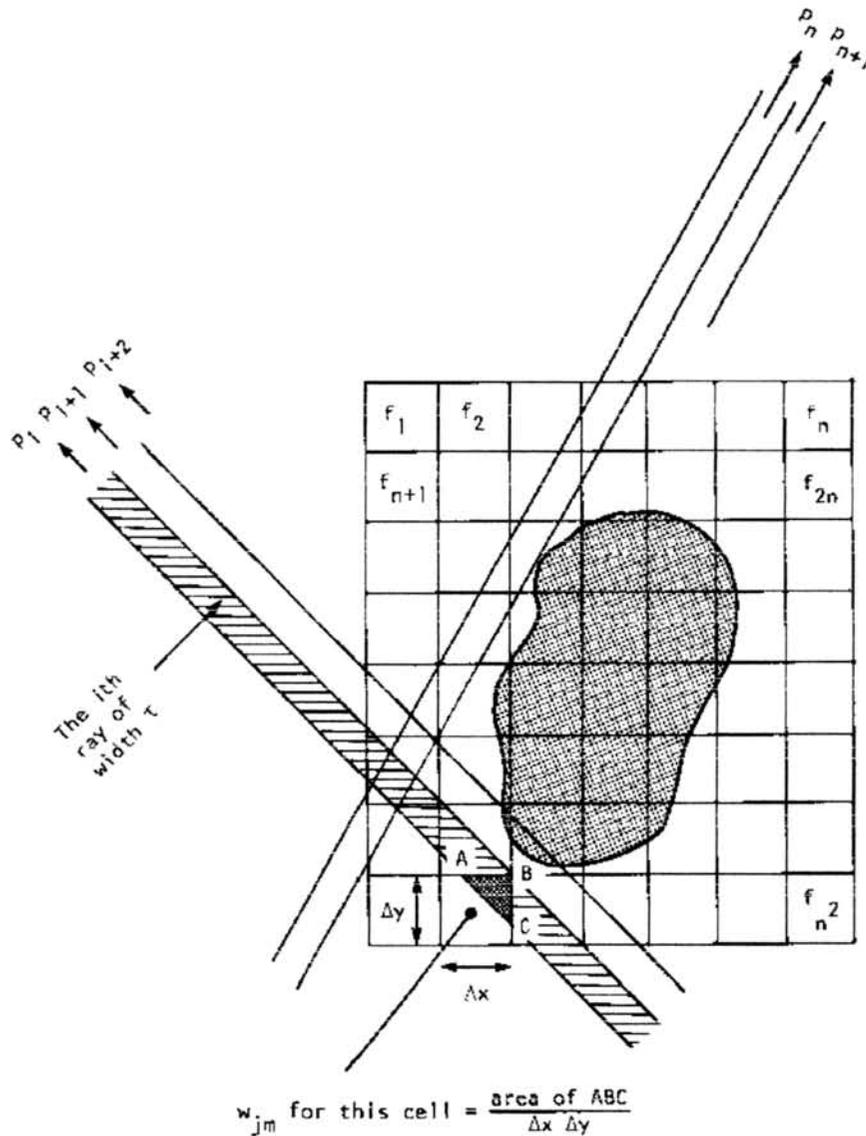


FIGURA 3.3 - Método de reconstrução bidimensional [ROS 82].

A imagem deve ser atravessada por raios para que seja gerada uma projeção. O raio é uma linha que corre através do plano xy , sua largura, representada por τ , corresponde normalmente à largura das células da imagem. A integral ao longo do raio, que representa a atenuação total do raio- x , ao longo desta linha do feixe, ou raio-soma P_j , pode ser obtida, como visto na seção 2.1.2, somando-se a absorção f_k em todas as células ao longo do raio j do feixe, como representado abaixo:

$$\sum_{k=1}^N W_{jk} f_k = P_j, \\ j = 1, 2, \dots, M \quad (10)$$

onde M é o número de raios de todas as projeções e W_{jk} é o peso da célula k para o raio-soma j , expressando o quanto esta célula representa para o raio em questão. O fator peso W_{jk} pode corresponder a uma parte fracionária da célula, como mostra a figura 3.3 para a célula m e o raio-soma j . Neste caso, o cálculo do peso pode ser obtido dividindo-se a área fracionária do raio pela área da célula, ou algo semelhante.

É necessária a composição dos M raios-soma do feixe para que uma projeção seja gerada, como ilustrado abaixo:

$$\begin{aligned} W_{11}f_1 + W_{12}f_2 + \dots + W_{1N}f_N &= P_1, \\ W_{21}f_1 + W_{22}f_2 + \dots + W_{2N}f_N &= P_2, \\ &\dots \\ W_{M1}f_1 + W_{M2}f_2 + \dots + W_{MN}f_N &= P_M \end{aligned} \quad (11)$$

Segundo Rosenfeld *et al.* [ROS 82], a representação de uma imagem por uma grade com N células, dá à mesma N graus de liberdade, ou seja, uma imagem representada por $\vec{f}(f_1, f_2, \dots, f_N)$, pode ser considerada como um único ponto em um espaço N -dimensional. Neste espaço, cada uma das equações acima representa um hiperplano. A intersecção de todos estes hiperplanos, se existir, é um ponto, o qual é a solução.

Primeiramente, é determinada uma suposição inicial, por exemplo $\vec{f}^{(0)}(0, 0, \dots, 0)$, que é projetada no primeiro hiperplano, representado pela primeira equação de (11), gerando uma solução $\vec{f}^{(1)}(f_1^{(1)}, \dots, f_k^{(1)}, \dots, f_N^{(1)})$. Esta passa a ser a nova suposição inicial, que será projetada no segundo hiperplano, representado pela segunda equação de (11), gerando uma nova solução $\vec{f}^{(2)}$. Este processo continua, até que a projeção seja levada ao último hiperplano, representado pela equação M de (11), terminando um ciclo completo. O processo inicia um novo ciclo projetando esta última solução no primeiro hiperplano do conjunto, representado pela primeira equação de (11), e assim por diante. Após todas as equações terem sido consideradas, o ponto de intersecção das mesmas, se existir, corresponde à solução do sistema.

A decisão de quando parar o processo, ou quantos ciclos executar, pode ocorrer através do processo de inspeção, onde o usuário interage com o sistema, ou de forma automática, associando o resultado a algum tipo de função que calcule o erro da imagem obtida, o qual deve estar contido dentro de um patamar pré-estabelecido.

O erro pode ser obtido, relacionando-se cada posição do vetor de imagem, do atual ciclo, com cada uma do ciclo anterior, de modo que se o maior erro encontrado estiver dentro do patamar estabelecido, a imagem é considerada reconstruída. Isto se deve ao fato de que a diferença entre o raio-soma medido e o

computado deve ser zero, quando a imagem estiver reconstruída, ou um valor muito próximo a este.

Para que o sistema convirja, isto é, para que o erro esperado seja alcançado, é necessário que se escolha a melhor combinação angular para as projeções. Este processo é explicado no capítulo 5, que trata da análise dos resultados expressos no anexo A-3.

O conceito acima é ilustrado pela figura 3.4, onde apenas duas células (f_1 e f_2) e dois raios-soma (P_1 e P_2) foram considerados, satisfazendo às seguintes equações:

$$\begin{aligned}W_{11}f_1 + W_{12}f_2 &= P_1, \\W_{21}f_1 + W_{22}f_2 &= P_2\end{aligned}\tag{12}$$

Foram consideradas apenas duas células da imagem, com suas absorções f_1 e f_2 , e seus pesos W_1 para o raio-soma P_1 e W_2 para o raio-soma P_2 .

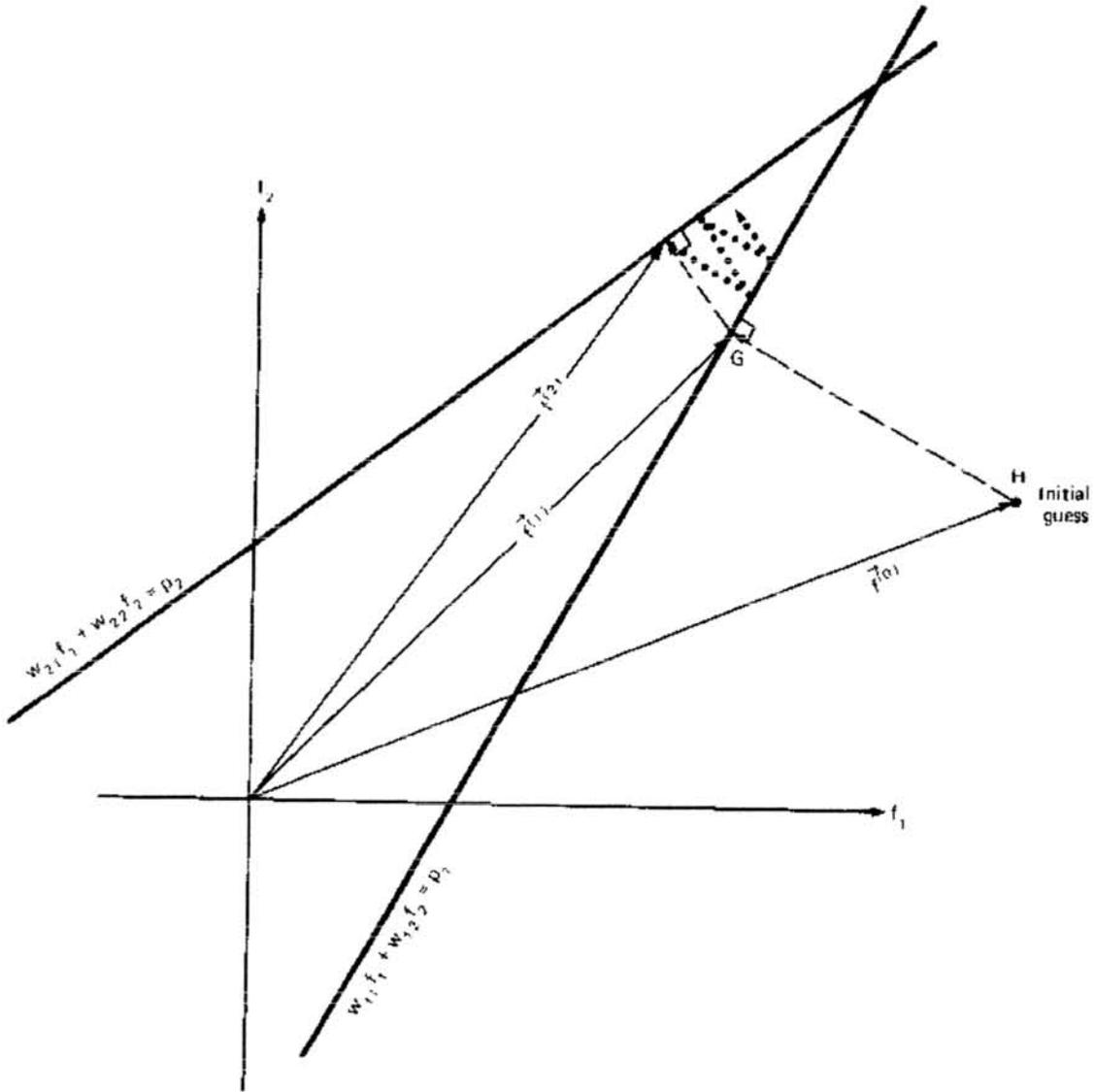


FIGURA 3.4 - Método de Kaczmarz [ROS 82].

Na figura 3.4, o ponto $H(f_1^{(0)}, f_2^{(0)})$ representa a suposição inicial da solução, vctorialmente representada por $\vec{f}^{(0)}$, este é projetado, como explicado posteriormente, no primeiro hiperplano de (12), gerando o ponto $G(f_1^{(1)}, f_2^{(1)})$, o qual consiste na nova solução, representada vctorialmente por $\vec{f}^{(1)}$. Agora, G é projetado no segundo hiperplano de (12), gerando uma nova solução, representada vctorialmente por $\vec{f}^{(2)}$. Esta deve ser projetada novamente no primeiro hiperplano de (12), gerando uma nova solução, e assim por diante, até que o sistema convirja ao ponto de intersecção de suas equações, o qual representa a solução do mesmo, como representado pela seta pontilhada do gráfico.

Faz-se primeiro uma suposição inicial, esta é projetada no hiperplano representado pela primeira equação de (12), gerando uma nova suposição à solução. Esta nova suposição é então projetada no próximo hiperplano representado pela

segunda equação de (12), e assim por diante, até que a última equação M seja considerada. Este processo é repetido até que o ponto de intersecção das mesmas seja encontrado.

Este processo de projeção representa subtrair da imagem a contribuição indesejada que cada raio-soma dá à mesma, de maneira incremental, fazendo a diferença entre o raio-soma computado, que corresponde à soma dos produtos das absorções de cada célula por seu peso, e o raio-soma medido, o qual representa a soma do valor real que as células deveriam possuir. A implementação deste passo pode ocorrer pela seguinte equação vetorial, ilustrada para o raio-soma $j=1$ e cujo desenvolvimento completo se encontra no anexo A-2:

$$\vec{f}^{(1)} = \vec{f}^{(0)} - \vec{W}_1(\vec{W}_1 \cdot \vec{f}^{(0)} - P_1) / (\vec{W}_1 \cdot \vec{W}_1) \quad (13)$$

onde $\vec{W}_{j=1} = \vec{W}_1 = (W_{11}, W_{12}, \dots, W_{1N})$, o que corresponde ao peso de todas as células N para o raio-soma 1. Estes valores correspondem aos coeficientes da primeira equação de (11), a qual pode ser reescrita como segue abaixo:

$$\vec{W}_1 \cdot \vec{f} = P_1 \quad (14)$$

onde $\vec{f} = (f_1, f_2, \dots, f_N)$, o que corresponde às absorções da imagem nas N células da grade.

A equação (13) representa subtrair da imagem $\vec{f}^{(0)}$ a diferença que ocorreu entre o atual raio-soma computado, que corresponde à soma dos produtos das absorções de todas as células por seus pesos, e o raio soma P_1 , medido na aquisição das projeções. Esta diferença deve ser normalizada, a fim de atribuir um valor individual a cada célula e não um valor acumulado. Estes valores devem ainda ser multiplicados pelos pesos das células para serem subtraídos das células da imagem de forma proporcional. Ou seja, se apenas a metade desta célula foi atravessada pelo raio, apenas cinquenta por cento do valor a ser subtraído deve ser considerado, pois o restante pertence a outros raios. Após todas as células da imagem $\vec{f}^{(0)}$ terem sido consideradas, é gerada uma nova imagem P_1 , que corresponde a uma nova suposição inicial, isto é, a imagem de partida para o novo raio do feixe.

Após $\vec{f}^{(1)}$ estar disponível, a projeção é levada ao hiperplano representado pela segunda equação de (11), gerando $\vec{f}^{(2)}$, como também ilustrado na figura 3.4. Este processo é repetido com o terceiro hiperplano, e assim por diante. A projeção $\vec{f}^{(j)}$ no hiperplano j pode ser obtida, a partir da $\vec{f}^{(j-1)}$ por meio da equação vetorial:

$$\vec{f}^{(j)} = \vec{f}^{(j-1)} - \vec{W}_j(\vec{f}^{(j-1)} \cdot \vec{W}_j - P_j) / (\vec{W}_j \cdot \vec{W}_j) \quad (15)$$

onde $\vec{W}_j = (W_{j1}, W_{j2}, \dots, W_{jN})$, peso das N células para o raio-soma j.

3.2.1.2 Refinamento

No método descrito acima, para computar cada raio-soma em (14), ou mesmo para considerá-los em (15), trabalha-se sempre com o número total de células N . Porém, a maior parte dos pesos associados a elas são zeros, não sendo necessário o cálculo com os mesmos. Supondo que o raio-soma j atravessasse apenas dez por cento das N células, apenas $N/10$ células necessitam ser consideradas, pois as demais possuem seus pesos zerados. Se f_m representa o valor constante presente nas m células pertencentes ao objeto, isto é, suas absorções, quando a imagem for atravessada pelo raio-soma j do feixe, apenas as células da mesma atravessadas por ele devem ser consideradas, sendo suas absorções expressas por f_j .

Caso isso não ocorra, a implementação de (15) terá limitações no que diz respeito ao cálculo, armazenamento e recuperação rápida dos pesos. Considerando uma imagem em uma grade de 128×128 células a ser reconstruída a partir de 150 raios, para cada projeção (150 projeções), o número total de pesos W_{ij} necessários neste caso é $128 \times 128 \times 150 \times 150$ (aproximadamente $2,7 \times 10^8$), o que representa um número consideravelmente grande.

Existem diversas abordagens para a implementação do processo descrito acima. Uma delas, e a mais simples, é considerar as células cujo centro se encontram dentro do raio. As que estiverem recebem peso 1, caso contrário, 0. Este procedimento se dá em tempo de execução. Uma outra abordagem, mais segura, consiste em detectar todas as células interceptadas pelo raio, e então utilizá-las como entrada para a função de reconstrução. Este procedimento se dá também em tempo de execução, a cada novo raio a considerar. O valor do peso é computado de acordo com a área da célula que se encontra dentro do raio, computado como explicado no início de 3.2.1.1. Este processo foi adotado neste trabalho.

Diante das dificuldade de implementação de (15), sugere-se uma aproximação à mesma, a qual consiste em calcular, a cada nova solução, a alteração que sofreu em relação à anterior, incorporando-a de forma incremental à imagem que está sendo gerada, para todos os raios do feixe, em todas as direções. Estes raios atravessam as células da imagem desprezando, como dito anteriormente, as células sem nenhuma contribuição.

Nestes termos, a alteração de (15) se dá, adicionando-se à solução anterior, a diferença entre o raio-soma medido, ao longo do raio j , e o computado para o mesmo raio:

$$f_m^{(j)} = f_m^{(j-1)} + \left[(P_j - q_j) / \sum_{k=1}^N W_{jk}^2 \right] W_{jm}, \quad (16)$$

onde

$$q_j = \vec{f}^{(j-1)} \cdot \vec{W}_j = \sum_{k=1}^N f_k^{(j-1)} W_{jk} \quad (17)$$

Isto significa que ao projetar a solução (j-1) no hiperplano j, o incremento $\Delta f_m^{(j)}$ no valor da célula m é dado por:

$$\Delta f_m^{(j)} = f_m^{(j)} - f_m^{(j-1)} = \left[(P_j - q_j) / \sum_{k=1}^N W_{jk}^2 \right] W_{jm} \quad (18)$$

onde P_j é o raio-soma medido ao longo do raio j e q_j é o raio-soma computado para o mesmo raio, baseado na solução (j-1). A correção $\Delta f_m^{(j)}$ para a célula m é obtida calculando a diferença entre o raio-soma medido e o computado. Esta diferença é então normalizada por $\sum_{k=1}^N W_{jk}^2$ e multiplicada pelo peso correspondente, W_{jm} .

O denominador da equação (18) pode ser alterado a fim de considerar apenas as células interceptadas pelo raio j, representado por N_j . A correção para a célula m da imagem através da equação j em (11) pode ser escrita por:

$$\Delta f_m^{(j)} = (P_j - q_j) / N_j \quad (19)$$

para todas as células interceptadas pelo raio. Está-se adicionando à imagem a diferença $(P_j - q_j) / N_j$, o que é análogo à retroprojeção dos algoritmos de projeção filtrada, como expresso no 2.1.3.2. Em (19), os q_j 's são calculados segundo (17), desprezando, como já comentado, as células que não interceptam o raio.

Segundo Rosenfeld [ROS 82], a equação (19) pode ser considerada uma aproximação à simplificação expressa por (20), subtraindo-se (17) da equação j em (11):

$$P_j - q_j = \sum_{k=1}^N \Delta f_m^{(j)} W_{jk} \quad (20)$$

onde $\Delta f_m^{(j)}$ é agora a diferença entre o valor médio correto da célula m e seu valor após usar a equação (j-1) em (11).

O método de reconstrução implementado neste trabalho seguiu este refinamento, e utilizou o processo de intersecção das células com os raios do feixe para minimizar o processamento.

3.2.2 Algoritmo de Intersecção

Este algoritmo é uma adaptação do *Scan-Converting Polygons*, encontrado em Newman [NEW 79].

Ele consiste em encontrar todos os *pixels* que estão dentro das fronteiras de um polígono e gerar sua visualização, através das intensidades dos mesmos. Através da descrição das bordas do polígono, é possível determinar se um certo ponto está dentro ou fora dele por contar as intersecções de uma linha imaginária, estendida do ponto até um outro ponto distante do polígono, com as bordas do mesmo. Se o número de intersecções encontrado for ímpar, o ponto está dentro do polígono, caso contrário encontra-se fora dele.

Este algoritmo foi adaptado para gerar a intersecção dos raios dos feixes com as células da grade, uma vez interceptadas, elas são armazenadas em uma lista que servirá como entrada para o algoritmo de reconstrução. A descrição detalhada do algoritmo original pode ser encontrada em Newman *et al.* [NEW 79].

A intersecção, no algoritmo original, se dá de forma incremental, ou seja, encontra-se um passo para o x e outro para o y , gerando um novo ponto, com o qual é testado se há intersecção ou não com as fronteiras do polígono, e assim por diante, até que se encontre o ponto final da linha imaginária. Este processo incremental está expresso na figura 3.5.

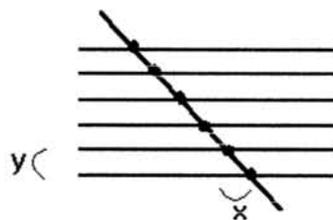


FIGURA 3.5 - Processo incremental de intersecção.

Na adaptação feita, a linha imaginária passa a ser cada raio do feixe, o qual possui uma largura. O primeiro passo é testar se pelo menos um dos vértices do polígono se encontra dentro deste intervalo. Se a resposta for afirmativa, insere-se o polígono em questão na lista encadeada de porções das faces. Se isto não ocorrer, passa-se para a segunda etapa, onde deve-se testar se pelo menos uma das arestas do polígono intercepta o raio. Para isto, é testado primeiramente se os quatro vértices do polígono se encontram simultaneamente do lado esquerdo, fora da faixa, ou simultaneamente do lado direito, também fora da faixa. Dessa forma nenhuma das arestas do polígono poderia interceptar o raio. Caso isto também não ocorra, pelo menos uma das arestas do polígono deve intersecionar o raio. Neste caso, é considerada cada aresta do polígono independentemente. Se ela intersecionar o raio, as demais são desconsideradas, o polígono é armazenado na lista de porções das faces e passa-se para o novo polígono.

Da mesma maneira como é feita a intersecção da linha imaginária com o polígono, no algoritmo original, faz-se, nesta adaptação do algoritmo, a intersecção da aresta com o raio, de forma incremental. Este processo consiste em determinar o x-inicial e o x-final da aresta. Uma vez determinados, é feita a diferença de ambos e o resultado é dividido pelo comprimento em y da aresta (o y-final menos o y-inicial da aresta), como demonstrado abaixo:

$$\begin{aligned} \text{tam} &= y_f - y_i \\ dx &= (x_f - x_i) / \text{tam} \\ \text{enquanto } x_i &< x_f \\ x_i &= x_i + dx \end{aligned}$$

onde, a cada valor intermediário de x_i , é testado se está dentro ou fora da faixa. Se estiver dentro, o polígono é inserido na lista de porções das faces, desprezando as demais arestas, caso contrário, a intersecção é processada com as mesmas.

Nesta adaptação do algoritmo, foi desprezado o incremento em y, pois o raio assume todos os valores de y das porções das faces dentro da faixa estabelecida em x.

Após todas as porções das faces que interseccionam o raio estarem armazenadas na lista de porções das faces, a mesma é organizada em y, de forma descendente, uma vez que é a forma como o algoritmo de projeção e o de reconstrução irão solicitá-las. O método utilizado para esta ordenação foi o de Pesquisa Binária.

O algoritmo segue abaixo em linguagem estruturada:

```

procedimento intercepta
  para cada polígono
    se os quatro vértices estiverem dentro do intervalo de x
      insere polígono na lista de porções das faces
    senão
      se os quatro vértices estiverem simultaneamente do
      lado direito ou do lado esquerdo, fora da faixa
        passa para o próximo polígono
    senão
      para cada aresta
        tam=(yf-yi)
        dx=(xf-xi)/tam
        enquanto xi<xf
          xi=xi+dx
          se xi dentro do intervalo de x
            insere polígono na lista de porções
            das faces
            passa para o próximo polígono
          fim do se
        fim do enquanto
      fim do para
    fim do senão
  fim do senão
fim do para
fim do procedimento

```

Pode-se, para acelerar o processo, ordenar os polígonos em x antes de processá-los neste algoritmo, evitando que todos sejam testados. O algoritmo de Pesquisa Binária pode também ser utilizado para esta ordenação.

3.2.3 Algoritmo de recorte

Uma vez selecionadas as porções das faces que interceptam o raio em questão, deve-se calcular o quanto de cada polígono se encontra dentro do raio, isto é, a área do mesmo. Uma vez encontrada, é possível calcular o peso do polígono para este raio, descobrindo qual o percentual que esta área representa.

Para encontrar a área do polígono que é interior ao raio, foi utilizada, neste trabalho, uma adaptação do algoritmo de recorte bidimensional. O algoritmo original pode ser encontrado em Newman *et al.* [NEW 79].

Assim como no algoritmo original, o recorte é feito sobre uma área quadrada que, ao invés de representar uma janela de seleção, representa uma parte do raio, ou seja, é tomado um intervalo em y, para o intervalo em x comentado no

algoritmo anterior. O recorte é feito individualmente a cada aresta. A cada uma delas, cada vértice do polígono é testado. Se estiver dentro do limite da mesma, é inserido em uma lista encadeada de vértices. Após terem sido considerados todos os vértices do polígono, esta lista serve de entrada para o teste com a próxima aresta, descartando os que não foram selecionados na aresta anterior. Isto representa um processo de exclusão, e os vértices que restarem na lista formam um novo polígono que representa a área do polígono testado que se encontra dentro da parte do raio em questão.

Porém, um determinado vértice de uma das arestas do polígono pode estar dentro do limite da aresta considerada, mas o vértice da outra extremidade pode estar fora. Isto representa que houve uma intersecção e este ponto deve ser encontrado, como expresso na figura 3.6.

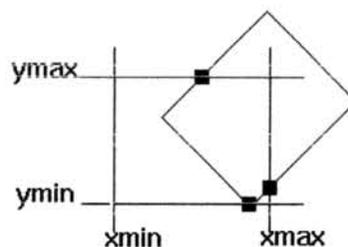


FIGURA 3.6 - Intersecção da aresta do polígono com as arestas limites.

Este ponto é calculado como ilustrado abaixo:

procedimento recorta extremo

caso a aresta em questão seja a da esquerda

$$y_n = y_1 + (y_2 - y_1) * (x_{min} - x_1) / (x_2 - x_1)$$

$$x_n = x_{min}$$

caso a aresta em questão seja a da direita

$$y_n = y_1 + (y_2 - y_1) * (x_{max} - x_1) / (x_2 - x_1)$$

$$x_n = x_{max}$$

caso a aresta em questão seja a de baixo

$$x_n = x_1 + (x_2 - x_1) * (y_{min} - y_1) / (y_2 - y_1)$$

$$y_n = y_{min}$$

caso a aresta em questão seja a de cima

$$x_n = x_1 + (x_2 - x_1) * (y_{max} - y_1) / (y_2 - y_1)$$

$$y_n = y_{max}$$

fim do procedimento

No caso ilustrado na figura 3.7, onde o segundo vértice de uma das arestas do polígono e o primeiro vértice de sua próxima aresta se encontram fora do quadrado que representa o raio, envolvendo um dos vértices do mesmo, deve-se garantir que, no polígono resultante, não apenas os pontos de intersecção 5 e 7 estejam presentes, mas também o vértice 6, pois este faz parte da nova figura que se formou (vértices 1,5,6,7,4). Isto é garantido, uma vez que as arestas são consideradas

separadamente. Quando a aresta de cima estiver sendo considerada, o ponto de intersecção 5 será determinado e incluído na lista de vértices. O vértice 3, em relação a esta aresta, é considerado dentro e, como o 2 está fora, uma intersecção deve ocorrer, que é o próprio vértice 6. Quando a aresta da direita for considerada, o vértice 3, que era considerado dentro para a aresta anterior, agora está fora, sendo determinado então o ponto de intersecção 7, uma vez que o vértice 4 se encontra dentro dos limites para esta aresta. O processo continua para as demais arestas do polígono.

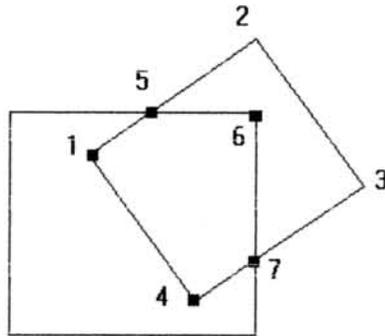


FIGURA 3.7 - Recorte do polígono.

O algoritmo descrito acima é ilustrado abaixo em linguagem estruturada:

```

procedimento recorte
  para cada polígono
    preenche lista1 com os vértices do polígono
    para cada aresta limite
      para cada vértice do polígono
        se o vértice estiver dentro do limite da aresta
          insere o vértice na lista2 de vértices
          se o vértice anterior está fora
            recorta extremo
            insere ponto de intersecção na lista2, em uma
            posição anterior
          fim do se
        fim do para
      se o vértice anterior está dentro do limite
        recorta extremo
        insere ponto de intersecção na lista2, na posição
        atual
      fim do se
      próximo vértice da lista1
    fim do para
    limpa lista1
    coloca conteúdo de lista2 em lista1
    limpa lista2
  fim do para
  limpa lista1
fim do para
fim do procedimento

```

São necessárias duas listas de vértices para que seja possível que os vértices considerados dentro, para a aresta anterior, sejam entrada para a aresta em questão, ocorrendo assim um processo de exclusão.

Os vértices considerados dentro dos limites da aresta são inseridos na lista de forma que estejam organizados em sentido horário.

3.2.4 Aquisição das Imagens Projetadas

Ao contrário do convencional, onde as projeções são geradas rotando-se a fonte de raio-x ao redor do objeto, neste trabalho, foi adotado o procedimento inverso, onde o objeto é rotado a cada novo ciclo de projeções. Sobre este, é colocada uma grade quadrada, que representa um conjunto de raios paralelos, sempre fixos, como ilustrado na figura 3.8.

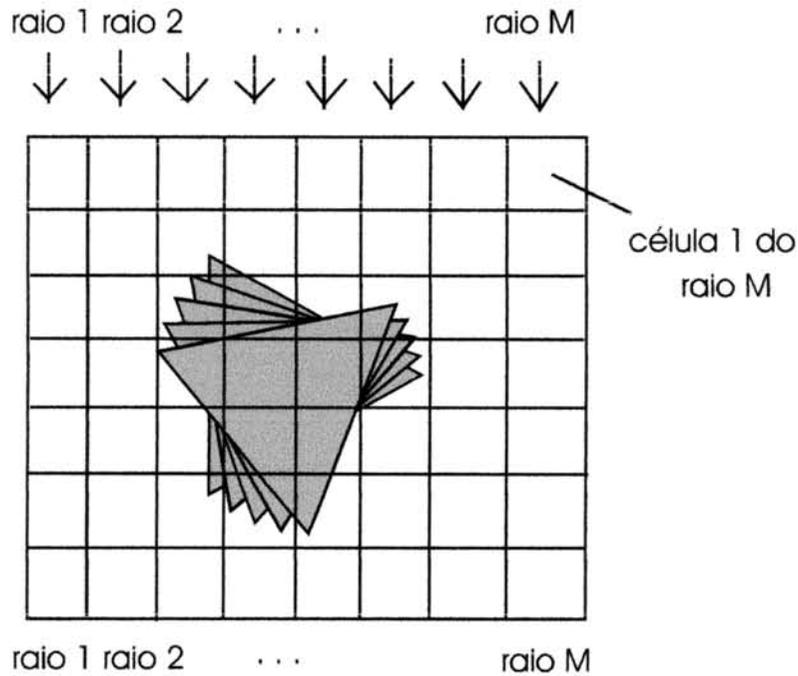


FIGURA 3.8 - Grade de raios.

Esta grade tem as dimensões do envelope do objeto e o número de raios que se desejar. A largura do raio é determinada pelo número deles e pelas dimensões do envelope. O fator x e o fator y são computados por:

$$\begin{aligned} \text{fatx} &= (\text{xmax}-\text{xmin}) / \text{raios}, \\ \text{faty} &= (\text{ymax}-\text{ymin}) / \text{raios} \end{aligned}$$

onde raios corresponde ao número de raios-soma do feixe, xmax , xmin , ymax , ymin correspondem aos extremos do envelope.

A grade é percorrida de cima para baixo, da esquerda para a direita. Ela começa a se formar a partir do xmin . Do xmin ao $\text{xmin} + \text{fatx}$ corresponde ao primeiro raio, que será percorrido do ymax ao $\text{ymax}-\text{faty}$, sucessivamente até que o ymin seja encontrado. A seguir, incrementa-se o atual x com fatx para que o segundo raio do feixe seja considerado, repetindo o processo acima. Este processo para todos os raios, até que o xmax seja encontrado.

Em cada célula desta grade, ou seja, em cada incremento em y no raio é necessário descobrir se existe algum objeto para que se some suas absorções ao longo do raio em questão, como ilustrado na figura 3.9.

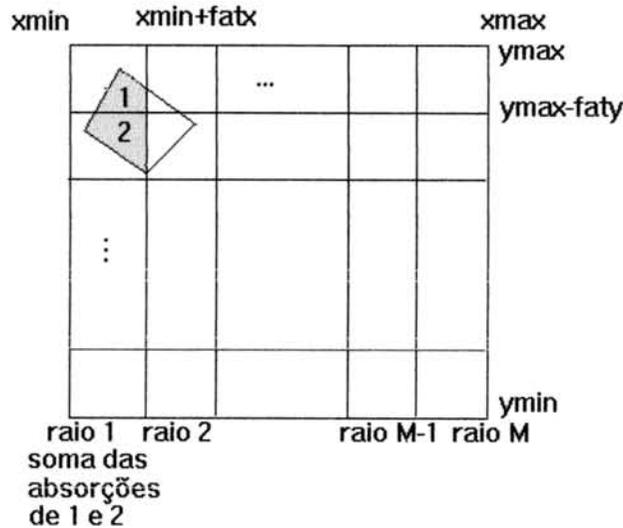


FIGURA 3.9 - Raio-soma.

Portanto, é necessário detectar que porções das faces interceptam este raio. Para isto, é executado o algoritmo descrito em 3.2.2, para cada raio a considerar. Este algoritmo irá criar uma lista encadeada contendo as porções das faces que se encontram na faixa x que corresponde ao raio em questão. Como o raio é percorrido de cima para baixo, a lista está organizada em y , no sentido descendente. Uma vez gerada a lista, o algoritmo começa a descer a grade, no respectivo raio, decrementando, o atual valor de y , de $fatx$. A primeira célula da grade, que representa a primeira área quadrada, do primeiro raio, corresponde aos extremos: x_{min} , y_{max} ; x_{max} , y_{min} . Este valor é então passado como parâmetro para o algoritmo *recorte*, descrito em 3.2.3, representando a área limite para o mesmo. Este considera esta área limite em suas arestas individuais que, neste caso, são:

x_{min}, y_{max} a x_{max}, y_{max} ,
 x_{max}, y_{max} a x_{max}, y_{min} ,
 x_{max}, y_{min} a x_{min}, y_{min} e
 x_{min}, y_{min} a x_{min}, y_{max}

representadas no sentido horário. Para cada célula do raio, são consideradas as várias porções das faces existentes na lista gerada pelo algoritmo de intersecção. A cada porção da face, o algoritmo *recorte* testa se existe alguma parte da mesma dentro desta área limitada do raio, se houver, seus vértices são inseridos em uma lista de vértices. É necessário, neste momento, calcular a área deste polígono, para que se possa determinar o peso desta fração de porção da face para a célula do raio-soma que se está considerando, como ilustrado na figura 3.10.

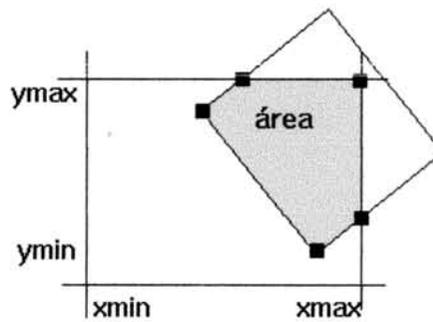


FIGURA 3.10 - Peso da porção da face para a célula.

Uma vez calculada a área, divide-se pela área da célula, a qual é facilmente obtida. Esta razão corresponde ao peso que a porção da face atual tem para a respectiva célula do raio em questão. Consiste no percentual da área da porção da face pertinente a esta célula. Este peso é então multiplicado pela intensidade da porção da face considerada, obtendo-se assim o valor da absorção da área da porção da face atravessada. Se, para esta célula do raio, mais de uma porção da face for interseccionada, o processo se repete para as demais e o valor de suas absorções é acumulado, como ilustrado na figura 3.11.

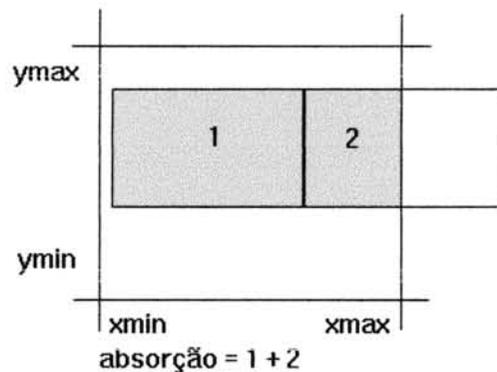


FIGURA 3.11 - Absorção da célula.

Ao sair deste algoritmo, volta-se à grade para que o raio avance mais uma célula (valor atual de y - faty). Esta célula é levada ao algoritmo *recorte*, e retorna com o valor da absorção de todas as frações de porções das faces interceptadas por ela, o qual é acumulado ao da célula anterior. Ao final deste raio, quando o y se igualar ao y mínimo, estarão acumulados os valores das absorções para todas as células do raio, o que representa o valor do primeiro raio-soma.

Este processo é repetido para os demais raios da grade, somando-se fatx ao valor atual de x , até que o x se iguale ao x máximo. Neste momento estarão disponíveis todos os raios-soma, o que corresponde ao término do primeiro ciclo de projeções.

A área do polígono interno à cada célula do raio é calculada em duas etapas. A primeira etapa é descobrir o x mínimo e o x máximo do polígono e, de posse destes dados, partir do vértice que contém o menor x , pegando-os dois a dois. Traça-se, de cada um deles, uma linha paralela ao eixo y até a origem, gerando um trapézio, cuja área é simples de se calcular. Avança-se um vértice a fim de gerar um novo trapézio cuja área é somada à do anterior, até chegar no x máximo. Neste momento, muda-se o sentido, partindo-se do vértice de maior x até chegar no de menor x , diminuindo-se, porém, da soma anterior, as áreas dos trapézios encontrados. O que resulta deste processo é a área do polígono cujos vértices estavam sendo considerados, como ilustrado na figura 3.12.

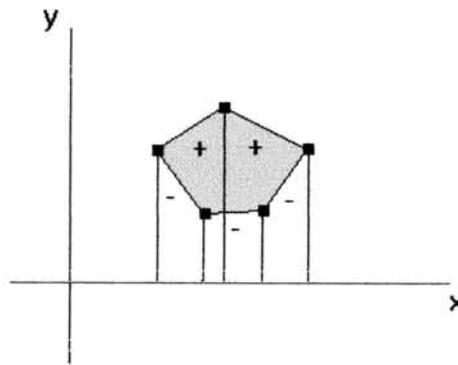


FIGURA 3.12 - Cálculo da área do polígono interno à célula.

Para facilitar o cálculo da área acima, optou-se por manter os vértices da lista ordenados no sentido horário, além de transformá-la, no momento do cálculo da área, em uma lista circular.

Após o fim do primeiro ciclo de projeções, é necessário que o objeto seja rotado a fim de que se inicie o segundo ciclo. Após o objeto ter sido rotado, começa-se a percorrer a grade de raios e o processo se repete, para todas as projeções.

As projeções são armazenadas em uma lista onde cada nodo endereça uma nova lista que corresponde aos raios-soma da respectiva projeção. Esta lista é passada como parâmetro para o algoritmo de reconstrução, juntamente com as dimensões da grade de raios.

Este processo é expresso no procedimento abaixo:

```

procedimento AIP_2D
  acha envelope objeto
  calcula fatx
  calcula faty
enquanto número de rotações menor que o limite
  enquanto xmin<xmax
    intercepta(xmin, xmin+fatx)
    enquanto ymax>ymin
      recorte(xmin, xmin+fatx, ymax, ymax-fatx)
      acumula absorção
      ymax = ymax - faty
    fim do enquanto
    raio_soma_medido[xmin] = absorção acumulada
    xmin = xmin + fatx
  fim do enquanto
  rota objeto(ângulo)
fim do enquanto
fim do procedimento

```

O procedimento *recorte*, a seguir, é baseado no algoritmo expresso em 3.2.3, porém, além de encontrar a lista de vértices do polígono resultante, calcula o peso do mesmo para a célula considerada e, logo após, a sua absorção. A função abaixo representa, para todas as células do raio, o raio-soma medido.

```

procedimento recorte
para todas as porções das faces da lista
  acha peso
  absorção= peso * intensidade da porção da face
  acumula absorção
fim do para
fim do procedimento

```

Após o término de cada ciclo de projeções, elas são armazenadas em arquivos.

3.2.5 Algoritmo de Reconstrução

Nesta etapa, faz-se a reconstrução da imagem com base nas projeções dela geradas. A reconstrução deve ser feita a partir dos mesmos ângulos pelos quais as projeções foram geradas, e com o mesmo número de raios. É evidente também que a largura do raio, bem como as dimensões da grade devem ser as mesmas utilizadas na geração das projeções.

Assim como na AIP_2D, o objeto é rotado, enquanto que a fonte de raios permanece fixa. A maneira pela qual os raios percorrem o objeto é a mesma utilizada na AIP_2D. Uma grade quadrada é colocada sobre o objeto, onde cada coluna corresponde a um raio do feixe. Ela é percorrida de cima para baixo e da esquerda para a direita.

A imagem (matriz geométrica do objeto) inicial considerada neste algoritmo, isto é, suposição inicial, possui todas as suas intensidades zeradas.

Assim como em AIP_2D, a cada faixa de x , isto é, a cada raio, é executado o algoritmo de intersecção para que só sejam processadas as porções das faces que interseccionam o raio, armazenadas em uma lista de porções das faces, como ilustrado na figura 3.13. Esta será utilizada a cada célula do raio, a fim de determinar a área interna de cada porção da face à célula, através do algoritmo de *recorte*.

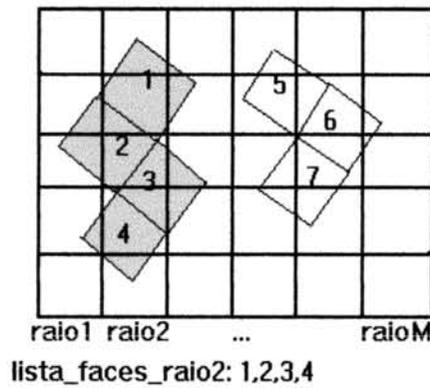


FIGURA 3.13 - Seleção de porções das faces.

Porém, são necessárias, agora, duas etapas. A primeira é calcular o raio-soma computado e a segunda é reconstruir a imagem. O raio-soma computado consiste em, para cada raio, somar as absorções das estruturas atravessadas por ele, para que seja comparado, na segunda etapa, com o raio-soma original, o qual contém a soma real das absorções das estruturas, obtido no momento da aquisição das projeções, na AIP_2D.

Dessa forma, cada raio da grade é considerado duas vezes. A primeira é semelhante a AIP_2D. A cada célula do raio, é chamado o algoritmo de *recorte*, o qual devolve a absorção atual das porções das faces que a interseccionam. Estas absorções, de todas as células do raio, são acumuladas. Na segunda vez que o raio é considerado, a cada célula dele, é chamado novamente o algoritmo de *recorte*. Este, como já explicado anteriormente, trabalha com as várias porções das faces da lista gerada no algoritmo de intersecção e, a cada porção da face, calcula seu peso em relação à célula do respectivo raio. Após calculado o peso, é aplicada a equação (16) de reconstrução bidimensional, apresentada em 3.2.1, a qual está duplicada abaixo:

$$f_m^{(i)} = f_m^{(i-1)} + \left[(P_j - q_j) / \sum_{k=1}^N W_{jk}^2 \right] W_{jm}$$

onde W_{jm} corresponde ao peso recém calculado, P_j corresponde ao respectivo raio-soma calculado em AIP_2D e q_j corresponde ao raio-soma computado na primeira vez que o raio-soma da grade em questão foi considerado. O somatório que divide a equação consiste na soma dos pesos das porções das faces que interseccionam o respectivo raio, computado também durante a primeira vez que este foi considerado. É necessário que se processe em duas etapas, uma vez que, para que se possa reconstruir a imagem, é necessário que o raio soma computado já esteja disponível.

Este processo expresso pela equação (16) é executado para todas as porções das faces interceptadas pela célula do raio, para todas as células do raio. Assim como na AIP_2D, este procedimento se repete para todos os raios da grade.

O procedimento, a seguir, ilustra as duas etapas em que um mesmo raio é considerado:

```

para cada raio
  para cada célula do raio
    recorte1
  fim para
  para cada célula do raio
    recorte2
  fim do para
fim do para

procedimento recorte1      /* primeira etapa */
  para todas as porções das faces da lista
    acha peso
    absorção= peso * intensidade da porção da face
    acumula absorção
    acumula peso
  fim do para
fim do procedimento

procedimento recorte2      /* segunda etapa */
  para todas as porções das faces da lista
    acha peso  $W_{jm}$ 
    
$$f_m^{(i)} = f_m^{(i-1)} + \left[ (P_j - q_j) / \sum_{k=1}^N W_{jk}^2 \right] W_{jm}$$

  fim do para
fim do procedimento
  
```

No algoritmo expresso acima, foram mencionados dois outros procedimentos: *recorte1* e *recorte2*. Ambos são baseados no algoritmo de *recorte* explicado em 3.2.3, porém o primeiro representa a primeira etapa, onde se calcula o raio-soma computado e onde é acumulado o peso (será utilizado como denominador da

equação de reconstrução). O segundo representa a segunda etapa, onde se reconstrói a imagem através da equação (16).

Neste momento tem-se a imagem após se considerar o primeiro ciclo de projeções, a qual serve de suposição inicial para a próxima projeção. A imagem é rotada de acordo com o ângulo utilizado na AIP_2D, para que seja considerado o segundo ciclo, e assim por diante, até que a imagem tenha sido reconstruída. Neste momento, o incremento à ela, expresso na equação (16), é aproximadamente zero.

```

procedimento reconstrói(projeções, dimensões grade)
enquanto número de rotações menor que o limite
enquanto xmin<xmax
    intercepta(xmin, xmin+fatx)
enquanto ymax>ymin
    recorte1(xmin, xmin+fatx, ymax, ymax-fatx)
    acumula absorção
    acumula peso
    ymax = ymax - fatx
fim do enquanto
    raio_soma_computado[xmin] = absorção acumulada
    ymax volta ao seu valor inicial
enquanto ymax>ymin
    recorte2(xmin, xmin+fatx, ymax, ymax-fatx, raio-
    soma-computado, peso acumulado)
    ymax = ymax - fatx
fim do enquanto
    xmin = xmin + fatx
fim do enquanto
    rota objeto(ângulo)
fim do enquanto
fim do procedimento
  
```

As imagens do objeto ilustrado em 3.1, até que o mesmo tenha sido reconstruído, estão demonstradas no anexo A-3, no resultado A-3.3.

3.3 Resumo

Este capítulo trata da reconstrução tomográfica bidimensional. Para que esta ocorra, é necessária a aquisição das projeções da imagem a ser reconstruída. Estas projeções são geradas na AIP_2D (Aquisição das Imagens Bidimensionais Projetadas), descrita neste capítulo.

Tanto na aquisição das projeções, como na reconstrução da imagem, uma grade quadrada é colocada sobre o objeto, a qual representa um conjunto de raios paralelos. Cada coluna da grade representa um raio-soma. Ela é percorrida de cima para baixo, da esquerda para a direita.

A cada raio é executado o procedimento *intercepta* a fim de que só se trabalhe com as porções das faces atravessadas por ele. Os raios são compostos por várias células, as quais são acessadas através do decrementando do y .

No procedimento de AIP_2D, a cada célula do raio, é executado o algoritmo de *recorte*, a fim de que seja determinado o peso de cada porção da face, interna à célula, para o cálculo da absorção da mesma. Quando todas as células do raio tiverem sido consideradas, o valor do primeiro raio-soma estará determinado. Repetindo este processo para os demais raios da grade, encerra-se o primeiro ciclo de projeções.

Rotando-se o objeto, novas projeções serão geradas. Uma vez geradas todas as projeções, estas são levadas ao algoritmo de reconstrução.

No procedimento de reconstrução, cada raio é considerado duas vezes. A primeira corresponde ao cálculo do respectivo raio-soma computado e do denominador da equação (16) de reconstrução bidimensional. A segunda vez que o raio é considerado, a equação (16) é aplicada a cada uma das células do mesmo. Assim como no algoritmo de AIP_2D, a cada célula do raio, é executado o algoritmo de recorte.

Este processo se repete para todos os raios do feixe. Logo, o objeto é rotado para que as demais projeções sejam consideradas.

4 Proposta de Trabalho – 3D

Como comentado no início desta dissertação, o objetivo da mesma, além de estudar os métodos de reconstrução bidimensional de imagem, expressos em 2.1.3, e implementar um deles, é expandi-lo para a terceira dimensão, a fim de que se consiga obter um determinado volume por completo, uma vez que, o mesmo processo executado para a segunda dimensão, é levado para a terceira, garantindo a mesma fidedignidade alcançada em cada fatia bidimensional reconstruída.

Este capítulo é, na verdade, uma extensão do capítulo 3, que trata da implementação bidimensional, uma vez que o processo tridimensional é apenas um complemento ao 2D.

Assim como no caso bidimensional, serão apresentados, neste capítulo: a modelagem dos objetos; a simulação do processo de tomografia computadorizada (SPTC), incluindo a aquisição das imagens projetadas e o processo de reconstrução tridimensional; além de um algoritmo utilizado pela SPTC.

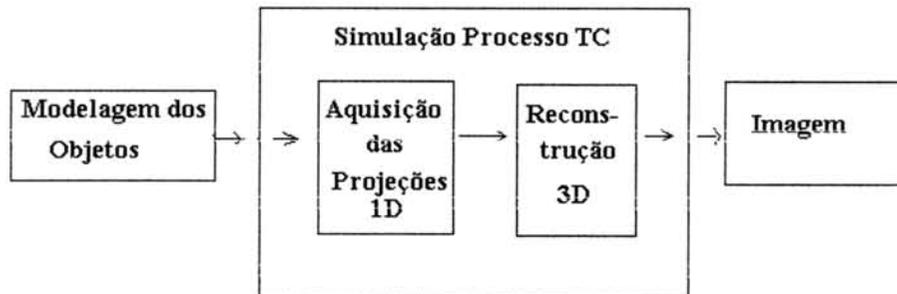


FIGURA 4.1 - Organograma da Proposta 3D.

O organograma acima apresenta o processo pelo qual a proposta tridimensional deste trabalho reconstrói seus objetos. Primeiro os objetos são modelados, a seguir, suas projeções são adquiridas e gravadas. No terceiro momento, então, as projeções são levadas ao algoritmo de reconstrução, a fim de que o objeto tridimensional seja reconstruído.

4.1 Modelagem dos Objetos

Os objetos tridimensionais, assim como os bidimensionais, são considerados como uma matriz geométrica, neste caso, de dimensão três. Ela pode ser encarada como se uma série de imagens bidimensionais fossem empilhadas, munidas, neste momento, de espessura. A figura 4.2 ilustra um destes objetos.

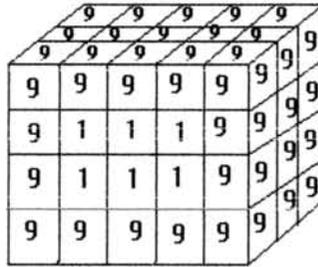


FIGURA 4.2 - Objeto tridimensional modelado.

Este cubo tridimensional é armazenado em um arquivo-objeto, contendo vértices e porções das faces, da seguinte maneira:

```

arquivo OBT
VÉRTICES número de vértices
0 x y z do vértice 0
1 x y z do vértice 1
...
n x y z do vértice n

PORÇÕES DAS FACES número de porções das faces
0 v1 v2 v3 v4 intensidade da porção da face 0
1 v1 v2 v3 v4 intensidade da porção da face 1
...
n v1 v2 v3 v4 intensidade da porção da face n

```

As porções das faces são originalmente *patches* quadrados.

4.2 Simulação do Processo de Tomografia Computadorizada

O objetivo desta etapa é a aquisição das imagens tridimensionais projetadas (AIP_3D), bem como, o processo de reconstrução tridimensional.

4.2.1 Algoritmo de Seção Tridimensional

Este algoritmo tem por finalidade cortar fatias tridimensionais do objeto, para que sejam projetadas e reconstruídas. Nos tomógrafos reais, a fonte de raio-x possui um tamanho limitado, exigindo que o paciente passe aos poucos na máquina, expondo uma parte do seu corpo de cada vez à radiação. O mesmo ocorre com o objeto tridimensional, uma vez que a fonte de raio-x não possui tamanho suficiente para atravessá-lo por completo, como expresso na figura 4.3.

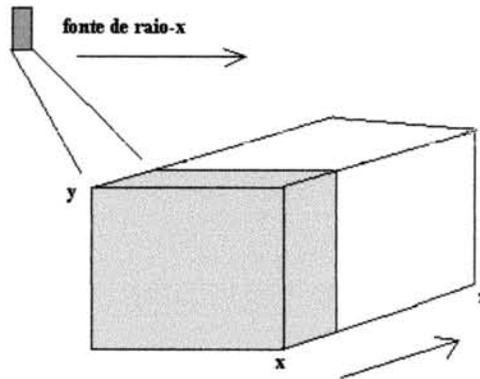


FIGURA 4.3 - Tamanho da fonte de raio-x.

Foi feito uma mescla do algoritmo de intersecção, expresso em 3.2.2, com o de recorte, apresentado em 3.2.3, resultando no procedimento chamado de *secção*. Este algoritmo consiste em testar cada porção da face do objeto em relação ao intervalo em z , o qual representa a espessura da fatia. As que possuírem todos os seus z 's simultaneamente do lado esquerdo ou direito da faixa são descartadas. As que restarem são testadas se possuem todos os seus z 's simultaneamente dentro da faixa limite. Se possuírem, são inseridas na lista de porções das faces, caso contrário, interseccionam o plano em z , devendo ser recortadas, como ilustrado na figura 4.4.

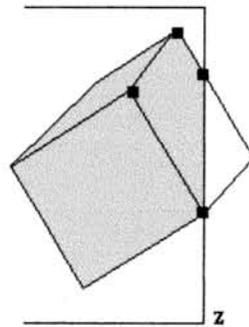


FIGURA 4.4 - Recorte das porções das faces interceptadas pelo plano em z .

O recorte é feito tomando cada vértice da porção da face e testando se o seu z está dentro do intervalo, se estiver, o vértice é inserido na lista de vértices endereçada pela atual porção da face. Porém, se um vértice se encontra dentro da faixa e o da outra extremidade da aresta se encontra fora, é sinal que esta intercepta o plano, e este ponto de intersecção deve ser encontrado e inserido na lista de vértices correspondente à porção da face atual.

O ponto de intersecção da aresta da porção da face com o plano em z é determinado como demonstrado abaixo:

procedimento *recorta porção da face* (x1,y1,z1,x2,y2,z2,zplano)
 $x_n = x_1 + (x_2 - x_1) * (z_{plano} - z_1) / (z_2 - z_1)$
 $y_n = y_1 + (y_2 - y_1) * (z_{plano} - z_1) / (z_2 - z_1)$
 $z_n = z_{plano}$

O processo completo está demonstrado abaixo, de forma estruturada:

procedimento *secção*(zi, zf)
para cada porção da face
 se todos os z's são menores zi ou maiores que zf
 descarta porção da face
senão
 se todos os z's estão entre [zi,zf]
 insere porção da face na lista de porções das faces
senão
para cada vértice
 se vértice dentro do intervalo
 insere na lista de vértices da porção da face
 se vértice anterior está fora do intervalo
recorta porção da face
 insere ponto de intersecção na lista de vértices, uma posição anterior
fim do se
fim do se
senão
 se vértice anterior está dentro do intervalo
recorta porção da face
 insere ponto de intersecção na lista de vértices, na posição atual
fim do se
fim do para
fim do para
fim do procedimento

A lista de porções das faces, mencionada no procedimento acima, corresponde a uma lista encadeada cujas posições endereçam uma outra lista encadeada que corresponde aos vértices de cada porção da face. Estes vértices são inseridos à lista ordenados no sentido horário.

As porções das faces que são entrada a este procedimento devem estar ordenadas em z, a fim de que não seja necessário processar com as que se encontram muito afastadas do intervalo em z.

4.2.2 Aquisição das Imagens Projetadas

Esta etapa é necessária, como já comentado anteriormente, pois gera as projeções do objeto para que o mesmo possa ser reconstruído. Optou-se pela simulação da mesma porque assim se pode conhecer melhor o processo, podendo ser monitorado, permitindo um maior controle na posterior reconstrução. Além disso, como não se costuma rotar o paciente dentro do aparelho e, nem tão pouco, a fonte de raio-x faz rotação tridimensional, a solução adotada foi sua simulação.

Assim como no caso bidimensional, ao invés de se rotar a fonte de raio-x, optou-se por rotar o objeto, permanecendo a fonte fixa. Porém, o objeto passa a ter espessura, enquanto que a fonte permanece inalterada. Isso implica a necessidade de dividir o volume em partes menores, que correspondam à largura da fonte. Deste modo, a fonte avança sobre o objeto, em determinado passo, tratando o mesmo como se fosse formado por uma série de volumes menores, como ilustrado na figura 4.5.

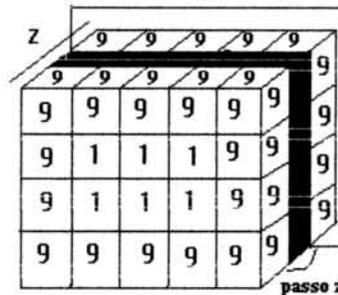


FIGURA 4.5 - Plano de corte.

Cada uma das partes do volume maior é submetida ao processo de AIP_3D (Aquisição de Imagens Tridimensionais Projetadas), individualmente. Isto consiste em determinar um passo em z e, ao final de cada avanço, o objeto é interseccionado pelo plano correspondente ao atual z , gerando uma fatia do mesmo, cuja espessura corresponde ao tamanho do passo, como ilustrado na figura 4.5. Este novo objeto é submetido ao processo de AIP_3D, sendo gerada sua primeira projeção. Logo após, é dado mais um passo em z , e nova fatia é considerada, gerando a primeira projeção da mesma. Este processo continua até que todo objeto tenha sido submetido à primeira projeção.

Neste momento o objeto tridimensional completo é rotado para que nova projeção seja gerada. Novamente os passos em z são efetuados e, após cada um, o novo objeto gerado é submetido ao processo de AIP_3D, para que a segunda projeção do mesmo seja gerada. Novos passos em z são efetuados, gerando a segunda projeção para todas as partes do objeto. Este processo se repete para todos os ângulos pelos quais o objeto é rotado.

O processo de intersecção do objeto pelo plano em z ocorre como explicado em 4.2.1. Esse algoritmo testa cada porção da face do objeto, desprezando as que se encontram fora do intervalo em z , determinado pelo seu passo. As porções das

faces que interseccionam o plano são recortadas ao encontrar o ponto de intersecção de cada uma de suas arestas com o mesmo, como ilustrado na figura 4.6.

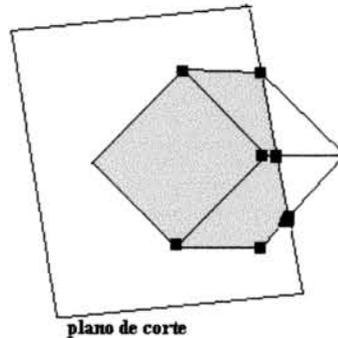


FIGURA 4.6 - Inteseccção das porções das faces com o plano de corte.

O algoritmo gera uma lista encadeada de porções das faces contendo, em cada posição, o endereço para a lista dos respectivos vértices. Esta lista é a entrada para o algoritmo de AIP_3D.

Assim como no procedimento bidimensional, uma grade quadrada é colocada sobre o objeto, cujas dimensões são determinadas de acordo com o x máximo, o x mínimo, o y máximo e o y mínimo do objeto. É desnecessário, neste momento, o controle da dimensão z , uma vez que as porções das faces inseridas na lista de porções das faces, proveniente do algoritmo de secção, respeitaram a faixa estabelecida pelo passo em z . Sendo assim, o processo é o mesmo que o bidimensional, ou seja, a grade é percorrida de cima para baixo, da esquerda para a direita. A cada passo em x (raio), o algoritmo de intersecção das porções das faces com a faixa em x , correspondente ao raio em questão, é executado, gerando uma outra lista de porções das faces, como ilustrado na figura 4.7.

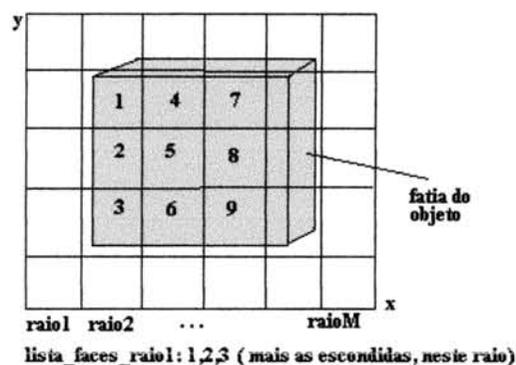


FIGURA 4.7 - Porções das faces da fatia interceptadas pelo raio1.

A cada célula do respectivo raio, esta lista de porções das faces é levada ao algoritmo de *recorte* descrito em 3.2.3, para que a soma das absorções das estruturas

presentes nesta célula seja determinada e somada às das demais células do raio. Este processo ocorre para todos os raios da grade de raios.

Neste momento, um novo passo é dado em z , gerando um novo plano que será levado ao algoritmo de *secção* para que a lista contendo as porções das faces presentes neste intervalo em z seja gerada. Novamente a grade de raios é colocada sobre o objeto, para que se gere a primeira projeção da fatia do objeto em questão. Este processo continua até que todo o objeto tenha sido projetado.

O objeto é então rotado, para que se inicie o segundo ciclo de projeções. Uma vez rotado, avança-se um passo em z , leva-se o plano correspondente ao algoritmo de *secção*, para que seja gerada a primeira fatia do objeto. Sobre ela é colocada a grade de raios, e o processo descrito acima se repete, a fim de gerar a segunda projeção da mesma. Novos passos em z são efetuados, até que todo o objeto tenha sido projetado pela segunda vez.

Novas rotações com o objeto são executadas, de modo a que novas projeções do objeto sejam geradas.

Estas projeções são armazenadas em uma estrutura de dados a fim de serem utilizadas pelo algoritmo de reconstrução. Esta estrutura consiste em uma lista de projeções, cujas posições endereçam uma lista de fatias cujas posições endereçam uma lista de raios, como ilustrado na figura 4.8.

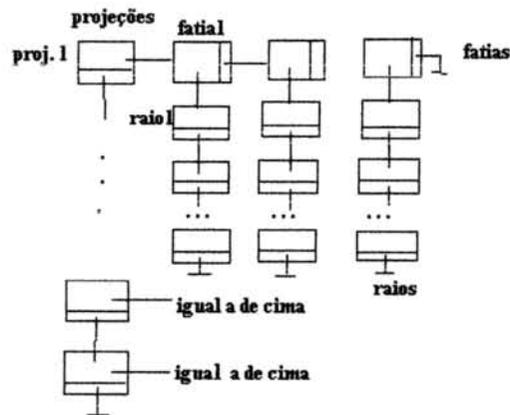


FIGURA 4.8 - Estrutura para o armazenamento das projeções.

O processo de AIP_3D está expresso abaixo, de forma estruturada:

```

procedimento AIP_3D
  acha envelope do objeto
  calcula fatx, faty
  calcula passo em z
  enquanto número de rotações menor que o limite
    enquanto  $z < z_{max}$ 
      secção( $z, z + passo_z$ )
      enquanto  $x_{min} < x_{max}$ 
        intercepta( $x_{min}, x_{min} + fatx, lista\_porções\ das\ faces$ )
        enquanto  $y_{max} > y_{min}$ 
          recorte( $x_{min}, x_{min} + fatx, y_{max}, y_{max} - faty$ )
          acumula absorção
           $y_{max} = y_{max} - faty$ 
        fim do enquanto
        raio_soma_medido[ $x_{min}$ ] = absorção acumulada
         $x_{min} = x_{min} + fatx$ 
      fim do enquanto
       $z = z + passo_z$ 
    fim do enquanto
    rota objeto(ângulo)
  fim do enquanto
fim do procedimento
  
```

Os procedimentos *recorte* e *intercepta* estão ilustrados na AIP_2D e o *intercepta* se encontra em 3.2.2.

4.2.3 Algoritmo de Reconstrução

Nesta etapa, é importante que se mantenha o passo em z , efetuado na AIP_3D, assim como as dimensões da grade de raios. O objeto deve ser rotado obedecendo aos mesmos ângulos utilizados na aquisição das projeções. Estes cuidados são necessários para que as informações presentes nas projeções sejam distribuídas às suas respectivas estruturas.

Assim como na AIP_3D, o objeto deve ser avançado em z , de modo que se obtenha uma fatia do mesmo. Este intervalo em z é levado ao algoritmo de *secção*, para que este gere uma lista contendo as porções das faces que se encontram presentes neste intervalo. A grade de raios é então considerada, avançando de x_{min} , até x_{max} , através do $fatx$, que representa a largura do mesmo. A cada raio, o procedimento *intercepta* é executado, de modo a gerar uma nova lista de porções das faces (presentes na estrutura do objeto), contendo apenas as que são atravessadas pelo raio, representado por $x_{min}, x_{min} + fatx$. Como já dito anteriormente, o intervalo em z não necessita ser

testado, uma vez que já foi considerado pelo procedimento de *secção*, como ilustrado na figura 4.9.

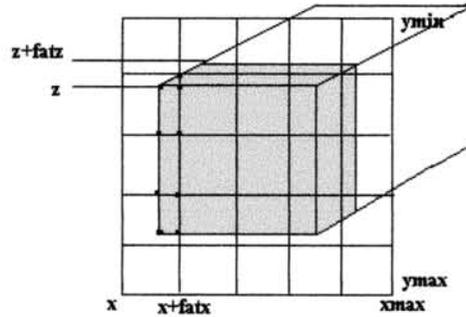


FIGURA 4.9 - Grade de raios sobre fatia do objeto.

Neste momento se inicia o passo em *y*, para que cada célula do raio seja considerada. Assim como no caso bidimensional, cada raio é considerado duas vezes. Na primeira é calculado o raio_soma computado, além do peso das células serem acumuladas, pois servirão de denominador para o processo de reconstrução. Na segunda vez em que cada raio é considerado, é efetuada a reconstrução da imagem através da equação (16), apresentada em 3.2.1. Este processo é apresentado abaixo, de forma estruturada:

```

para cada raio
  para cada célula do raio
    recorte1
  fim para
  para cada célula do raio
    recorte2
  fim do para
fim do para

procedimento recorte1      /* primeira etapa */
  para todas as porções das faces da lista
    acha peso
    absorção = peso * intensidade da porção da face
    acumula absorção
    acumula peso
  fim do para
fim do procedimento
  
```

procedimento *recorte2* /* segunda etapa */
para todas as porções das faces da lista
 acha peso W_{jm}

$$f_m^{(i)} = f_m^{(i-1)} + \left[(P_j - q_j) / \sum_{k=1}^N W_{jk}^2 \right] W_{jm}$$

fim do para
fim do procedimento

Este processo é repetido para todos os raios da grade, de forma que a primeira projeção, gerada na AIP_3D, seja considerada.

Um novo passo em z deve ser executado, a fim de que uma nova fatia do objeto seja inserida na lista de porções das faces, através do algoritmo *secção*. Esta nova lista é então levada à grade de raios. Para cada raio da mesma, o algoritmo *intercepta* é executado, gerando uma nova lista de porções das faces, a qual contém apenas as porções das faces presentes no intervalo em x que representa o raio em questão. A cada célula do mesmo, na primeira vez em que é considerado, o procedimento *recorte1* é executado, gerando, ao final de todas as porções das faces, o raio_soma computado referente às mesmas. Novamente o raio é considerado, executando agora, para cada célula do mesmo, o algoritmo *recorte2*, o qual irá distribuir, às células, seu respectivos incrementos, através da equação (16), com base no raio-soma computado anteriormente e no correspondente raio-soma medido, gerado na AIP_3D. Este sistema é repetido para todos os raios, de modo que a primeira projeção desta fatia do objeto seja considerada.

Outros passos em z são efetuados, de modo que a primeira projeção de todo o objeto tenha sido considerada, obtendo-se então o objeto após a primeira projeção.

Para que as demais projeções sejam consideradas, é necessário rotar o objeto. A cada rotação, o processo descrito acima é executado, até que o objeto já tenha sido rotado em todas as direções, correspondentes às da AIP_3D. Neste momento, o objeto tridimensional encontra-se reconstruído em todos os *voxels* do grande cubo.

É importante ressaltar que a reconstrução ocorre em todas as faces que compõem a estrutura do objeto, não tendo portanto nenhuma relação com sua posterior visualização.

Este processo é demonstrado abaixo, de forma estruturada:

```

procedimento reconstrói_3D(projeções, dimensões da grade,
passo em z)
  enquanto número de rotações menor que o limite
    enquanto  $z < z_{max}$ 
      secção( $z, z + passo_z$ )
      enquanto  $x_{min} < x_{max}$ 
        intercepta( $x_{min}, x_{min} + fat_x, lista\_porções\ das\ faces$ )
        enquanto  $y_{max} > y_{min}$ 
          recorte1( $x_{min}, x_{min} + fat_x, y_{max}, y_{max} - fat_y$ )
          acumula absorção
          acumula peso
           $y_{max} = y_{max} - fat_y$ 
        fim do enquanto
        raio_soma_computado[ $x_{min}$ ] = absorção acumulada
         $y_{max}$  volta ao seu valor inicial
        enquanto  $y_{max} > y_{min}$ 
          recorte2( $x_{min}, x_{min} + fat_x, y_{max}, y_{max} - fat_y,$ 
            raio-soma-computado, peso acumulado)
           $y_{max} = y_{max} - fat_y$ 
        fim do enquanto
         $x_{min} = x_{min} + fat_x$ 
      fim do enquanto
       $z = z + passo_z$ 
    fim do enquanto
    rota objeto(ângulo)
  fim do enquanto
fim do procedimento

```

Uma vez reconstruído o objeto, é necessária a utilização de um de um sistema de visualização, para que se possa explorar esta massa de dados gerada.

No anexo A-3 encontram-se alguns exemplos de fatias reconstruídas.

4.3 Resumo

Este capítulo trata da reconstrução tridimensional dos objetos. Estes são tratados como se fossem uma série de fatias tridimensionais do objeto. Cada fatia é processada individualmente, tanto no procedimento de AIP_3D, como no de reconstrução.

As fatias são determinadas, fixando um passo em z , o qual representa a espessura da mesma. A cada passo em z , é determinado um plano que irá seccionar o objeto, através do procedimento *secção*.

Cada fatia do objeto é tratada como se fosse uma secção bidimensional, devido ao fato de que, quando foi determinado o intervalo em z , todas as fatias pertinentes a ele foram selecionadas, bastando, agora, processá-las sob a grade de raios.

O processo de percorrer a grade é o mesmo executado na etapa bidimensional, tanto na AIP_2D, como na reconstrução da imagem.

Após a ser processada a primeira fatia, é efetuado um passo em z , gerando uma nova fatia. Quando todo o objeto tiver sido considerado, é efetuada a rotação do mesmo.

Nesta etapa tridimensional, primeiro todo o objeto é projetado (todas as fatias), em todas as direções, para, então, começar a reconstrução.

5 Análise dos Resultados

Este capítulo trata da análise dos dados resultantes da proposta discutida nesta dissertação e apresentados no Anexo 3. Ele está dividido em duas partes: a etapa bidimensional e a tridimensional. São feitos comentários quanto à convergência do processo de reconstrução adotado.

5.1 Etapa Bidimensional

A etapa bidimensional consiste em gerar as projeções de uma imagem bidimensional e reconstruí-la. Em ambos os casos, é necessário que o objeto seja rotado, como explicado no capítulo três. O ângulo escolhido, bem como o número de raios em relação ao número de células pertencentes à imagem são decisivos para a convergência do sistema.

Independente do número de rotações, não é necessário que o objeto seja rotado mais que 180° . Isto porque o que importa, em cada rotação, é gerar um raio-soma linearmente independente dos anteriores. Isto deve-se ao fato de que as projeções são somatórios, o que independe da ordem em que os valores são somados. Portanto, as projeções levadas a ângulos maiores que 180° são redundantes.

No anexo A-3, foi apresentada uma imagem A-3.1, a qual foi reconstruída no quinto ciclo. O mesmo objeto foi submetido ao processo de aquisição de projeções utilizando 8 ciclos, diminuindo-se o ângulo entre as projeções. Notou-se que a imagem, em determinado momento, apresentou números negativos, como demonstrado em A-3.2.

Este fato ocorre, devido ao fato do algoritmo de reconstrução tratar-se de um sistema de equações lineares. Desta forma, deve-se escolher uma relação entre raio e ângulos de forma que as equações que representam os raios sejam linearmente independentes, não repetindo informações de uma equação para outra.

Segundo Rosenfeld [ROS 82], se dois hiperplanos, que representam duas equações de (12), em 3.2.1, fossem perpendiculares, um em relação ao outro, seria possível, dada uma suposição inicial e duas constantes, partir à correta solução em apenas dois passos. Por outro lado, se dois hiperplanos possuem um ângulo muito pequeno entre si, dependendo da suposição inicial, o número de raios das equações de (11), encontradas em 3.2.1, a considerar pode assumir um valor elevado, antes que a solução correta seja encontrada. Dessa forma, os ângulos influenciam na razão de convergência da solução. Se os M hiperplanos de (11) pudessem ser trabalhados de forma ortogonal, a solução correta seria encontrada na primeira passada pelas M equações de (11) [ROS 82].

Outro fator que interfere na convergência dos resultados é a relação entre o número N de células da imagem e o número M de raios. Tratando-se de uma célula

com N elementos, ou seja, N variáveis, são necessários M raios (equações), sendo $N=M$, para que exista uma única solução, tratando-se de um sistema de equações lineares.

Caso o número de raios M seja maior que o total de células N da imagem, $M>N$ em (11), isto é, número de equações maior que o número de variáveis, os raios-soma podem ser alterados por ruídos, interferências, as quais não serão constantes, o que faz com que o sistema não possua uma única solução. Neste caso, a solução não converge para um único ponto no espaço N -dimensional, mas oscilará na vizinhança do mesmo [ROS 82].

Se $M<N$, não existe uma única solução para o conjunto de equações lineares em (11) e, de fato, é possível um número infinito de soluções. Se apenas a primeira equação de (12) for considerada, para dois valores desconhecidos f_1 e f_2 , a solução poderia estar em qualquer valor da linha correspondente à equação. Segundo Rosenfeld [ROS 82], Tanabe provou que a melhor solução seria projetá-la na linha e considerar o resultado como solução. Esta solução corresponde ao ponto da linha mais próximo à suposição inicial.

É necessário, então, que o número N de células seja igual ao número M de raios, e que estes estejam dispostos de modo que sejam todos linearmente independentes entre si.

Supondo uma imagem representada por uma grade 5×5 . Cada coluna da grade pode ser atravessada pelo raio, se modo que este passe pelo meio da célula. Os demais vinte raios podem ser divididos em quatro conjuntos, que correspondem a quatro novos ciclos. Cada raio do ciclo atravessará a imagem como recém descrito. É importante que o ângulo entre os raios deste novos ciclos seja tal que os torne independentes. Uma forma é dividir 180 pelo número de ciclos, para obter-se o ângulo desejado. Dessa forma, uma matriz imagem 5×5 , terá cinco ciclos, com cinco raios em cada um, levados ao ângulo de 36° . Os resultados presentes em A-3.1, seguiram estes valores.

No anexo A-3, o resultado A-3.3, ilustra a reconstrução de uma imagem no primeiro ciclo de projeções. Isto ocorreu devido ao fato de ter-se associado a cada projeção uma informação adicional. Sabendo-se de início que a imagem possuía uma área zerada, esta informação foi levada às projeções, as quais zeraram o valor do peso dessas células. Se este procedimento não tivesse ocorrido, a reconstrução se daria no quinto ciclo, como ilustrados em A-3.4.

5.2 Etapa Tridimensional

Esta etapa consiste em uma série de etapas bidimensionais, cujo número corresponde ao número de fatias do objeto. Na verdade, a abordagem matemática é utilizada para reduzir o problema tridimensional para bidimensional, assumindo a ausência da espessura nas fatias. Este problema bidimensional é, então, dividido em uma série de problemas unidimensionais, projeções unidimensionais.

Para que se calcule a eficiência e o custo desta etapa, basta fazê-lo para o caso bidimensional, e considerá-lo para todas as fatias do objeto tridimensional em questão.

A reconstrução do objeto tridimensional, neste trabalho, foi considerada como se fossem reconstruídas várias imagens bidimensionais e, colocadas, uma sobre a outra, formando um cubo de intensidades. Cada imagem deste cubo foi comparada com a respectiva fatia original para provar sua fidelidade.

Segue abaixo uma seqüência de imagens que representa a reconstrução da primeira fatia de um objeto tridimensional:

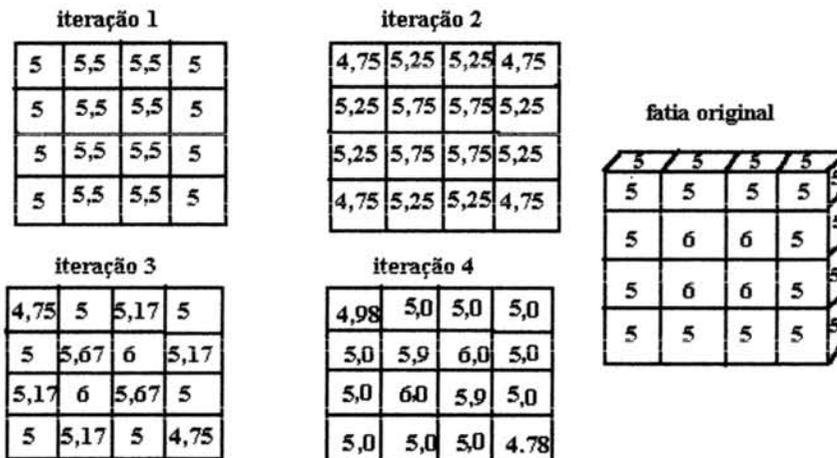


FIGURA 5.1 - Resultados.

Foram utilizados quatro raios, em cada projeção, num total de quatro projeções. A suposição inicial à imagem utilizada foi uma matriz zerada. Alguns arredondamentos foram efetuados.

6 Conclusões

O fato de as estruturas anatômicas do corpo humano serem inerentemente tridimensionais, sugeriu um aumento nas pesquisas referentes à reconstrução tomográfica tridimensional.

Porém, a maioria dos métodos utilizados para se obter a visualização tridimensional de imagens tomográficas, são procedimentos que, através da imagem de secções do corpo humano, montam uma representação visual tridimensional para as mesmas, utilizando técnicas de interpolação entre as fatias. Se a distância entre as fatias for grande, pode haver perda na confiabilidade de seus valores.

Com o método desenvolvido neste trabalho, não apenas as imagens das secções do corpo humano são reconstruídas, mas todo o volume. Isto ocorre devido ao fato de se utilizar, para a reconstrução de todo o volume, a mesma técnica utilizada pelos tomógrafos para gerar as fatias bidimensionais do corpo.

Primeiramente, foi desenvolvida a reconstrução bidimensional de imagens, através do Método das Projeções, o qual reconstrói a imagem incrementalmente, fazendo uma distribuição dos valores obtidos nas projeções. Cada projeção é distribuída à imagem através do valor médio que representa, nesta direção.

O processo de distribuição torna desnecessária a interpolação entre as fatias do volume. Como já explicado anteriormente, o volume do objeto é tratado como várias fatias do mesmo, cuja espessura corresponde ao tamanho do passo dado em z. Quando a fatia foi gerada, um intervalo em z foi determinado, e as porções das faces do objeto que estavam contidas nele, foram selecionadas. Dessa forma, pode-se tratar a fatia como se não possuísse espessura, pois o processo de distribuição irá considerar o valor de todas as células presentes no intervalo, a fim de obter o valor médio a ser distribuído naquela direção.

Dessa forma, independente da espessura da fatia, todas as porções das faces, presentes na mesma, farão parte do processo. Isto significa que o problema tridimensional pode ser dividido numa série de problemas bidimensionais, os quais podem ser tratados independentemente.

Foram desenvolvidos um protótipo bidimensional e um tridimensional. É claro que o 3D se utiliza do 2D, mas é possível suas utilizações de forma independente. Cada protótipo gera dois produtos: os arquivos de projeções dos objetos e os arquivos das imagens reconstruídas (incluindo as imagens intermediárias, após cada ciclo de projeções ter sido considerado).

O protótipo permite:

- a aquisição de projeções de objetos bidimensionais, gravadas em arquivos;
- a aquisição de projeções de objetos tridimensionais, gravadas em arquivos;
- a reconstrução de objetos bidimensionais, gerando arquivos;

- a reconstrução de objetos tridimensionais, gerando arquivos;
- que os usuários definam a espessura das fatias geradas e
- que os usuários definam qual o erro aceitável no processo de reconstrução.

As imagens geradas pelos protótipos são gravadas em arquivos, além dos algoritmos terem sido desenvolvidos utilizando recursos básicos computacionais, o que permite a portabilidade destes protótipos a outras arquiteturas, sem muitas adaptações.

Nesta dissertação, foram tratadas apenas projeções paralelas. Poder-se-ia, no futuro, trabalhar com projeções cujos raios partem de um determinado ponto (*fan-beam*). Além disso, seria interessante que outros tipos de tomografia, além da por transmissão, fossem trabalhadas.

Poder-se-ia, também, no futuro, adicionar à solução um tipo de informação inicial sobre a imagem. Supondo ser conhecido inicialmente que a imagem $f(x,y)$ é negativa, então, em cada uma das soluções, sucessivamente obtida pela equação (16), presente em 3.2.1, um possível conjunto de valores pode ser igual a zero. É possível, simplesmente, incorporar a informação de que a função $f(x,y)$ é zero fora de uma certa área, caso isto seja conhecido.

A eficiência deste procedimento acima foi comprovada em um teste efetuado com uma imagem, cujos resultados são analisados no capítulo 5.

Anexo 1

GERAÇÃO DE PROJEÇÕES POR INTEGRAÇÃO

Como dito em 2.1.2, cada raio-soma do feixe representa a atenuação total de raio-x, ao longo do mesmo.

Cada raio-soma pode ser computado integrando-se as densidades, ou absorções, da região de interesse ao longo do mesmo, o que representará a soma das densidades das estruturas atravessadas por ele. Esta integral ao longo do raio, como já dito anteriormente, pode ser chamada de raio-soma [VIA 90].

Pode-se considerar a região de interesse como uma função bidimensional $f(x,y)$. Com a integração desta função, ao longo da direção do raio, obtém-se a projeção.

A equação da linha A, na figura A1.1, é dada por:

$$x \cos \theta + y \sin \theta = t_1 \quad (21)$$

onde t_1 é a distância perpendicular entre a linha e a origem. A integral da função $f(x,y)$, ao longo desta linha, pode ser expressa por:

$$P_\theta(t_1) = \int_{\text{raioAB}} f(x, y) ds = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(x \cos \theta + y \sin \theta - t_1) dx dy \quad (22)$$

onde t representa a linha do feixe. Logo, sendo a função P_θ uma função de t , como expresso na figura A1.1, para um dado ângulo θ , define a projeção paralela de $f(x,y)$, neste ângulo. A função bidimensional $P_\theta(t)$ é também chamada de Transformada de Randon de $f(x,y)$ [ROS 82].

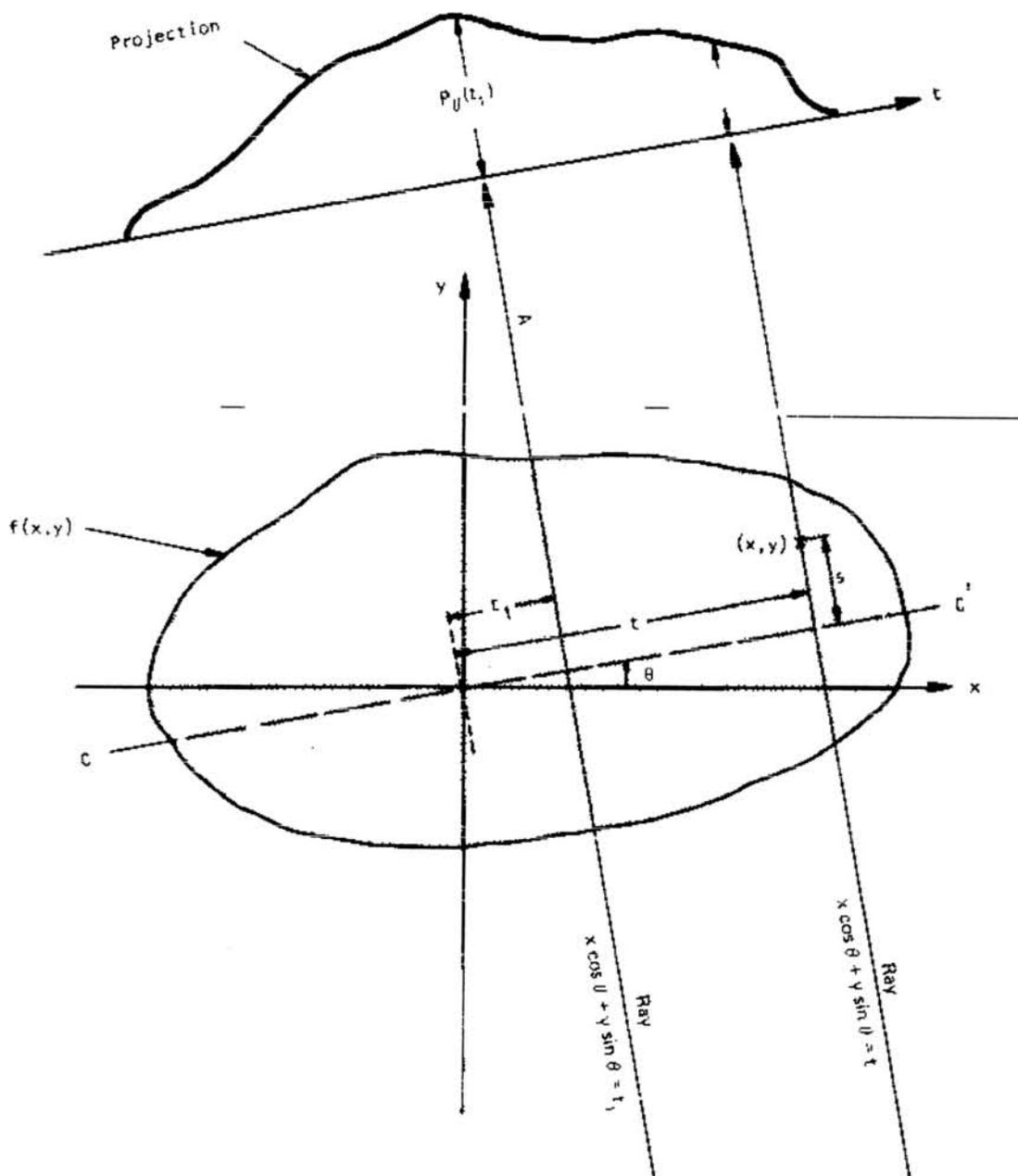


FIGURA A1.1 - Projeção $P_\theta(t)$ de $f(x,y)$ no ângulo θ [ROS 82].

O vetor normalizado \vec{OU} , ao longo de \vec{W}_1 , é dado pela expressão vetorial:

$$\vec{OU} = \vec{W}_1 / \sqrt{\vec{W}_1 \cdot \vec{W}_1} \quad (23)$$

A distância perpendicular do hiperplano à origem, igual ao comprimento de \vec{OA} , na figura A-2.1, é dado por:

$$\begin{aligned} |\vec{OA}| &= \vec{OU} \cdot \vec{OC} = \vec{OU} \cdot \vec{f}, \\ &= \left(\vec{OU} \cdot \vec{f} \right) \sqrt{\vec{W}_1 \cdot \vec{W}_1} / \sqrt{\vec{W}_1 \cdot \vec{W}_1}, \\ &= \left(1 / \sqrt{\vec{W}_1 \cdot \vec{W}_1} \right) \left(\sqrt{\vec{W}_1 \cdot \vec{W}_1} \vec{OU} \cdot \vec{f} \right) \end{aligned}$$

Substituindo-se \vec{OU} com a (23), tem-se:

$$\begin{aligned} &= \left(1 / \sqrt{\vec{W}_1 \cdot \vec{W}_1} \right) \left(\vec{W}_1 \cdot \vec{f} \right), \\ &= \frac{P_1}{\sqrt{\vec{W}_1 \cdot \vec{W}_1}} \quad (24) \end{aligned}$$

Para gerar $\vec{f}^{(1)}$, deve-se subtrair o vetor \vec{GH} do vetor $\vec{f}^{(0)}$:

$$\vec{f}^{(1)} = \vec{f}^{(0)} - \vec{GH} \quad (25)$$

onde o comprimento do vetor \vec{GH} é dado por:

$$|\vec{GH}| = |\vec{OF}| - |\vec{OA}| = \vec{f}^{(0)} \cdot \vec{OU} - |\vec{OA}| \quad (26)$$

Substituindo-se, esta equação, por (23) e (24), tem-se:

$$|\vec{GH}| = \left(\vec{f}^{(0)} \cdot \vec{W}_1 - P_1 \right) / \sqrt{\vec{W}_1 \cdot \vec{W}_1} \quad (27)$$

Como a direção de \vec{GH} é a mesma do vetor unitário \vec{OU} , vem:

$$\vec{GH} = |\vec{GH}| \vec{OU} = \vec{W}_1 \left(\vec{f}^{(0)} \cdot \vec{W}_1 - P_1 \right) / \left(\vec{W}_1 \cdot \vec{W}_1 \right) \quad (28)$$

Substituindo-se (28) em (25), obtém-se (13), a qual está duplicada a seguir:

$$\vec{f}^{(1)} = \vec{f}^{(0)} - \vec{W}_1(\vec{W}_1 \cdot \vec{f}^{(0)} - P_1) / (\vec{W}_1 \cdot \vec{W}_1)$$

Anexo 3

Resultados Obtidos

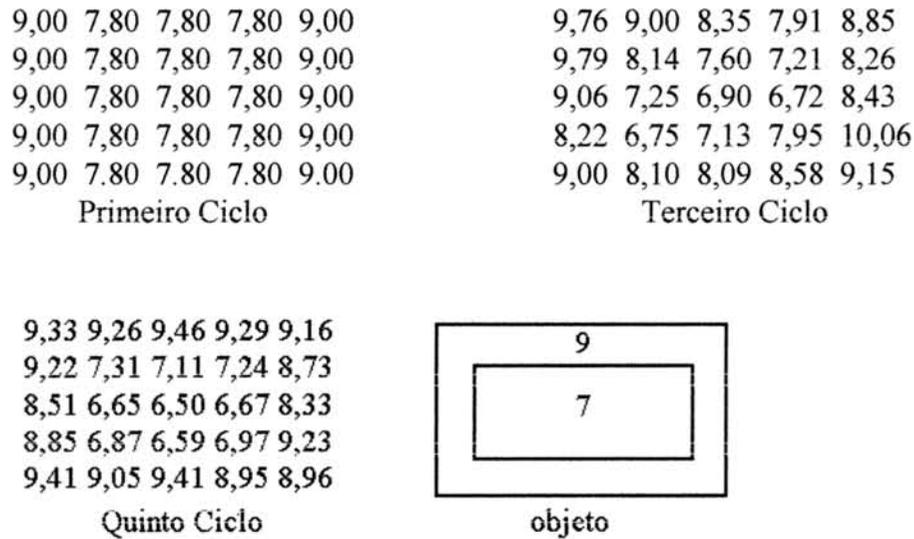


FIGURA A3.1 - Imagem 5x5, onde foram executados cinco ciclos

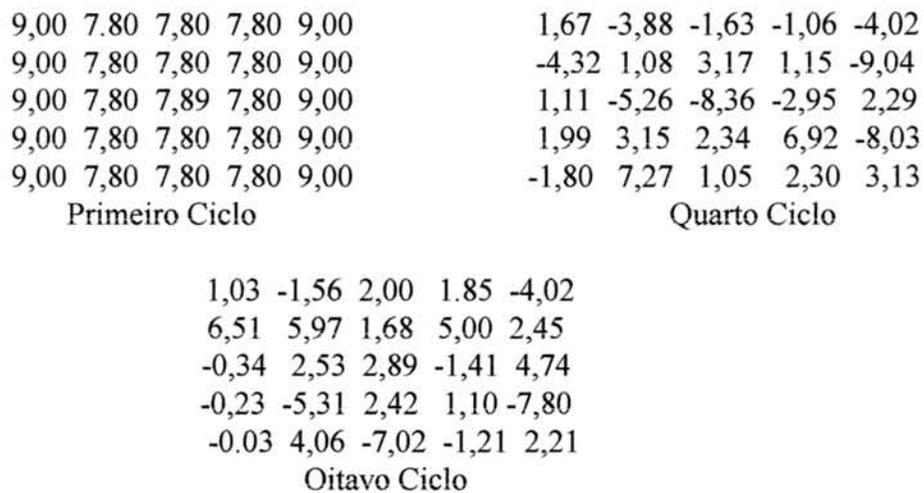
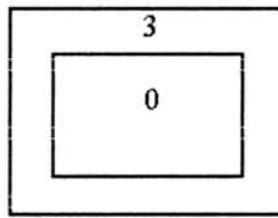


FIGURA A3.2 - Imagem 5x5, onde foram executados oito ciclos. A imagem foi alterada por ruído.



3,00	3,00	3,00	3,00	3,00
3,00	0,00	0,00	0,00	3,00
3,00	0,00	0,00	0,00	3,00
3,00	0,00	0,00	0,00	3,00
3,00	3,00	3,00	3,00	3,00

Primeiro Ciclo

FIGURA A3.3 - Imagem 5x5 reconstruída no primeiro ciclo, devido a algum conhecimento prévio da mesma.

7,00	5,00	3,80	5,00	7,00
7,00	5,00	3,80	5,00	7,00
7,00	5,00	3,80	5,00	7,00
7,00	5,00	3,80	5,00	7,00
7,00	5,00	3,80	5,00	7,00

Primeiro Ciclo

3,49	3,39	3,68	3,43	3,99
3,78	0,77	0,17	0,36	2,88
2,46	-0,46	-0,04	-0,70	3,10
2,90	0,19	-0,55	-0,03	3,40
3,61	3,18	3,40	2,93	3,20

Quinto Ciclo

FIGURA A3.4 - Imagem 5x5, onde foram executados cinco ciclos.

7,00	7,00	7,00	7,00	7,00
7,00	4,00	2,00	4,00	7,00
7,00	2,00	1,00	2,00	7,00
7,00	3,00	2,00	3,00	7,00
7,00	7,00	7,00	7,00	7,00

Imagem Original

7,52	7,70	7,22	8,81	8,06
7,03	4,53	2,85	4,20	6,70
7,03	2,17	1,35	1,83	6,07
6,23	3,63	1,59	3,69	8,18
7,59	7,54	7,17	7,08	6,97

Quinto Ciclo

FIGURA A3.5 - Imagem 5x5, onde foram executados cinco ciclos. Densidades variadas.

Bibliografia Referenciada

- [AXE 83] AXEL, Leon; ARGER, Peter H.; ZIMMERMAN, Robert A. Applications of Computerized Tomography to Diagnostic Radiology. **Proceedings of the IEEE**, New York, v.71, n.3, Mar. 1983.
- [KNO 83] KNOLL, Glenn F.. Single-Photon Emission Computed Tomography. **Proceedings of the IEEE**, New York, v.71, n.3, Mar. 1983.
- [MAT 93] MATTOS, Patrícia N.. **Um Estudo Sobre Reconstrução Tomográfica a Partir de Projeções**: trabalho individual. Porto Alegre: CPGCC da UFRGS, 1993. (TI - 332).
- [NET 89] NETTO, João C. **CV2D - Operador de Convolução para Processamento de Imagens em Tempo Real**. Porto Alegre: CPGCC da UFRGS, 1989. Dissertação de Mestrado.
- [NEW 79] NEWMAN, William M.; SPROULL, Robert F. **Principles of Interactive Computer Graphics**. [S.l.]: McGraw-Hill, 1979. 541p.
- [PHE 77] PHELPS, Michael E. Emission Computed Tomography. **Seminars in Nuclear Medicine**, [S.l.], v.7, n.4, Oct. 1977.
- [POG 77] TER-POGOSSIAN, Michael M. Basics Principles of Computed Axial Tomography. **Seminars in Nuclear Medicine**, [S.l.], v.7, n.2, Apr. 1977.
- [ROS 82] ROSENFELD, Azriel; KAK, Avinash C. **Digital Picture Processing**, [S.l. : s.n.], 1982.
- [ROX 90] ROXO, Maria Fernanda E. B. Convolucionador de Imagens. In: SIMPÓSIO BRASILEIRO DE COMPUTAÇÃO GRÁFICA E PROCESSAMENTO DE IMAGENS, SIBGRAPI, 3., 1990, Gramado. **Anais...** Porto Alegre: SBC/UFRGS, 1990.
- [VIA 90] VIANA, Márcia M. et al. RECONV: Reconstrução de Imagens Tomográficas. In: JORNADA EPUSP/IEEE EM COMPUTAÇÃO VISUAL, 1990, São Paulo. **Anais...** São Paulo: EPUSP, 1990.

Bibliografía Consultada

- [BAT 83] BATES, R.H.T.; GARDEN, K.L.; PETERS, T. M. Overview of Computerized Tomography with Emphasis on Future Developments. **Proceedings of the IEEE**, New York, v.71, n.3, Mar. 1983.
- [BLO 83] BLOCH, Peter; UDUPA, Jayaram K. Application of Computerized Tomography to Radiation Therapy and Surgical Planning. **Proceedings of the IEEE**, New York, v.71, n.3, Mar. 1983.
- [CEN 83] CENSOR, Yair. Finite Series-Expansion Reconstruction Methods. **Proceedings of the IEEE**, New York, v.71, n.3, Mar. 1983.
- [CRE 93] CREASE, Robert P. Biomedicine in The Age of Imagin. **Science**, New York, v.261, July 1993.
- [ELV 92] ELVINS, T. Todd. A Survey of Algorithms for Volume Visualization. **Computer Graphics**, New York, v.26, n.3, Aug. 1992.
- [FOL 90] FOLEY, James D. et al. **Computer Graphics: Principles and Praticce**. [S.l.]: Addison-Wesley, 1990.
- [FRE 87] FREEMAN, Maynard L.; KAPLAN, Ervin. Time-Based Multiple Organ Functional Images. **Seminars in Nuclear Medicine**, [S.l.], v.7, n.1, Jan. 1987.
- [HER 92] HERMAN, Garbor T.; ZHENG, Jingsheng; BUCHOLTZ. Shape-Based Interpolation. **IEEE Computer Graphics & Aplications**, New York, 1992. p. 69-79.
- [LEW 83] LEWITT, Robert M. Reconstruction Algorithms: Transform Methods. **Proceedings of the IEEE**, New York, v.71, n.3, Mar. 1983.
- [POG 77] TER-POGOSSIAN, Michel M. Basic Principles of Computed Axial Tomography. **Seminars in Nuclear Medicine**, New York, v.7, n.2, Apr. 1977.
- [RAY 90] RAYA, S.; UDUPA, J. Shape-Based Interpolation of Multidimensional Objects. **IEEE Transition Medical Imaging**, New York, v.9, n.1, p.33-42, Mar. 1990.
- [ROB 83] ROBB, Richard A. et al. High-Speed Three-Dimensional X-Ray Computed Tomography: The Dynamic Spatial Reconstructor. **Proceedings of the IEEE**, New York, v.71, n.3, Mar. 1983.
- [ROG 88] ROGERS, David F. **Procedural Elements for Computer Graphics**. [S.l.]: McGrawHill, 1988.

- [UDU 83] UDUPA, Jayaram K. Display of 3D Information in Discrete 3D Scenes Produced by Computerized Tomography. **Proceedings of the IEEE**, New York, v.71, n.3, Mar. 1983.
- [MCA 88] MC CANN, Hugh A. et al. Multidimensional Ultrasonic Imaging for Cardiology. **Proceedings of the IEEE**, New York, v.76, n.9, p. 1063-1072, Sept. 1988.