

RECONSTRUÇÃO DE SUPERFÍCIES BASEADA EM CONCEITOS DE MULTI-RESOLUÇÃO E O MODELO DE T-SUPERFÍCIES

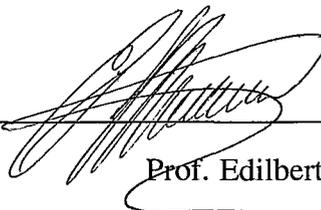
Rodrigo Luis de Souza da Silva

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

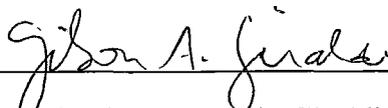
Aprovada por:



Prof. Antonio Alberto Fernandes de Oliveira, D.Sc.



Prof. Edilberto Strauss, Ph.D.



Prof. Gilson Antônio Giraldo, D.Sc.



Prof. Fabiana Rodrigues Leta, D.Sc.

RIO DE JANEIRO, RJ - BRASIL

MAIO DE 2002

SILVA, RODRIGO LUIS DE SOUZA

Reconstrução de Superfícies Baseada em
Conceitos de Multi-Resolução e o Modelo de T-
Superfícies [Rio de Janeiro] 2002

IX, 63 p., 29,7 cm, (COPPE/UFRJ, M.Sc.,
Engenharia de Sistemas e Computação, 2002)

Tese – Universidade Federal do Rio de Janeiro,
COPPE

1 – Modelo de T-Snake

2 – Modelo de T-Superfícies

3 – Modelo Proposto

I. COPPE/UFRJ II. Título (série)

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

RECONSTRUÇÃO DE SUPERFÍCIES BASEADA EM CONCEITOS DE MULTI-RESOLUÇÃO E O MODELO DE T-SUPERFÍCIES

Rodrigo Luis de Souza da Silva

Maio/2002

Orientadores: Antonio Alberto Fernandes de Oliveira

Edilberto Strauss

Programa: Engenharia de Sistemas e Computação

Esta tese apresenta uma nova abordagem que integra o modelo das T-Superfícies e conceitos de multi-resolução em uma metodologia unificada para segmentação e reconstrução de superfícies. Para imagens com ruído, propomos a utilização de difusão anisotrópica para melhorar os resultados. Apesar da eficiência da filtragem, as vezes é necessário intervenção manual para completar a reconstrução. Desta forma, baseado na capacidade topológica das T-Superfícies, o usuário pode modificar manualmente a topologia da superfície. O método foi testado em imagens médicas e sintéticas.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

SURFACE RECONSTRUCTION APPROACH BASED ON MULTI-RESOLUTION
CONCEPTS AND THE T-SURFACES FRAMEWORK

Rodrigo Luis de Souza da Silva

May/2002

Advisors: Antonio Alberto Fernandes de Oliveira

Edilberto Strauss

Department: Computing and Systems Engineering

In this thesis we present a new approach which integrates the T-Surfaces framework and a multi-resolution concept in a unified methodology for segmentation and surface reconstruction. For noise images, we can improve the result by anisotropic diffusion. Despite this improvement, some manual intervention may be required to complete the reconstruction. Thus, we take advantage of the topological capabilities of T-Surfaces to enable the user to modify the topology of a surface. We test the method for synthetic and medical volume images.

Sumário

1	Introdução	1
2	Modelo de <i>T-Snake</i>	6
2.1	<i>Snakes</i>	6
2.1.1	Modelo Original de <i>Snake</i>	7
2.1.2	Limitações do Modelo Original	8
2.2	<i>T-Snakes</i>	8
2.2.1	Decomposição do Domínio	8
2.2.2	Modelo Discreto	12
2.2.3	Função Característica	16
2.2.4	Limitações das <i>T-Snakes</i>	21
3	Modelo de T-Superfícies	22
3.1	Descrição da T-Superfície	22
3.2	Inicialização do Modelo	22
3.2.1	Decomposição Simplicial do Espaço	22
3.2.2	Superfície Inicial	23
3.2.3	Função Característica Inicial	23
3.2.4	Armazenamento dos tetraedros transversos	25
3.2.5	Projeção da Superfície Inicial	26
3.2.6	Reconstrução do Modelo	26
3.3	Etapa Dinâmica	26
3.3.1	Evolução da T-Superfície	27
3.3.2	Atualização da Função Característica	29
3.3.3	CrITÉRIOS de Parada	29
4	Modelo Proposto	31
4.1	Inicialização utilizando <i>Threshold</i>	31

4.1.1	Multi-Resolução	31
4.2	Atualização da Função Característica	32
4.3	Filtragem por Difusão Anisotrópica	34
4.4	Interação com o modelo de T-Superfícies	36
4.5	<i>Offsets</i>	36
4.6	Overview do método	37
5	Análise dos Resultados	41
5.1	Precisão do Método	42
5.2	Comparação entre os métodos de Inicialização	43
5.3	Aplicações da Filtragem por Difusão Anisotrópica	44
5.4	Multi-resolução	45
5.5	Geração de <i>offset</i>	45
6	Conclusões e Trabalhos Futuros	50
6.1	Comparação com sistema dinâmico de partículas	50
6.2	Comparação com <i>Level Sets</i>	51
6.3	Trabalhos futuros	51
A	T-Surface Builder	53
A.1	Visualizador de Superfícies	53
A.2	Visualizador de Imagens	54
A.3	Construtor de Superfícies	55
A.3.1	Parâmetros da T-Superfície	56
A.3.2	Métodos de Inicialização	57
A.4	Informações Técnicas	60
	Referências Bibliográficas	61

Lista de Figuras

2.1	Triangulação do tipo <i>Freudenthal</i>	9
2.2	Exemplos de <i>k-simplex</i>	10
2.3	Exemplo de triângulo e tetraedros transversos.	11
2.4	Reparametrização do modelo de <i>T-Snake</i> . Na figura, <i>T-Snake</i> expande e muda de posição, novos nós são computados e novos <i>snaxels</i> criados, respectivamente.	15
2.5	Snake projetada sobre uma triangulação do tipo Coxeter-Freudenthal.	16
2.6	Merge entre duas curvas. Nós marcados são aqueles onde χ tem valor 1.	17
2.7	Segunda fase da Reparametrização da T-Snake	18
2.8	Subespaços usados para classificação de nós.	19
2.9	Possíveis casos para classificação.	20
3.1	Célula cúbica decomposta em tetraedros.	23
3.2	Evolução da T-Superfície através de uma semente esférica. (a) Primeira evolução. (b) Iteração 19. (c) Iteração 32. (d) Iteração 49.	24
3.3	Exemplos de tetraedro de borda. Os sinais indicam se o vértice é interno (positivo) ou externo (negativo) ao objeto.	27
3.4	Diagrama da Força Elástica sobre um ponto P pertencente às faces f_1, f_2, f_3, f_4 e f_5	28
3.5	Diagrama da Força Normal sobre o ponto P	28
3.6	(a) Evolução do elemento triangular, (b) triangulação das faces formando poliedro de 8 faces e (c) Formação dos 8 subespaços.	30
4.1	Evolução da T-Superfície através de inicialização via <i>threshold</i> . (a) Primeira evolução. (b) Iteração 3. (c) Iteração 6. (d) Iteração 9 (final)	33
4.2	Visão do processo de atualização em um dos eixos coordenados. (a) Inicialmente, elemento triangular encontra-se na sua posição original. (b) Ocorre deformação do modelo e elemento triangular altera posição. (c) Vértice do <i>grid</i> por onde elemento triangular passou é queimado.	34

4.3	Atualização da Função Característica para o caso de 2 células vizinhas contendo elementos triangulares. (a) Caso de expansão. (b) Expansão após evolução. (c) Duas componentes se aproximam (d) Operação de Merge.	35
4.4	Em (a)-(b) temos o resultado da inicialização da via <i>threshold</i> com difusão Gaussiana e Anisotrópica, respectivamente. (c)-(d) Corte de (a),(b) para <i>slice</i> 40, respectivamente, sobre a imagem original. (d)-(e) Solução Final para (c),(d).	38
4.5	(a) T-Superfície antes e (b) após corte manual.	39
4.6	Diagrama das etapas da T-Superfície.	40
5.1	Corte transversal das esferas imersas no ruído (a) Inicialização da T-Surface. (b) Resultado Final.	43
5.2	Comparativo entre os métodos de inicialização (a) Primeira evolução. (b) Iteração 32. (c) Iteração 49 (final). (d) Inicialização por <i>threshold</i> . (e) Iteração 6. (f) Iteração 9 (final).	44
5.3	Aplicação da filtragem por difusão anisotrópica (a) Segmentação da artéria com volume de dados original (b) Segmentação após filtragem.	47
5.4	Aplicação da multi-resolução (a) Artéria inicialmente segmentada na resolução 3 x 3 x 3 (b) Segmentação final na resolução 1 x 1 x 1.	48
5.5	Geração de <i>Offsets</i> (a) Artéria Inicial (b) Artéria com <i>offset</i>	49
A.1	Visualizador de superfícies apresentando duas artérias.	54
A.2	Visualizador de superfícies. (a) Objeto visto de frente (Ventrículo), (b) Objeto rotacionado, (c) Objeto afastado e (d) <i>Wireframe</i>	55
A.3	Visualizador de superfícies sendo utilizado para mostrar o interior de um objeto.	56
A.4	Visualizador de imagens. (a) e (b) Artérias (<i>Visible Man</i>) (c) e (d) Interior da cabeça (MRI).	57
A.5	Janela de parâmetro para inicializar processo segmentação.	58
A.6	Informações sendo mostradas durante execução da T-Superfície.	59
A.7	Inicialização através de objeto esférico.	59
A.8	Inicialização através de <i>threshold</i>	59
A.9	Inicialização através de arquivo.	60

Lista de Tabelas

5.1	Análise da precisão do método com imagens sintéticas.	42
5.2	Análise da precisão do método com imagens sintéticas com ruído.	43
5.3	Análise comparativa da eficiência dos métodos de inicialização.	44

Capítulo 1

Introdução

Os Modelos de Contorno Ativo (*Snakes*) vêm se estabelecendo como um importante recurso para a solução de problemas em processamento de imagens digitais, particularmente para segmentação de imagens e *motion tracking* (Black & Yuille, 1993). A segmentação é um passo fundamental para a análise e identificação das características relevantes em uma imagem (Jain, 1989). Segmentar uma imagem significa separá-la em regiões com propriedades comuns (cor, textura, etc) as quais correspondem a objetos ou partes de objetos na imagem, ou mesmo ao “*background*” (Jones & Metaxas, 1998).

De maneira geral, os métodos para segmentação fazem uso de limiares (*thresholds*) e hipóteses iniciais para a imagem tais como tipos de texturas, distribuição do histograma, etc (Zhu & Yuille, 1996). De uma forma geral, estes métodos podem ser classificados em quatro grupos: filtragem local (Canny, 1983); contornos ativos (Black & Yuille, 1993); crescimento e agrupamento de regiões (Adams & Bischof, 1994); e otimização de funções de energia definidas globalmente sobre a imagem (Mumford & Shah, 1989).

Os métodos baseados em filtragem local são particularmente úteis para o realce de características de interesse (Ex.: bordas (Canny, 1983)) e para a detecção de regiões com uma mesma textura, como é o caso típico do filtro Gaussiano (Jain, 1989).

Nos métodos que usam crescimento e agrupamento de regiões procura-se dividir o domínio da imagem em sub-regiões, cada uma destas satisfazendo algum critério de homogeneidade. Os objetos segmentados por estas técnicas apresentam em geral bordas irregulares podendo ocorrer pequenos buracos no interior dos mesmos, os quais deverão ser preenchidos em uma etapa de pós-processamento.

As técnicas envolvendo critérios globais segmentam a imagem utilizando funções de energia definidas globalmente, tendo a vantagem de tratar a imagem como um todo, mas apresentando dificuldades extras na obtenção do mínimo global que fornece a solução desejada (Zhu

& Yuille, 1996) (Mumford & Shah, 1989).

Os Modelos de Contornos Ativos por sua vez, na sua formulação original proposta por Kass (Kass *et al.*, 1988), fazem uso de informações locais sobre um contorno, podendo incorporar também informações (a priori) sobre a imagem como um todo, ou sobre os segmentos procurados. Esses métodos são usados para a extração das bordas dos objetos da cena, sendo em geral aplicados conjuntamente com técnicas de filtragem usadas na detecção de pontos de bordas (Canny, 1983) (Cohen, 1991). Uma dificuldade com estes métodos é a necessidade de uma boa estimativa inicial do contorno procurado para garantir o bom funcionamento do método.

Snakes são um tipo particular de modelos deformáveis. Em geral, o termo modelo deformável em processamento de imagens refere-se a uma curva/superfície inicial (*template*) a qual se deforma sob a ação de forças internas (intrínsecas à geometria da curva) e externas (derivadas dos dados) atingindo o equilíbrio quando sobre a fronteira do objeto.

As limitações topológicas do modelo original são tratadas no contexto dos modelos paramétricos basicamente por procedimentos compostos por duas etapas: (1) determinação da necessidade de mudanças topológicas; (2) um método para efetuá-las. Por mudança topológica, entendemos o fato de uma *snake* (fechada) se subdividir em duas (ou mais), ou várias *snakes* se juntarem em uma única.

Durikovic (Durikovic *et al.*, 1995) propôs um método que combina características dinâmicas associadas a mudanças topológicas. No método desenvolvido a alteração topológica é implementada primeiramente construindo-se um histograma da intensidade da força de imagem ao longo da *snake* com o objetivo de identificar uma região apropriada para “cortar” a *snake* (região com contraste mais fraco). A seguir, o método procura dois pontos nesta região para serem os extremos do segmento que cortará a curva em duas partes. Este método tem a desvantagem de não tratar o problema do *merge* (junção) entre contornos e sua extensão para superfícies não é óbvia.

Uma metodologia mais geral para tratar o problema de mudanças topológicas é o modelo das *T-Snakes* (McInerney & Terzopoulos, 1995). A idéia básica deste método é imergir (projetar) o modelo de *snakes* em uma decomposição simplicial (triangulação) do domínio de interesse, permitindo a utilização de resultados em métodos de continuação numérica para a execução eficiente e robusta de mudanças topológicas (Allgower & Georg, 1990). Neste método, os *snaxels* (elementos discretos da *T-Snake*) são determinados a partir das interseções da *snake* com a triangulação. Este processo corresponde a uma reparametrização da curva (*snake*) baseada na subdivisão do espaço no qual a mesma está imersa. Tal procedimento (extrínseco) tem a desvantagem de perturbar a *snake* (particularmente nas proximidades das bordas desejadas) e

tornar o método muito sensível à triangulação usada.

Estas limitações fazem com que a evolução da *T-Snake* seja definida através de informações estritamente locais, não envolvendo nenhum processo de minimização explícito (McInerney & Terzopoulos, 1997). Apesar das dificuldades citadas, uma vez projetada a *snake*, é possível distinguir eficientemente os nós (vértices dos triângulos) internos e externos à mesma, criando-se uma *função característica* que tem valor 1 para nós interiores e 0 para os exteriores (condição de entropia). Esta função é o elemento central do método uma vez que dispo de um algoritmo suficientemente rápido para atualizá-la (alteração da entropia de um *snaxel*), as mudanças topológicas da(s) *snake(s)* se reduzem a encontrar os triângulos de borda correspondentes (triângulos onde a função característica muda seu sinal). Este procedimento simples para a implementação de mudanças topológicas é o grande benefício obtido a partir da reparametrização descrita acima (McInerney & Terzopoulos, 1997).

Dentre as vantagens deste método estão a sua generalidade, permitindo tanto a junção quanto a separação (*split*) da(s) *snake(s)*, e a possibilidade de estendê-lo naturalmente para superfícies uma vez que seus elementos básicos (triangulação, reparametrização, função característica) são eficientemente extensíveis ao caso 3D (McInerney & Terzopoulos, 1997). As *T-Snakes* têm como um dos métodos inspiradores as formulações implícitas do modelo de *snakes*, particularmente o modelo dos Conjuntos de Níveis (*Level-Sets*) proposto por Malladi et al (Malladi et al., 1995).

No método das *T-Snakes* usa-se uma força normal para guiar a *snake* em direção às bordas desejadas. O sentido desta força é definido em função de um limiar para a intensidade da imagem. Pode-se gerar problemas na obtenção do contorno devido a não homogeneidade da intensidade que define a borda do objeto (mínimos locais). Uma forma de evitar esses mínimos locais é variar este limiar dentro de um intervalo estabelecido por estatísticas da imagem.

Foi proposto por McInerney (McInerney, 1997) o modelo das T-Superfícies, que é a extensão 3D das *T-Snakes*. Este modelo também depende de um certo *threshold* para definir a força normal que será usada para guiar a superfície em direção ao seu alvo.

Tomando como ponto de partida os elementos básicos das *T-Snakes* (*threshold* e divisão simplicial do espaço), Girdali (Girdali et al., 2000) propôs uma abordagem de segmentação de imagens 2D baseada em métodos de multiresolução e o modelo das *T-Snakes*.

O presente trabalho descreve a extensão dessa abordagem para o espaço tridimensional através das T-Superfícies. O enfoque deste trabalho foi no estudo de técnicas de multi-resolução aplicado ao modelo de T-Superfícies com a finalidade de reconstruir superfícies de forma mais eficiente. As técnicas desenvolvidas podem ser aplicadas na segmentação de objetos em um

volume de imagens. Os testes foram feitos utilizando-se imagens médicas e sintéticas.

Assim como no modelo 2D proposto por Giraldi, foi assumida uma restrição de escala para os objetos alvo. Num primeiro estágio, essa restrição foi utilizada para definir uma resolução de imagem mais baixa que garanta a não separação dos objetos. Para o *grid* (divisão celular) correspondente, foi feita uma decomposição simplicial (triangulação 3D) do domínio de interesse. O campo de imagem de baixa resolução é então “marcado” (mudança da entropia) através do *threshold* para obtermos uma função binária, a qual chamamos de *Função Característica do objeto*. Então, um simples método de continuação é usado para extrair um conjunto de superfícies poligonais fechadas que contenham as estruturas anatômicas que definem o contorno do objeto 3D (*Shell*).

A resolução da triangulação depende da aplicação. Entretanto, um importante aspecto do método implementado é sua natureza *multi-grid* e *multi-resolução*: após segmentar a imagem numa resolução de *grid* menor, pode-se localizar regiões onde o *grid* tem que ser refinado e o método é recursivamente aplicado sobre essas regiões.

As superfícies poligonais obtidas são em geral uma aproximação pouco precisa das superfícies de interesse. Nesse ponto, aprimoram-se essas aproximações utilizando o modelo das T-Superfícies, inicializando o mesmo com a superfície inicial obtida nos primeiros estágios. Para imagens com ruído, pode-se aumentar a eficiência do método através de um pré-processamento da imagem original utilizando difusão anisotrópica (Perona & Malik, 1990).

Se a segmentação continuar incompleta numa resolução mais fina, introduzem-se métodos de interatividade baseados no conceito das T-Superfícies para “cortá-la” e completar a segmentação.

Nos capítulos seguintes, serão apresentadas as metodologias que precedem o trabalho desenvolvido, indicando suas características e limitações. Após introduzidos os conceitos básicos para esse trabalho, serão apresentadas as inovações propostas por esta tese e suas formas de aplicação. O texto que segue está organizado da seguinte forma.

No capítulo 2 será introduzido o modelo de *Snake* e sua extensão com capacidades topológicas, a *T-Snake*. Será apresentado o sistema de forças que atua sobre o modelo e suas limitações. Posteriormente, será introduzido o modelo de *T-Snake*. Serão enfocados os três componentes básicos do modelo: (1) uma triangulação do domínio de interesse, (2) um modelo discreto da curva deformável e (3) uma função característica.

O Capítulo 3 apresenta o modelo de T-Superfície. Serão discutidas com detalhes as etapas necessárias para reconstruirmos superfícies utilizando esse modelo.

O Capítulo 4 apresenta o modelo proposto nesse trabalho. Será discutida a inicialização

baseada no estudo do *threshold*, utilizando conceitos de multi-resolução aplicada ao modelo de T-Superfícies. Será também apresentada uma nova proposta para o problema de atualização da Função Característica. Será apresentada uma forma de aplicação do modelo proposto para geração de *offsets*. Por fim, serão apresentadas propostas de interação com o modelo com a finalidade de, eventualmente, separar manualmente componentes conexas.

A análise dos resultados será apresentada no Capítulo 5. Será demonstrada a eficiência da metodologia desenvolvida em relação a proposta original do modelo de T-Superfícies. Será analisada a precisão do método além da forma como o método se comporta em imagens com ruído. Para este último caso, foi proposta a utilização de filtragem por difusão anisotrópica. Serão apresentados casos onde a multi-resolução foi necessária para obter o resultado esperado. As propostas para trabalhos futuros e conclusões sobre o presente trabalho serão apresentadas no Capítulo 6.

O Apêndice A foi utilizado para apresentar o aplicativo desenvolvido com a metodologia discutida nos capítulos anteriores. Este aplicativo foi desenvolvido em colaboração com o Laboratório Nacional de Computação Científica, dentro do contexto do Projeto “Modelagem e Simulação Computacional do Sistema Cardiovascular Humano, daquele instituto”.

Capítulo 2

Modelo de *T-Snake*

Nesse capítulo será apresentada uma introdução ao modelo tradicional de *Snakes* e sua extensão com capacidade de tratar alterações topológicas, a *T-Snake*. Inicialmente será explicado como funciona o modelo de *Snakes* e suas limitações. Posteriormente, será apresentado o modelo de *T-Snakes*, incluindo uma visão geral das etapas que compõem o modelo além de suas limitações.

2.1 *Snakes*

As *Snakes* são um tipo de modelo deformável no qual uma curva *spline* é deformada pela ação de forças internas e externas. Essa deformação provoca um deslocamento da *Snake* em direção à borda de interesse (Kass *et al.*, 1988; Black & Yuille, 1993). O objetivo do processo é fazer com que a energia associada a *Snake* atinja a posição de equilíbrio apenas quando o modelo aproximar a borda desejada.

O resultado do processo fornece, desta forma, informações a respeito da forma e da posição da borda na imagem. É neste sentido que o método de *Snakes* é uma técnica de *extração de bordas* de objetos em uma imagem.

Note que os pontos de uma borda para imagens em tons de cinza pode ser definida como:

$$\|\nabla I\|^2 > T, \quad (2.1)$$

onde T é um limiar previamente estabelecido usando-se, por exemplo, um histograma da intensidade do campo gradiente na imagem.

2.1.1 Modelo Original de *Snake*

A energia da *Snake* será composta por um termo diretamente relacionado à imagem que guarda as características de interesse da mesma, além de um termo de *regularização* ou de *suavização* (Poggio *et al.*, 1985; Snyder, 1991). Este último, além de garantir estabilidade numérica, irá guardar características geométricas supostas (a priori) para a solução.

O primeiro passo para a definição precisa da energia é definir o espaço de curvas (Cohen, 1991). Seja então K um espaço de curvas parametrizadas por um parâmetro s , $c(s) = (x(s), y(s))$, tal que:

$$K = \{c : [0, 1] \rightarrow D \subset \mathbb{R}^2; c \in C^4\}, \quad (2.2)$$

onde D é um domínio de interesse, podendo ser todo o domínio de definição da imagem ou apenas uma parte deste.

Como descrito em Kass *et al.* (Kass *et al.*, 1988), a energia associada a *Snake* é discretizada em dois domínios: (1) Energia Externa e (2) Energia Interna. O funcional de energia da *Snake* tem a forma geral dada por:

$$E(c) = E_{ext}(c) + E_{int}(c), \quad (2.3)$$

O primeiro domínio, Energia Externa (E_{ext}), refere-se às características extraídas da própria imagem e que servirá para caracterizar a ocorrência de bordas de interesse durante o processo dinâmico de movimentação da *Snake*.

$$E_{ext}(c) = \int_0^1 P(c(s)) ds, \quad (2.4)$$

sendo P o potencial externo.

Já o segundo termo, Energia Interna (E_{int}), é responsável pela continuidade e suavidade de forma sobre toda a extensão da *Snake*. É um termo de regularização, dado por:

$$E_{int}(c) = \int_0^1 (\omega_1 \|c'(s)\|^2 + \omega_2 \|c''(s)\|^2) ds, \quad (2.5)$$

onde ω_1 e ω_2 são parâmetros internos denominados parâmetros de elasticidade e rigidez, respectivamente, $c'(s) = dc/ds$ e $c''(s) = d^2c/ds^2$.

O potencial externo P vai gerar um campo de forças externo à curva que deverá “puxar” a mesma em direção às bordas.

Assim, a energia da *Snake* toma a forma:

$$E(c) = \int_0^1 (\omega_1 \|c'(s)\|^2 + \omega_2 \|c''(s)\|^2) ds + \int_0^1 P(c(s)) ds. \quad (2.6)$$

Este funcional corresponde à energia potencial de uma corda fina imersa em um campo de forças externo definido pelo potencial P .

2.1.2 Limitações do Modelo Original

Apesar de suas vantagens para extração de contornos, o modelo original de *Snakes* apresenta várias limitações (McInerney, 1997) (Gunn & Nixon, 1997) (Xu *et al.*, 1994).

Primeiramente, o funcional de energia é em geral não convexo e assim, os métodos de otimização baseados em informações locais não oferecem garantia de que a solução encontrada seja o mínimo global (Gunn & Nixon, 1997). Este problema implica que a solução é sensível à inicialização do método.

Uma outra limitação inerente ao modelo original é a sua dificuldade em tratar possíveis mudanças da topologia da curva inicial, ou seja, aplicação de operações de *splits* sobre a curva inicial (McInerney & Terzopoulos, 1995; McInerney, 1997). A menos que acrescentemos outros recursos ao modelo, a evolução da curva baseada na minimização do funcional impõe que a variedade correspondente à curva permaneça conexa durante toda a evolução, isto é, a curva não pode sofrer subdivisões (*splits*).

2.2 T-Snakes

O método das *T-Snakes* (McInerney & Terzopoulos, 1995) (McInerney, 1997) tem como idéia fundamental converter o problema das questões topológicas para o contexto da Topologia Combinatória a fim de obter uma formulação paramétrica com habilidades topológicas para as *Snakes*.

O método das *T-Snakes* é composto basicamente por três componentes: (1) uma triangulação do domínio de interesse, (2) um modelo discreto da curva deformável, (3) uma função característica.

2.2.1 Decomposição do Domínio

Dado um subconjunto fechado $D \subset \mathbb{R}^n$, existem dois métodos básicos para decomposição de domínio: não-simpliciais e simpliciais.

Os métodos não-simpliciais são baseados em decomposições celulares do espaço e não podem ser usados para representar superfícies ou contornos sem a utilização de métodos para retirar ambiguidades. Caso típico de um método que utiliza este tipo de decomposição do espaço é o *Marching Cubes* (Lorensen & Cline, 1987).

Métodos usando decomposição simplicial do domínio não apresentam ambigüidades para a geração de aproximações poligonais para curvas e superfícies (Allgower & Georg, 1990).

Na decomposição simplicial do espaço, também conhecida como triangulação, o espaço é particionado em células formadas pelos mais simples objetos geométricos de sua respectiva dimensão, isto é, triângulos em 2D e tetraedros em 3D. Sendo todas as células idênticas, a computação necessária para lidar com essa estrutura é menor, possibilitando a criação de algoritmos eficientes e de fácil manuseio. O tipo mais simples de triangulação do espaço Euclidiano \mathfrak{R}^n com essa propriedade é o *Coxeter-Freudenthal* (Figura 2.1). Essa decomposição é construída dividindo o espaço por um *grid* cúbico uniforme, sendo a triangulação obtida através da subdivisão de cada um dos cubos que compõem o *grid* em triângulos (2D).

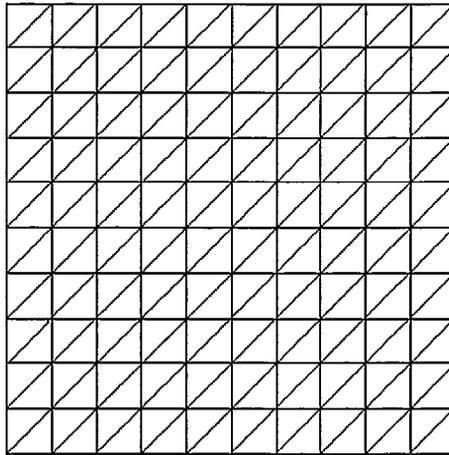


Figura 2.1: Triangulação do tipo *Freudenthal*.

Um conjunto de pontos $\{v_1, v_2, \dots, v_{k+1}\} \subset \mathfrak{R}^n$ é dito “*afim-independente*” se as diferenças $v_2 - v_1, v_3 - v_1, \dots, v_{k+1} - v_1$ formam um conjunto de vetores linearmente independente no \mathfrak{R}^n .

O fecho convexo $\left\{v = \sum_{i=1}^{k+1} \alpha_i v_i \mid \alpha_i \geq 0, i = 1, \dots, k + 1, \sum_{i=1}^{k+1} \alpha_i = 1\right\}$, com vértices dados por $k + 1$ pontos *afim-independentes* $\{v_1, v_2, \dots, v_{k+1}\}$ do \mathfrak{R}^n , é chamado um *k-simplex* e será denotado por $[v_1, v_2, \dots, v_{k+1}]$.

Seja $\sigma = [v_1, v_2, \dots, v_{k+1}]$ um *k-simplex* e seja $1 \leq l \leq k + 1$. Se $\{w_1, w_2, \dots, w_{l+1}\}$ é um subconjunto de vértices de σ , chama-se *l-simplex* $\tau = \{w_1, w_2, \dots, w_{l+1}\}$ uma *face l-dimensional* de σ , ou simplesmente uma *l-face*. As *l-faces* de particular interesse são:

- (1) 0-faces as quais correspondem a vértice de σ ;
- (2) 1-faces as quais são também chamadas *arestas* de σ ;
- (3) $(k - 1)$ -faces as quais são chamadas *faces* de σ .

Seja Γ uma família não-vazia de $(n+1)$ -simplices em \mathbb{R}^{n+1} . Chama-se Γ uma triangulação do \mathbb{R}^{n+1} se:

(1) $\sigma \in \Gamma \cup \sigma = \mathbb{R}^{n+1}$;

(2) a intersecção $\sigma_1 \cap \sigma_2$ de dois *simplices* quaisquer $\sigma_1, \sigma_2 \in \Gamma$ é vazia ou uma face comum aos *simplices*;

(3) a família Γ é localmente finita, isto é, qualquer subconjunto compacto de \mathbb{R}^{n+1} corta um número finito de simplices $\sigma \in \Gamma$.

Seja Γ uma triangulação do \mathbb{R}^{n+1} , e seja $0 \leq k \leq n + 1$. Denota-se por

$$\Gamma^k = \{ \tau \mid \tau \text{ é uma } k\text{-face de algum simplex } \sigma \in \Gamma \},$$

a família de todas as k -faces em Γ .

Para o \mathbb{R}^2 um 2-simplex é um triângulo e para o \mathbb{R}^3 um 3-simplex é um tetraedro (Figura 2.2). Seguindo a nomenclatura padrão, chama-se *nó* a um vértice de um *simplex*, e à coleção de nós e arestas de Γ chama-se *malha* (identificada com Γ^1). A triangulação Γ é identificada com Γ^{n+1} ($\Gamma = \Gamma^{n+1}$).

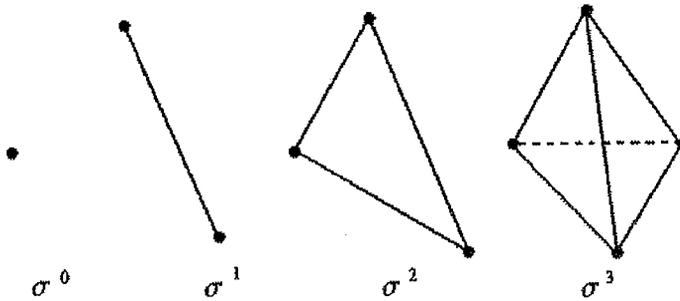


Figura 2.2: Exemplos de k -simplex ($k = 0, 1, 2$ e 3 , respectivamente).

Dois simplices $\sigma_1, \sigma_2 \in \Gamma$ são ditos adjacentes se eles têm uma face em comum.

Dado um *simplex*, o procedimento usado para obter os adjacentes a este é denominado *pivoteamento* e depende do tipo de triangulação usada. O lema a seguir prepara sua introdução.

Seja Γ uma triangulação do \mathbb{R}^{n+1} . Considere um *simplex* $\sigma \in \Gamma$ e seja τ uma face de σ . Então, existe um único *simplex* $\tilde{\sigma} \in \Gamma$ tal que:

- (1) $\tilde{\sigma} \neq \sigma$;

(2) τ uma face de σ ;

Denotemos por $delv_i$ a operação de eliminação de um vértice v_i .

Seja $\sigma = [v_1, v_2, \dots, v_{n+2}]$ um $(n+1)$ -simplex de uma triangulação de \mathbb{R}^{n+1} , e seja $\tau = [v_1, \dots, delv_i \dots, v_{n+2}]$ a face de σ oposta ao vértice v_i . Pelo lema anterior, deve existir um único nó \tilde{v}_i diferente de v_i e tal que $\tilde{\sigma} = [v_1, \dots, \tilde{v}_i \dots, v_{n+2}] \in \Gamma$. A passagem de σ para $\tilde{\sigma}$ é chamada *pivoteamento*. Dizemos que o vértice v_i de σ é *pivoteado* no \tilde{v}_i , e que o *simplex* σ é *pivoteado* no *simplex* $\tilde{\sigma}$ através da face τ .

Variedade Linear por Partes

A decomposição simplicial do espaço nos permite criar aproximações robustas das *Snakes* através da avaliação do sinal (ou valor binário) dos vértices que compõem a estrutura (Figura 2.3). Em 2D, um contorno inicial separa a imagem em dois conjuntos abertos de dimensão 2 (os pontos internos e os pontos externos) e um conjunto fechado de dimensão 1 (pontos pertencentes a curva). Um *simplex* σ^2 pode ser classificado como totalmente interior, ou exterior, se todos os seus vértices possuírem o mesmo sinal. Se os sinais forem diferentes, a curva deve interceptar o *simplex* em algum ponto. Em um k -*simplex*, os vértices negativos (internos) podem sempre ser separados dos positivos (externos) através de um plano, fazendo com que sempre exista uma solução para o problema de ambiguidade na classificação de um *simplex*.

Defini-se como *Completamente Rotulado* uma aresta de um triângulo σ em relação a χ (Equação 2.12) se sua função mudar de valor nas extremidades da aresta. Um triângulo σ no \mathbb{R}^3 é chamado *transverso* em relação a χ se ele contiver uma aresta completamente rotulada (Figura 2.3).

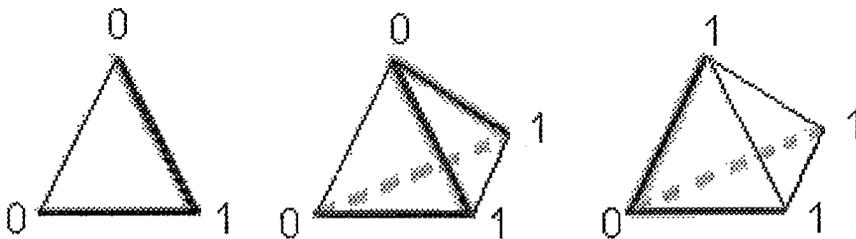


Figura 2.3: Exemplo de triângulo e tetraedros transverso. As arestas completamente rotuladas estão em negrito.

Um conceito fundamental nessa teoria é a Variedade Linear por Partes (*Piecewise Linear Manifold*) \widehat{M} . Dada uma variedade bidimensional M implicitamente definida, é suficiente dizer que uma variedade linear por partes é uma representação poligonal (célula) desta variedade com

as seguintes propriedades: (1) A intersecção $\sigma_1 \cap \sigma_2$ de duas células $\sigma_1, \sigma_2 \in \widehat{M}$ é vazia ou uma aresta comum a duas células; (2) Uma aresta $\tau \in \widehat{M}^N$ é comum a pelo duas células de \widehat{M} ; (3) \widehat{M} é localmente finito, isto é, qualquer subconjunto compacto de M contém apenas um número finito de células de \widehat{M} .

Cada componente conexa numa triangulação Γ da variedade linear por partes representando o conjunto $\chi^{-1}(0)$ pode ser gerada seguindo o algoritmo abaixo (Allgower & Georg, 1990) :

Algoritmo Door-In-Door-Out

Inicialização

$\sigma_0 \in \Gamma$ triângulo de borda (simplex inicial);

τ_0 uma n-face completamente rotulada de σ_0 ;

Repetir para $n = 1, 2, \dots$

Encontrar $\sigma_n \in \Gamma, \sigma_n \neq \sigma_{n-1}$ tal que $\tau_{n-1} = \sigma_n \cap \sigma_{n-1}$;

Encontrar a n-face completamente rotulada τ_n de σ_n tal que $\tau_n \neq \tau_{n-1}$

Parar quando retornar ao triângulo inicial $\sigma_0 \in \Gamma$

A variedade linear por partes obtida geralmente não é suave e não possui precisão em relação à borda.

2.2.2 Modelo Discreto

A *T-Snake* é definida por um conjunto de N pontos (*snaxels*) cujas posições $\{v_i = (x_i, y_i), i = 0, \dots, N - 1\}$, são conectadas em seqüência, formando uma curva fechada.

Cada par de pontos consecutivos v_i, v_{i+1} é chamado de “elemento do modelo”.

Os pontos da *Snake* são supostamente ligados por molas, as quais são definidas por um parâmetro de elasticidade a_i e um comprimento natural l_i , isto é, a *Snake* resiste à expansão ou compressão somente quando a distância entre *snaxels* $\|r_i(t)\| = \|v_{i+1} - v_i\|$ é maior ou menor que l_i , respectivamente.

Assim, dada a deformação $e_i = \|r_i(t)\| - l_i$, defini-se a força de tensão correspondente pela expressão:

$$\alpha_i = a_i e_i r_i(t) - a_{i-1} e_{i-1} r_{i-1}(t). \quad (2.7)$$

Uma vez que o conjunto de *snaxels* e molas não permanece constante, o comprimento de repouso das molas no instante t é definido em termos dos comprimentos das molas no instante

anterior ($t - \Delta t$).

Em adição à força (2.7), é conveniente definir uma força de rigidez para minimizar as curvaturas locais da *Snake* garantindo suavidade para a solução. Esta força é definida da seguinte forma:

$$\beta_i = b_i \left(v_i - \frac{1}{2} (v_{i-1} + v_{i+1}) \right), \quad (2.8)$$

a qual atua no sentido de minimizar a distância entre um ponto v_i e o centróide do segmento definido pelos seus vizinhos.

O modelo tem também uma força normal:

$$F_i = sig(v_i) kn_i, \quad (2.9)$$

onde $sig(v_i) = -1$ se $I(v_i) < T$ e $sig(v_i) = 1$ caso contrário (T é um limiar para a intensidade de imagem previamente determinado).

Na Equação 2.9, n_i é a normal ao vértice v_i e k é o fator que define a intensidade da força. Esta é uma força tipo *balloon*, sendo usada para puxar a *Snake* em direção às bordas dos objetos procurando evitar que a *Snake* pare em regiões onde o campo externo é nulo.

Outra possibilidade para definir o sinal da força normal é usar estatísticas da imagem para definir a função $sig(v_i)$:

$$sig(v_i) = \begin{cases} 1 & \text{se } |I(v_i) - \mu| \leq \sigma, \\ 0, & \text{caso contrário,} \end{cases}$$

onde μ e σ são a média e variância da imagem, respectivamente.

Note que as forças dadas (2.7)-(2.9) são consideradas forças internas da *Snake*. As forças externas são definidas em função das características de interesse na imagem, no caso, bordas dos objetos da cena. Uma possibilidade é definir o campo externo através do gradiente do potencial usual:

$$P = - \|\nabla I\|^2. \quad (2.10)$$

A equação de evolução para a *T-Snake* é finalmente dada por:

$$v_i^{(t+\Delta t)} = v_i^t + \frac{\Delta t}{\gamma} \left(\alpha_i^t + \beta_i^t + F_i^t + f_i^t \right), \quad (2.11)$$

Durante a evolução da *T-Snake* alguns nós da malha tornam-se interiores à mesma (supondo expansão). O valor de entropia associado a estes nós é “*queimado*”, ou seja, seu valor é alterado de 0 para 1.

Para evitar os problemas conhecidos para evolução de curvas na direção normal (desenvolvimento de singularidades) o método das *T-Snakes* adota uma condição de *entropia*: *uma vez que um nó é queimado, ele permanecerá queimado*. Esta condição tem também a finalidade de permitir a definição de um critério de parada eficiente.

Neste sentido, primeiramente define-se uma *temperatura* para cada *snaxel*, definida pelo número de deformações que o triângulo no qual um elemento do modelo correspondente se encontra permaneceu como um triângulo de borda. Uma *T-Snake* é considerada em equilíbrio quando a temperatura de todos os seus *snaxels* ultrapassar um limiar denominado *ponto de congelamento (freezing point)*. Este limiar é estabelecido empiricamente.

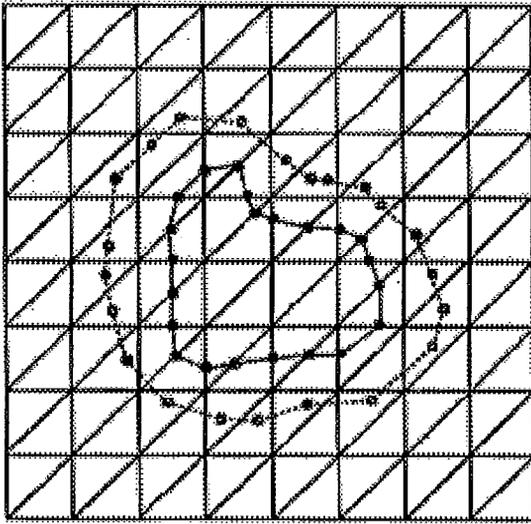
A condição de entropia acima torna esta definição mais eficiente como critério de parada pois, do contrário, uma *Snake* poderia oscilar expandindo e contraindo, e assim o número de interações necessárias para atingir o equilíbrio poderia ser muito alto.

Uma vez que a *T-Snake* atingiu o equilíbrio, a malha correspondente à divisão simplicial pode ser descartada e a *Snake* evolui como uma *Snake* discreta segundo a Equação 2.11.

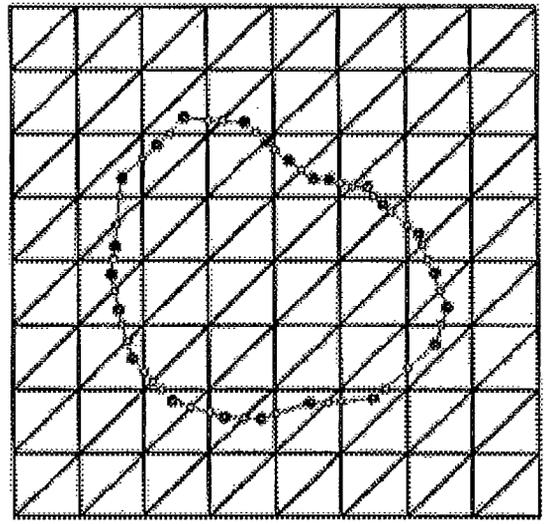
Reparametrização do modelo

Durante cada evolução da curva (Equação 2.11), a *T-Snake* movimenta-se de sua posição original para uma nova posição. No início de cada evolução, os *snaxels* estão localizados nas arestas do *grid* triangular previamente definido para a evolução da *T-Snake*. Ao final de cada passo, os *snaxels* estarão localizados, em geral, “fora” das arestas dos triângulos do *grid*. É necessário restabelecer a correspondência entre o modelo e o *grid*, computando-se os novos *snaxels* da *T-Snake* deformada através de um algoritmo de reparametrização.

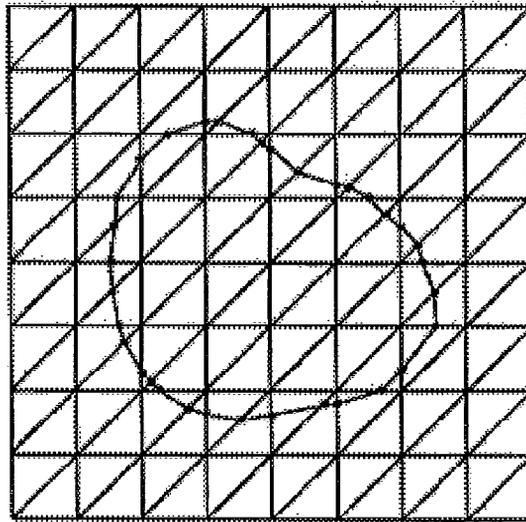
Inicialmente, é feita uma busca local e testes de intersecção para cada elemento do modelo. Isto é, para cada segmento que conecta dois nós, será computado uma *bounding box* composta por esse segmento e sua nova posição. Para cada uma das arestas contidas nesta *bounding box*, é feito um teste de intersecção em relação ao elemento do modelo. Se um ponto de intersecção é encontrado, ele é guardado e poderá tornar-se um *snaxel* no modelo atualizado (Figura 2.4).



(a)



(b)



(c)

Figura 2.4: Reparametrização do modelo de *T-Snake*. (a) *T-Snake* expande e muda de posição durante deformação, (b) novos nós são computados, (c) novos *snaxels* criados

2.2.3 Função Característica

A *função característica* (McInerney & Terzopoulos, 1995; McInerney, 1997) é uma função linear por partes definida por:

$$\chi : D \subset \mathbb{R}^2 \rightarrow \{0, 1\} \quad (2.12)$$

tal que: $\chi(p) = 1$ se $p \in O$ e $\chi(p) = 0$, caso contrário, onde p é um nó da malha e D o domínio de interesse.

Para construir esta função na inicialização, o método usado consiste primeiramente em projetar a *Snake* sobre a triangulação usada como mostra a Figura 2.5.

Os triângulos nos quais a *Snake* é projetada são denominados triângulos de borda. Aplicando o Algoritmo Door-In-Door-Out (seção 2.2.1), pode-se obter cada uma das componentes conexas da *Snake* resolvendo assim as mudanças topológicas (*splits* e *merges*).

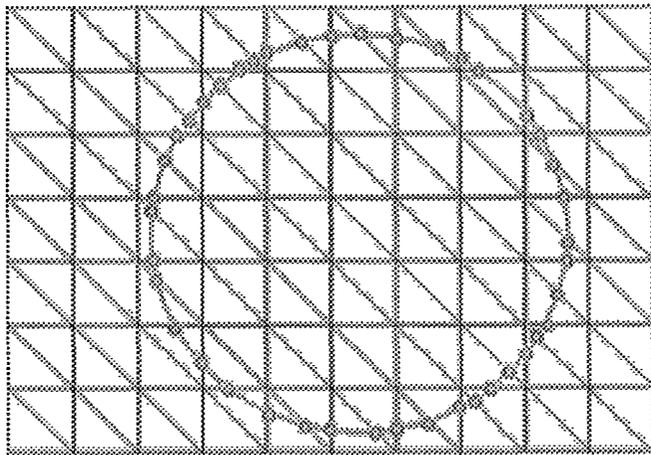


Figura 2.5: Snake projetada sobre uma triangulação do tipo Coxeter-Freudenthal.

Ao projetar-se a curva sobre a triangulação, deve-se calcular os pontos de intersecção da mesma em relação a triangulação definida. Os pontos assim obtidos serão usados posteriormente para definir os *snaxels* da *Snake*. Feito esta projeção, a determinação de χ pode ser feita por um algoritmo do tipo *scanline* (Rogers, 1985). A Figura 2.6 é um exemplo de como a função χ e algoritmo Door-In-Door-Out fornecem naturalmente o *merge* entre duas curvas.

Nesta figura, os nós marcados são aqueles onde a função característica tem valor 1, tendo os demais nós valor 0. Aplicando o algoritmo *Door-In-Door-Out*, obtêm-se ao final a sequência dos triângulos de borda. O *merge* entre as curvas pode então ser representado por uma poligonal fechada contida na região dada pela união destes triângulos.

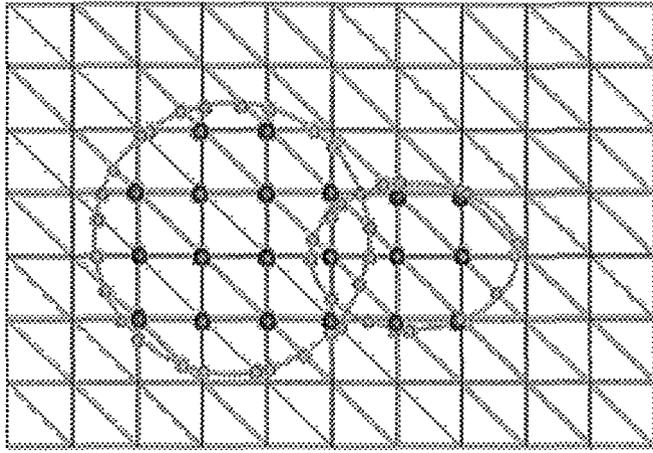


Figura 2.6: Merge entre duas curvas. Nós marcados são aqueles onde χ tem valor 1.

Para garantir a consistência do método é necessário apenas garantir que a seqüência aresta-triângulo-aresta obtida pelo algoritmo seja cíclica e finita.

Além da triangulação do espaço (no caso, o \mathbb{R}^2), da projeção da *Snake* e da função característica, um outro componente do método das *T-Snakes* é um modelo discreto de contorno ativo que será definido a seguir.

Atualização da Função Característica

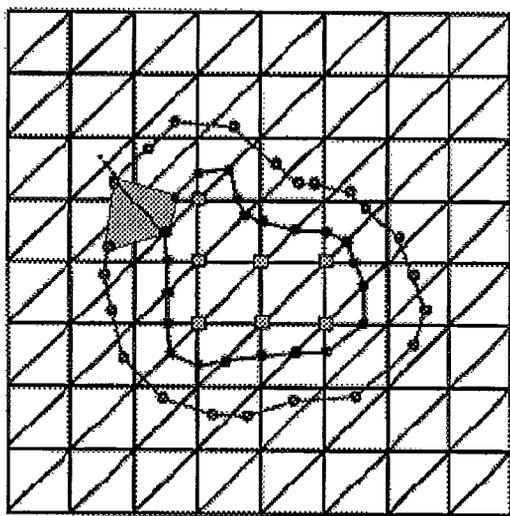
Ao final de cada passo de evolução da *Snake*, devemos atualizar a função característica definida na Equação 2.12. Isto é equivalente a determinar o conjunto de nós da malha que foram queimados durante a evolução da *Snake*.

Cada elemento do modelo (par de *snaxels* consecutivos) pode ter passado sobre nenhum, apenas um, ou vários nós durante sua evolução. Inicialmente, para cada elemento do modelo, forma-se um polígono usando a atual posição e a posição anterior. Este polígono permite facilmente determinar os nós que podem ter sido queimados (Figura 2.7).

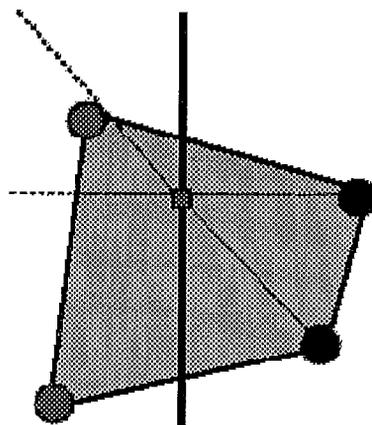
Para cada nó no interior do polígono, a imagem é particionada em quatro subespaços através das semi-retas indicadas na Figura 2.8:

Estas linhas são denominadas $L1$ e $L2$ na figura 2.8, e são formadas ligando o *snaxel* p_1 com o nó v e o *snaxel* p_2 com o nó v , respectivamente.

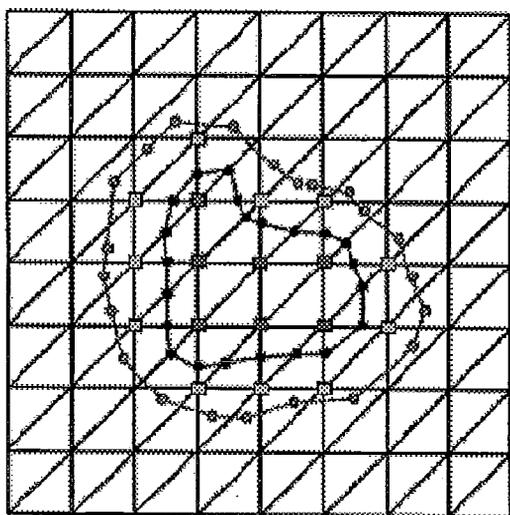
Durante um passo de evolução, p_1 e p_2 são movidos para novas posições p_{1n} e p_{2n} respectivamente. O passo máximo de cada *snaxel* é restringido a um valor muito menor que a dimensão do domínio da imagem, evitando assim comportamentos degenerados. Com isto, podemos assumir que p_{1n} e p_{2n} podem estar em qualquer um dos 4 subespaços da Figura 2.8. Portanto, temos um total de 16 possibilidades a considerar mostradas na Figura 2.9.



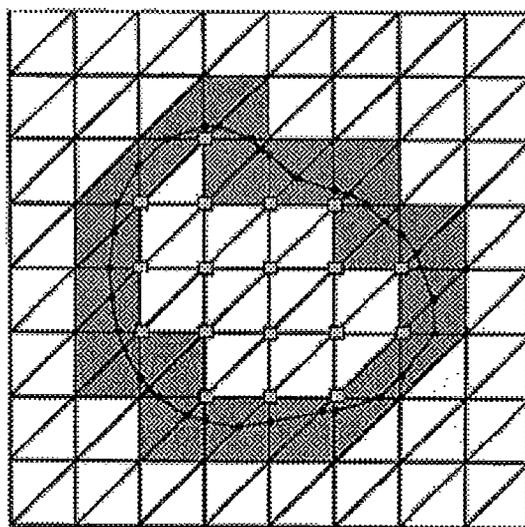
(a)



(b)



(c)



(d)

Figura 2.7: Segunda fase da Reparametrização da T-Snake. (a) Durante a expansão, T-Snake pode passar por vários vértices do grid, (b) particionamento do subespaço, (c) Modificação da entropia dos novos vértices do grid, (d) Nova T-Snake

Os quatro pontos p_1 , p_2 , p_{1n} e p_{2n} formam o polígono fechado Q que pode ser convexo ou não (Figura 2.9). Para tal, duas definições importantes devem ser consideradas (McInerney, 1997):

(1). Um ponto p é dito interior ao polígono Q se um raio partindo de p intercepta exatamente uma aresta de Q ou exatamente 3 arestas de Q , ou se p pertence à poligonal definida pelos vértices de Q .

(2). Um nó da malha v é dito “queimado” se ele for interior ao polígono Q .

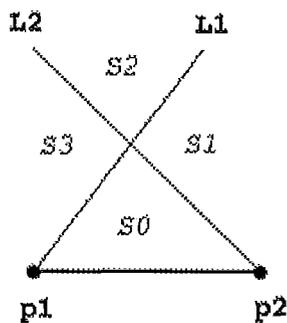


Figura 2.8: Subespaços usados para classificação de nós.

De acordo com estas definições, vemos que os casos (1), (2), (7) da Figura 2.9 classificam v como queimado, enquanto que os casos (3)-(6) e (8)-(12), classificariam v como não queimado. Assim, a simples classificação de p_{1n} e p_{2n} em um dos subespaços da figura 2.8 informa imediatamente se v foi queimado, ou não, para a maioria dos casos. Os casos com ambigüidade (casos (13)-(16)) necessitam de um teste adicional (McInerney, 1997).

Algoritmo das *T-Snakes*

Finalmente, podemos apresentar o algoritmo das *T-Snakes* como segue (McInerney, 1997).

Algoritmo:

Inicialização: Projetar as *Snakes* iniciais sobre a malha

Enquanto o ponto de congelamento não for atingido:

1. Computar as forças internas e externas e atualizar a posição dos *snaxels* de acordo com a equação (2.11).
2. Computar as intersecções entre a malha e os elementos de modelo.
3. Atualizar a função característica definida pela equação (2.12).
4. Usando a função característica determinar o correspondente conjunto de triângulos de borda usando o algoritmo *Door-In-Door-Out*.
5. Para cada aresta obtida da sequência de arestas, escolher um ponto dentre os calculados no item 2 para ser um novo *snaxel*. Se não houver nenhum, criar um ponto na aresta em questão. Descartar todas as outras intersecções.
6. Atualizar temperatura dos *snaxels*.

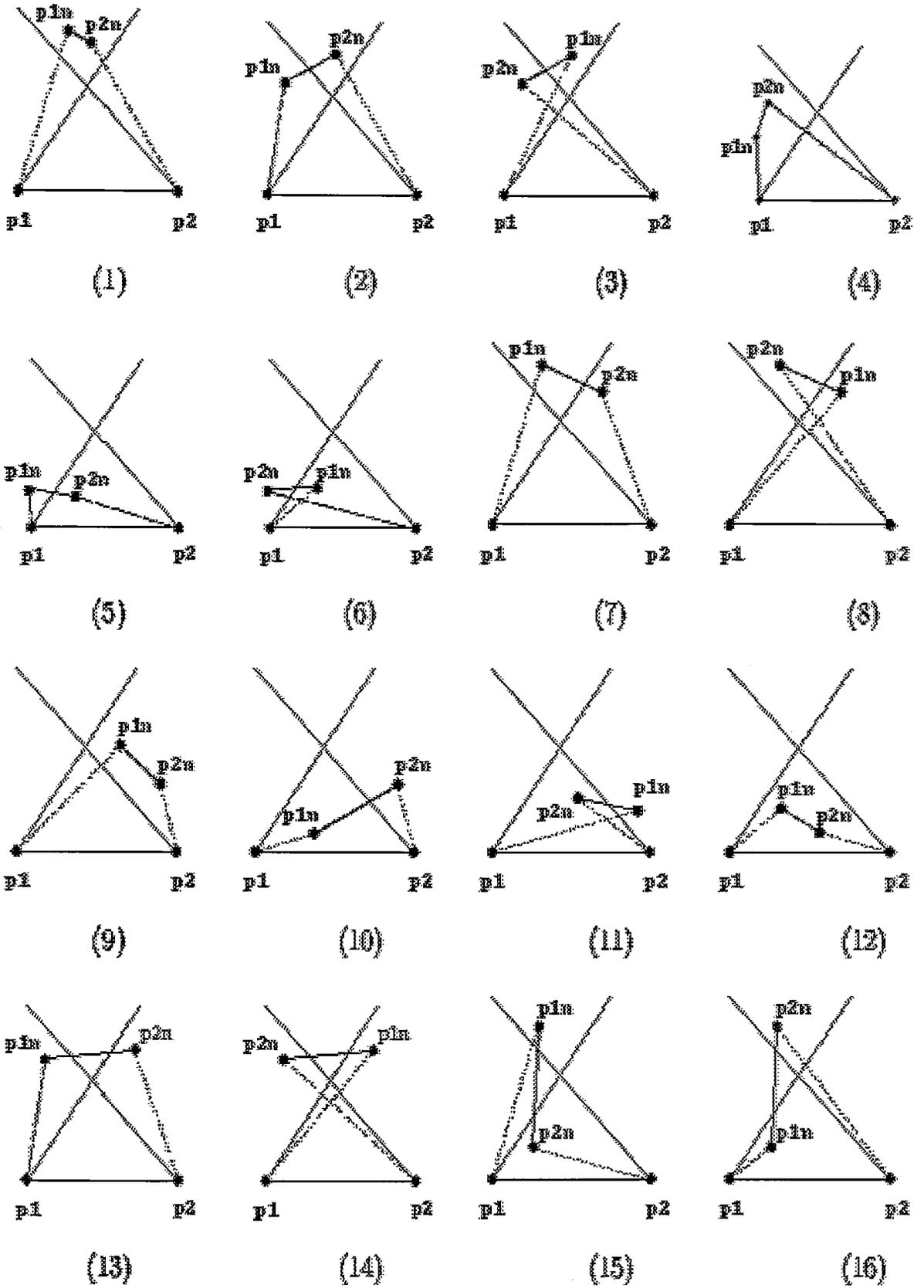


Figura 2.9: Possíveis casos para classificação.

2.2.4 Limitações das *T-Snakes*

McInerney (McInerney, 1997) discute algumas limitações da *T-Snake* e algumas soluções são propostas.

O primeiro ponto é a restrição à evolução imposta pela condição de entropia usada: “se um nó é queimado ele permanece queimado”. Esta política é vantajosa para mudanças topológicas, mas tem a desvantagem de limitar a evolução da *Snake* em apenas expansão, ou apenas contração, mas não para ambos. Esta característica pode dificultar a interação com o usuário caso esta se torne necessária.

Uma possível solução para este problema seria reverter a direção de evolução por um pequeno intervalo de tempo (McInerney, 1997).

A resolução da malha controla o grau da flexibilidade geométrica da *T-Snake*. Se o objeto de interesse contém cavidades profundas, ou protuberâncias finas, a resolução da malha deverá ser fina suficiente para permitir que a *T-Snake* se acomode à borda desejada.

Contudo, uma vez que estamos usando triangulações regulares, tal procedimento implica em um número excessivo de *snaxels* em regiões com baixa curvatura, trazendo problemas de performance.

A utilização de métodos adaptativos (Berger, 1986) seria uma solução caso fosse possível saber automaticamente onde a malha necessita ser refinada, o que na prática é difícil. No presente trabalho será apresentado uma possível solução para este problema, baseada em uma metodologia que parte de uma malha maior, refinando-a até a resolução desejada.

Outra limitação da *T-Snake* é o fato do método de reparametrização da *Snake* (projeção sobre a malha) depender não somente da forma do contorno mas também da maneira como o espaço foi dividido (decomposição simplicial), bem como da posição da *Snake* neste espaço.

Este ponto é menos problemático para a *T-Snake* uma vez que sua evolução não é baseada explicitamente na minimização de um funcional de energia e adota-se uma condição de entropia. Apesar disto, tal reparametrização perturba a *Snake*. Em última análise, este procedimento de reparametrização é o motivo pelo qual a *T-Snake* não é em geral um bom método de finalização exigindo que, após sua estabilização, a malha seja descartada e a *Snake* evolua independente da triangulação.

Esta reparametrização implicaria também na não-invariância da energia interna por transformações de translação e rotação, uma vez que os *snaxels* dependem da posição da *Snake* no espaço.

Capítulo 3

Modelo de T-Superfícies

Neste capítulo será apresentado o método das T-Superfícies, que é a extensão 3D das *T-Snakes*. Primeiramente, uma descrição do modelo será apresentada analisando-se as etapas necessárias para o sua evolução. A seguir, será discutida a decomposição 3D do espaço, e a forma com que o modelo utiliza essa divisão no processo dinâmico de alteração de forma, bem como suas características topológicas.

3.1 Descrição da T-Superfície

O modelo de T-Superfície corresponde a uma malha triangular elástica e fechada. Do ponto de vista topológico, uma T-Superfície é uma Variedade Linear por Partes de dimensão 2 (Seção 2.2.1). Os pontos da malha são denominados nós, ou snaxels, e os triângulos denominados *elementos triangulares*. Os vértices (nós) dos triângulos do modelo agem como um sistema dinâmico de partículas interligadas por pequenas “molas” (Vasilescu & Terzopoulos, 1992).

3.2 Inicialização do Modelo

Basicamente, o modelo da T-Superfície é composto por três componentes (McInerney & Terzopoulos, 1999): (1) uma triangulação do espaço (no nosso caso um subconjunto fechado $D \subset \mathbb{R}^3$); (2) um modelo discreto da superfície deformável; (3) uma Função Característica que distingue o interior do exterior da superfície S , respectivamente, $Int(S)$ e $Ext(S)$.

3.2.1 Decomposição Simplicial do Espaço

A decomposição simplicial 3D, como já vimos na seção 2.2.1, utiliza tetraedros. A triangulação 3D do tipo Coxeter-Freudenthal é obtida dividindo-se o espaço Euclidiano por um *grid* cúbico

uniforme e subdividindo cada cubo em seis tetraedros (Figura 3.1).

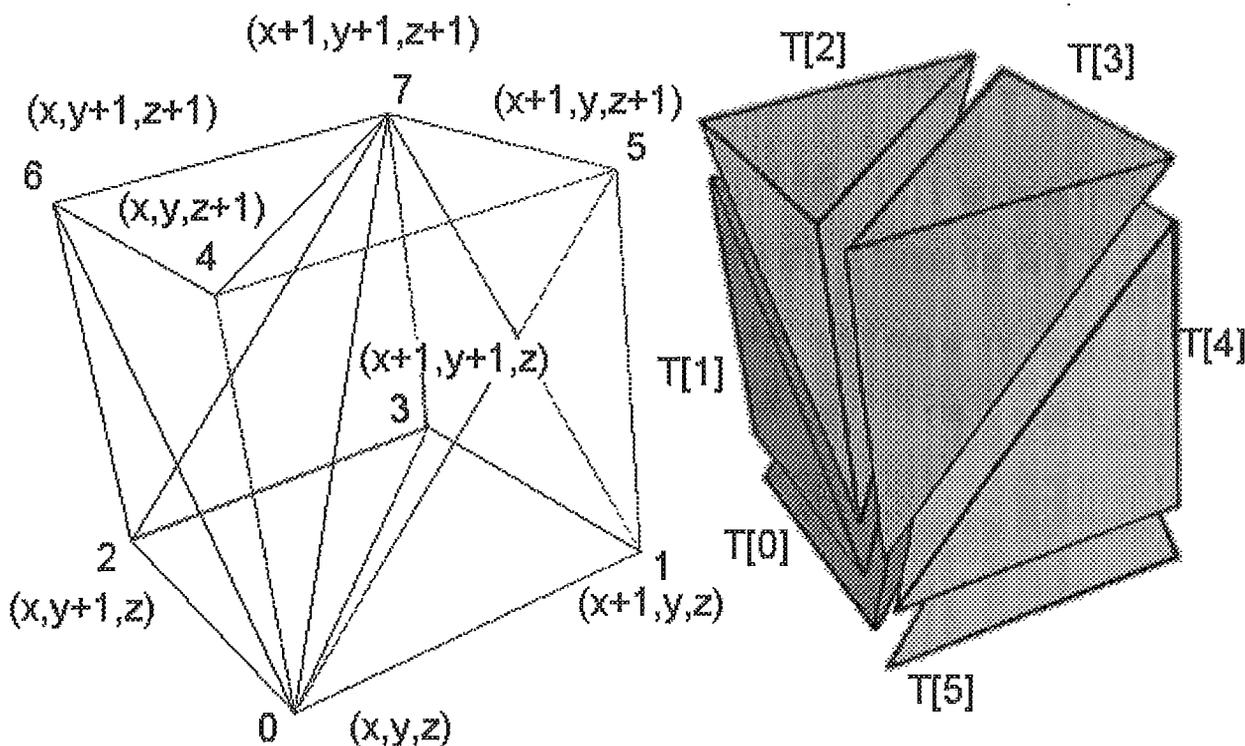


Figura 3.1: Célula cúbica decomposta em tetraedros.

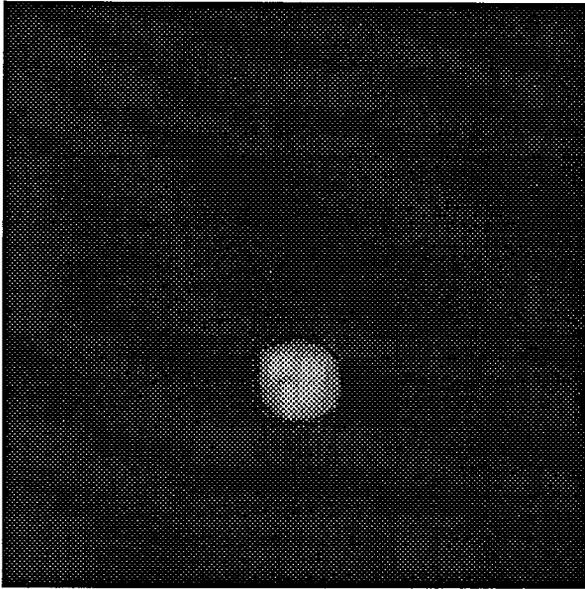
Os tetraedros do *grid* que interceptam a “casca” da superfície da estrutura anatômica que deseja-se aproximar são denominados *tetraedros transversos*.

3.2.2 Superfície Inicial

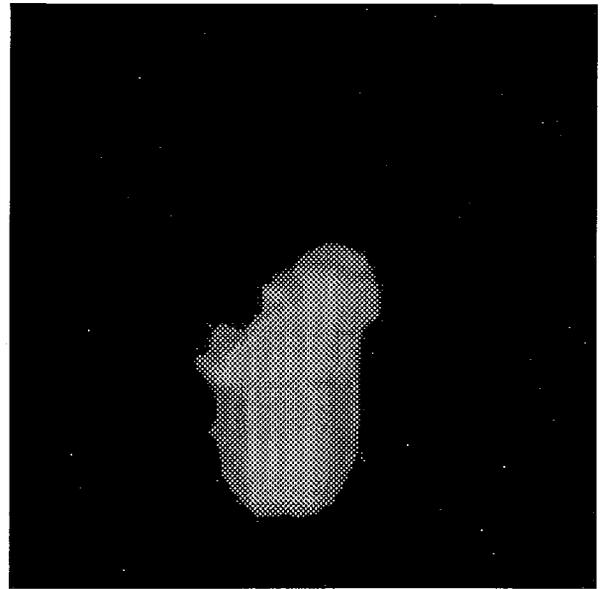
Após a definição da triangulação 3D do espaço, é necessário inicializarmos o método com uma superfície inicial. No modelo original das T-Superfícies, a superfície inicial (semente) era um objeto de topologia esférica posicionado em algum ponto dentro da área de interesse. Uma vez posicionado, o objeto deveria expandir-se até aproximar à superfície alvo, como ilustra a figura 3.2. Nesse exemplo, foram necessárias 49 iterações para obtermos o resultado final.

3.2.3 Função Característica Inicial

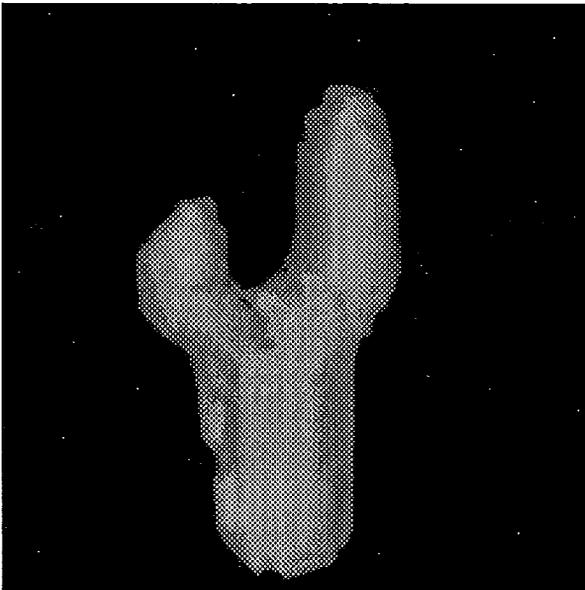
Para poder desenvolver o método, os padrões de intensidade (níveis de cinza) de um objeto O (ou do plano de fundo) devem ser caracterizados por um threshold T , ou estatísticas (μ e variação σ) de um campo de imagem I (McInerney & Terzopoulos, 1999).



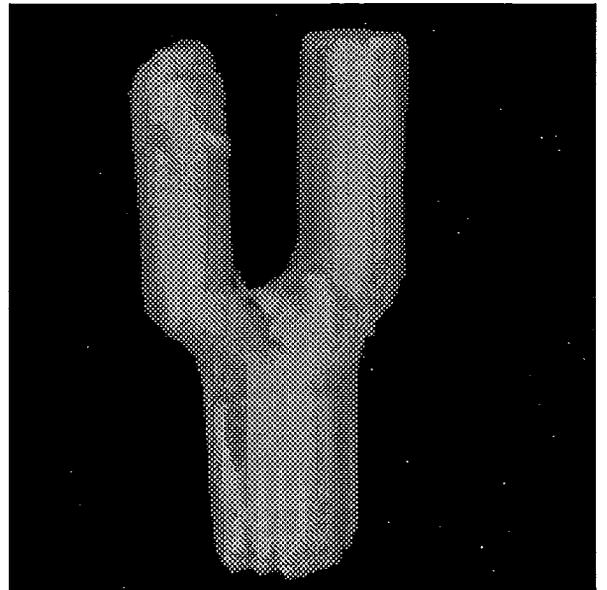
(a)



(b)



(c)



(d)

Figura 3.2: Evolução da T-Superfície através de uma semente esférica. (a) Primeira evolução. (b) Iteração 19. (c) Iteração 32. (d) Iteração 49 (final).

Inicialmente, assumi-se uma propriedade de escala local. Dado um ponto $p \in O$, sendo r_p o raio de uma *hyperball* B_p , que contem p e encontra-se inteiramente dentro da região do objeto, sendo $r_p > 1$, tem-se que:

$$p \in O \implies I(p) > T \text{ or } |I(p) - \mu| \leq k\sigma \quad (3.1)$$

Como observado na Seção 2.12, pode-se definir uma função simples, denominada Função Característica, como segue:

$$\chi : D \subset \mathbb{R}^3 \rightarrow \{0, 1\} \quad (3.2)$$

onde $\chi(p) = 1$ se $p \in O$ e $\chi(p) = 0$ caso contrário, sendo p um nó da triangulação 3D.

Ao inicializarmos o método das T-Superfícies com uma forma geométrica simples (esfera), os nós da triangulação interiores a esfera serão os nós onde $\chi(p) = 1$. Todos os nós restantes, independente do seu valor em relação ao *threshold* estabelecido, terão valor 0 (zero).

3.2.4 Armazenamento dos tetraedros transversos

O armazenamento dos tetraedros transversos é uma etapa importante no processo das T-Superfícies, pois os elementos triangulares que compõem a superfície estarão contidos nesses tetraedros. O problema de encontrar os tetraedros transversos é semelhante ao problema visto na seção 2.2.1 para encontrar os triângulos de borda para o caso 2D.

No algoritmo *Door-In-Door-Out* (Seção 2.2.1), cada triângulo *transverso* tem apenas uma aresta completamente rotulada de entrada, e outra aresta completamente rotulada de saída. No caso 3D, pode-se ter mais de uma saída, tornando o algoritmo mais complexo, como mostrado a seguir:

Algoritmo de Continuação 3D

Encontrar um tetraedro *transverso* σ_0 ;

$\Sigma = \{\sigma_0\}$;

$V(\sigma)$ = conjunto de vértices de σ ;

Enquanto $V(\sigma) \neq 0$ para algum $\sigma \in \Sigma$

 selecionar $\sigma \in \Sigma$ tal que $V(\sigma) \neq 0$;

 selecionar $v \in V(\sigma)$;

 obter σ' de σ pivoteando de v para v'

se σ' não for *transverso* **então**

 retire v de $V(\sigma)$;

senão

se $\sigma' \in \Sigma$ **então**

 retire v de $V(\sigma)$, v' de $V(\sigma')$

senão

$\Sigma \leftarrow \Sigma + \sigma'$;

$V(\sigma') \leftarrow$ conjunto de vertices de σ' ;

retire v de $V(\sigma), v'$ de $V(\sigma')$;
fim-se
fim-se
fim-Enquanto

Após a execução do *Algoritmo de Continuação 3D*, pode-se iniciar a projeção da superfície inicial.

3.2.5 Projeção da Superfície Inicial

A projeção da superfície é obtida através do cálculo da intersecção entre a superfície inicial e as arestas dos tetraedros transversos. Cada tetraedro de borda conterá um ou dois elementos triangulares formados pelos pontos gerados nas intersecções (Seção 3.2.6). Ao final das projeções, uma aresta poderá conter um ou mais pontos. É necessário usar um critério de decisão para definir qual ponto será utilizado em cada aresta. No presente trabalho, foi escolhido o ponto mais externo, calculado através da distância em relação a extremidade de valor 1 na aresta.

Esse processo é válido quando se inicializa o modelo com uma forma geométrica simples, pois na inicialização via *threshold*, não se tem uma superfície inicial. Para esse caso, geramos automaticamente os pontos referentes as intersecções do caso anterior através do cálculo dos pontos médios das arestas *transversas*.

3.2.6 Reconstrução do Modelo

Cada tetraedro *transverso* conterá algumas arestas com pontos que pertencem à superfície. Esses pontos podem formar um triângulo ou um quadrilátero (que será dividido em dois triângulos) aproximando a superfície dentro de cada um dos tetraedros transversos (Figura 3.3).

Os triângulos (quadriláteros) separam os vértices positivos dos vértices negativos dos tetraedros. O conjunto de todos os triângulos contidos nos tetraedros transversos formam a T-Superfície inicial.

3.3 Etapa Dinâmica

Sendo a T-Superfície um modelo dinâmico, é necessário utilizarmos um sistema de forças para “evoluir” a superfície na direção dos objetos de interesse. Várias etapas já vistas no processo

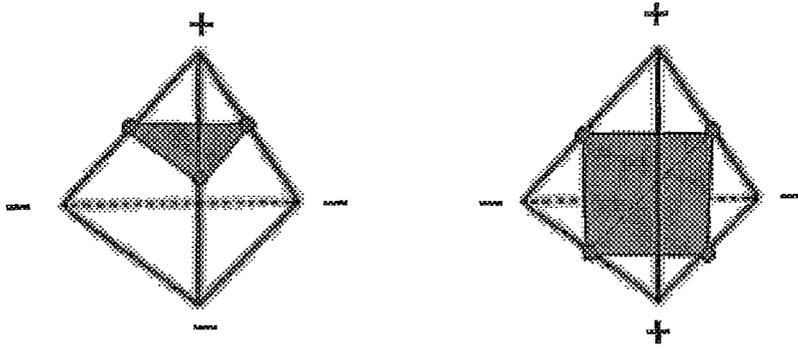


Figura 3.3: Exemplos de tetraedro de borda. Os sinais indicam se o vértice é interno (positivo) ou externo (negativo).

de inicialização serão utilizadas na Etapa Dinâmica, por esse motivo, o enfoque desta seção será nas novas etapas necessárias para concluir a evolução da T-Superfície.

3.3.1 Evolução da T-Superfície

O comportamento da T-Superfície é governado pelo seguinte sistema dinâmico:

$$v_i^{(t+\Delta t)} = v_i^t + h_i \left(\vec{\alpha}_i^t + \vec{F}_i^t + \vec{f}_i^t \right), \quad (3.3)$$

onde h_i é o passo da evolução e v_i um nó da malha. A cada nó v_i estão associadas forças elásticas, força normal e forças de imagem.

Forças Elásticas

O principal objetivo da Força Elástica é minimizar a curvatura local da superfície. Os nós são interligados por *molas* de comprimento natural. Desta forma, dada a deformação $r_{i,j} = \|v_i(t) - v_j(t)\|$, podemos definir a força elástica através da equação:

$$\vec{\alpha}_i = c \sum_j \vec{r}_{ij}, \quad (3.4)$$

onde c é um fator de escala (Figura 3.4).

Força Normal

A Força Normal é usada para mover a superfície em direção a borda do objeto até ser pressionada por alguma força em sentido contrário, neste caso, as forças de imagem. Como em (McInerney & Terzopoulos, 1999), esta força é governada pela equação:

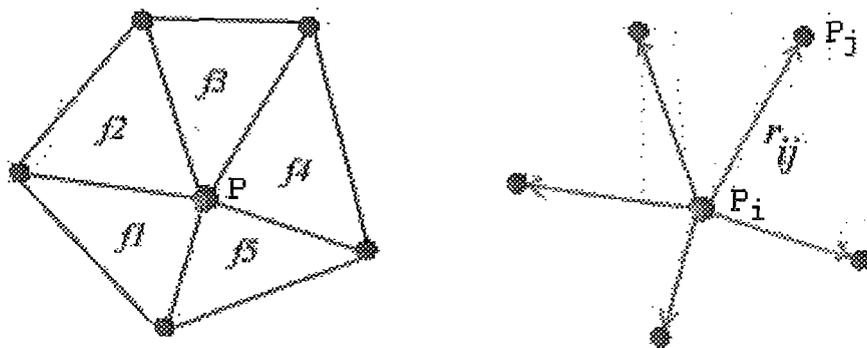


Figura 3.4: Diagrama da Força Elástica sobre um ponto P pertencente às faces f_1, f_2, f_3, f_4 e f_5 .

$$\vec{F}_i = k \cdot \text{sign}_i \frac{M \vec{n}_i}{\|M \vec{n}_i\|}, \quad M \vec{n}_i = \sum \frac{\vec{n}_i}{l} \quad (3.5)$$

onde n_i é o vetor referente ao nó i , $\text{sign}_i = +1$ se $I(v_i) < T$, e $\text{sign}_i = -1$ caso contrário (T é definido na expressão (3.1), $I(v_i)$ é a intensidade da imagem em v_i), k é o fator de escala e l o número de faces que contêm o ponto P , como mostra a Figura 3.5.

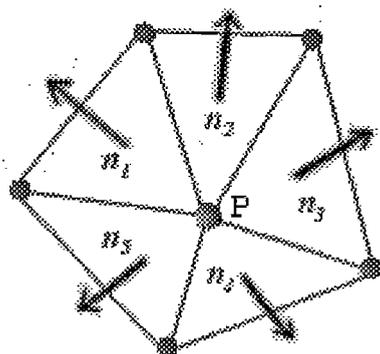


Figura 3.5: Exemplo de Força Normal sobre o ponto P com $l = 5$.

Forças Externas

A Força Externa ou Força de Imagem é utilizada para “puxar” o modelo em direção aos objetos de interesse na imagem. Essa força atua no sistema em oposição à força normal quando o modelo está próximo das bordas dos objetos que desejamos encontrar.

A força externa é dada por:

$$f_i^t = -\gamma_i \nabla \|\nabla I\|^2. \quad (3.6)$$

3.3.2 Atualização da Função Característica

Na formulação original das T-Superfícies, a atualização da função característica é idêntica ao caso das *T-Snakes* (Seção 2.2.2). Durante a evolução, cada elemento triangular pode ter passado sobre zero, um, ou vários nós do *grid*. Formam-se poliedros no formato de um prisma utilizando os elementos triangulares na sua posição evoluída e na sua posição anterior. Os lados do prisma são subdivididos em triângulos, formando um poliedro de 8 lados. Para determinar se um vértice do *grid* foi queimado por um elemento triangular durante uma evolução, deve-se determinar se o vértice está contido no poliedro.

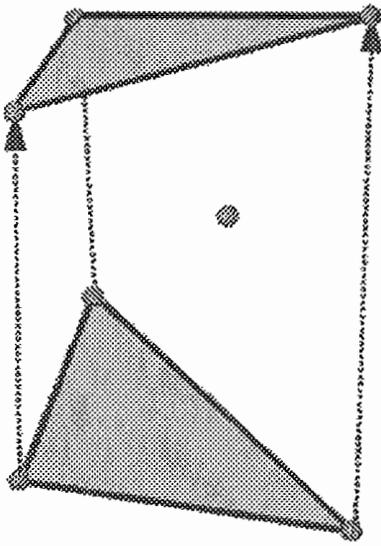
O caminho percorrido por cada elemento triangular definirá a complexidade da forma do poliedro. Ao invés de trabalhar diretamente com a forma complexa do poliedro gerado, foi usado em (McInerney, 1997) um algoritmo que dependia apenas dos vértices dos elementos triangulares em sua posição original e sua nova posição (Figura 3.6).

O algoritmo de classificação 3D utilizado é uma extensão do algoritmo de classificação aplicado às *T-Snakes*. Inicia-se o processo dividindo-se o espaço em 8 subespaços através da definição de 3 planos. Os planos são formados pela junção dos dois nós de cada aresta dos elementos triangulares com um vértice do *grid*. Então, classificam-se os 3 nós do elemento triangular em sua nova posição $p1n$, $p2n$, $p3n$ em relação aos 8 subespaços formados. Ao final, tem-se 512 possíveis combinações de classificação.

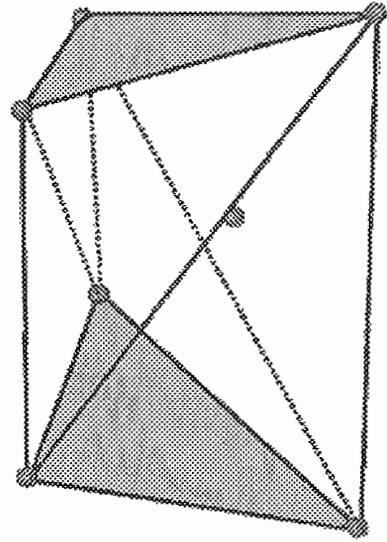
3.3.3 Critérios de Parada

Como visto na seção 2.2.2, a T-Superfície adota um critério de parada semelhante ao utilizado no método das *T-Snakes*. Esta condição tem também a finalidade de permitir a definição de um critério de parada eficiente.

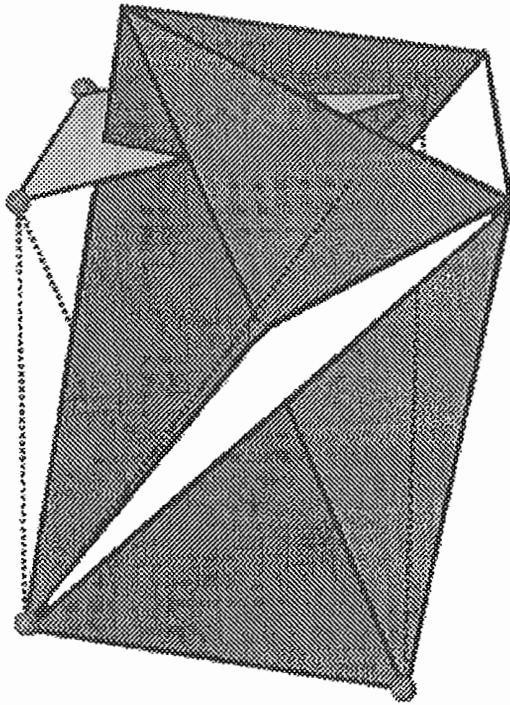
De forma análoga ao caso das *T-Snakes*, define-se uma *temperatura* para cada *snaxel*, obtida através da informação sobre o número de deformações (iterações) que o tetraedro foi classificado como sendo um tetraedro transverso. Uma T-Superfície é considerada em equilíbrio quando a temperatura de todos os seus *snaxels* ultrapassar um limiar denominado *ponto de congelamento*.



(a)



(b)



(c)

Figura 3.6: (a) Evolução do elemento triangular, (b) triangulação das faces formando poliedro de 8 faces e (c) Formação dos 8 subespaços.

Capítulo 4

Modelo Proposto

Neste capítulo, serão apresentadas as contribuições deste trabalho utilizando o modelo das T-Superfícies além de suas aplicações. Será apresentada uma nova abordagem que integra o modelo de T-Superfícies e conceitos de multi-resolução numa metodologia unificada para a segmentação e a reconstrução de superfícies. Em relação ao problema da atualização da Função Característica, foi feito um estudo que possibilitou realizar a atualização desta função através da análise da vizinhança de cada elemento triangular após sua evolução. Para tratar imagens com ruído, foi proposta a utilização de filtragem por difusão anisotrópica. Para melhor explorar as capacidades topológicas do modelo de T-Superfícies é usada, quando necessário, a interação entre o usuário e o modelo.

4.1 Inicialização utilizando *Threshold*

O enfoque principal desse trabalho foi desenvolver uma metodologia que permitisse criar uma superfície inicial mais próxima dos objetos de interesse. A utilização de dados da imagem (*threshold*) e conceitos de multi-resolução reduzem consideravelmente o número de iterações do método aumentando sua eficiência.

4.1.1 Multi-Resolução

Para utilizar os conceitos de multi-resolução, é necessário definir uma propriedade de escala local (Seção 3.2.3). Isso implica em poder reduzir a resolução da imagem sem perdermos os objetos de interesse. Com isso, foi incorporado ao modelo a filosofia básica de algumas técnicas não-paramétricas de multi-resolução usadas em segmentação de imagens (Jolion & Montanvert, 1992): reduzindo a resolução, pequenos artefatos da imagem tornam-se menos

significativos em relação ao(s) objeto(s) de interesse.

A técnica proposta é classificada como adaptativa no sentido de podermos aumentar a resolução nos locais onde for necessário. Para aumentarmos a resolução, simplesmente refinamos o *grid* inicial e tiramos uma amostragem da imagem sobre os nós do *grid* correspondente. As diversas componentes que formam a T-Superfície são tratadas separadamente, estando a resolução do *grid* para cada uma delas relacionado com as características de cada componente. Desta maneira, pode-se tratar num mesmo modelo, várias componentes com resoluções diferentes. O restante das células que não pertencerem à componente que está sendo processada não será computado. Mesmo as células que contiverem tetraedros transversos podem ser retiradas da computação durante a evolução, desde que os elementos triangulares contidos nessas células estejam congelados (Seção 3.3.3).

Através do *threshold*, “marcam-se” todos os nós do *grid* tidos como válidos. Desta forma, regiões de dimensão inferior ao tamanho das células do *grid*, que possuam intensidade de imagem válida serão, em geral, descartadas. A superfície inicial será composta pelos elementos triangulares formados pela união dos pontos contidos nas arestas *completamente rotuladas*, obtidas através dos nós do *grid* que foram marcados anteriormente (Seção 2.2.1).

A Figura 4.1 mostra um exemplo de inicialização via *threshold*. As primeiras iterações geralmente já fornecem uma superfície bastante próxima da superfície de interesse. Para esse exemplo, foram necessárias 9 iterações para obter o resultado final.

4.2 Atualização da Função Característica

Como visto na Seção 3.2.3, os nós da triangulação interiores à superfície inicial serão os nós onde $\chi(p) = 1$. No caso da inicialização através de *threshold*, todos os nós serão “marcados”, inicialmente, de acordo com a Equação 3.1. Em uma segunda etapa, serão mantidos apenas os nós que estiverem contidos nas regiões de interesse, descartando-se pequenos artefatos na imagem.

Uma nova forma de atualizar a Função Característica foi desenvolvida para esse trabalho. A atualização é baseada no estudo dos vértices vizinhos do *grid*, em relação aos elementos triangulares evoluídos. Durante a deformação da superfície, um elemento triangular pode passar por zero, um ou vários vértices do *grid*. Supondo que durante o deslocamento, em um dos eixos coordenados do *grid*, um elemento triangular passe por um vértice não queimado. Ao atingir sua posição final, o vértice só terá vizinhos não queimados. Percorrendo-se o eixo na direção contrária à normal do elemento triangular evoluído, haverá um vértice queimado. Essa

situação indica que, provavelmente, o sinal do vértice não queimado, encontrado entre o elemento triangular e o vértice queimado no eixo de deslocamento, deverá ser modificado (Figura 4.2). Para garantir que um determinado vértice do *grid* realmente sofrerá mudança de entropia, o teste deve ser feito em todos os eixos. Se em todos os eixos o teste for positivo, o vértice será queimado. Se em um dos eixos o teste falhar, o vértice permanecerá como candidato para um segundo teste, que consiste na repetição da avaliação anterior, levando-se em conta os novos vértices já queimados.

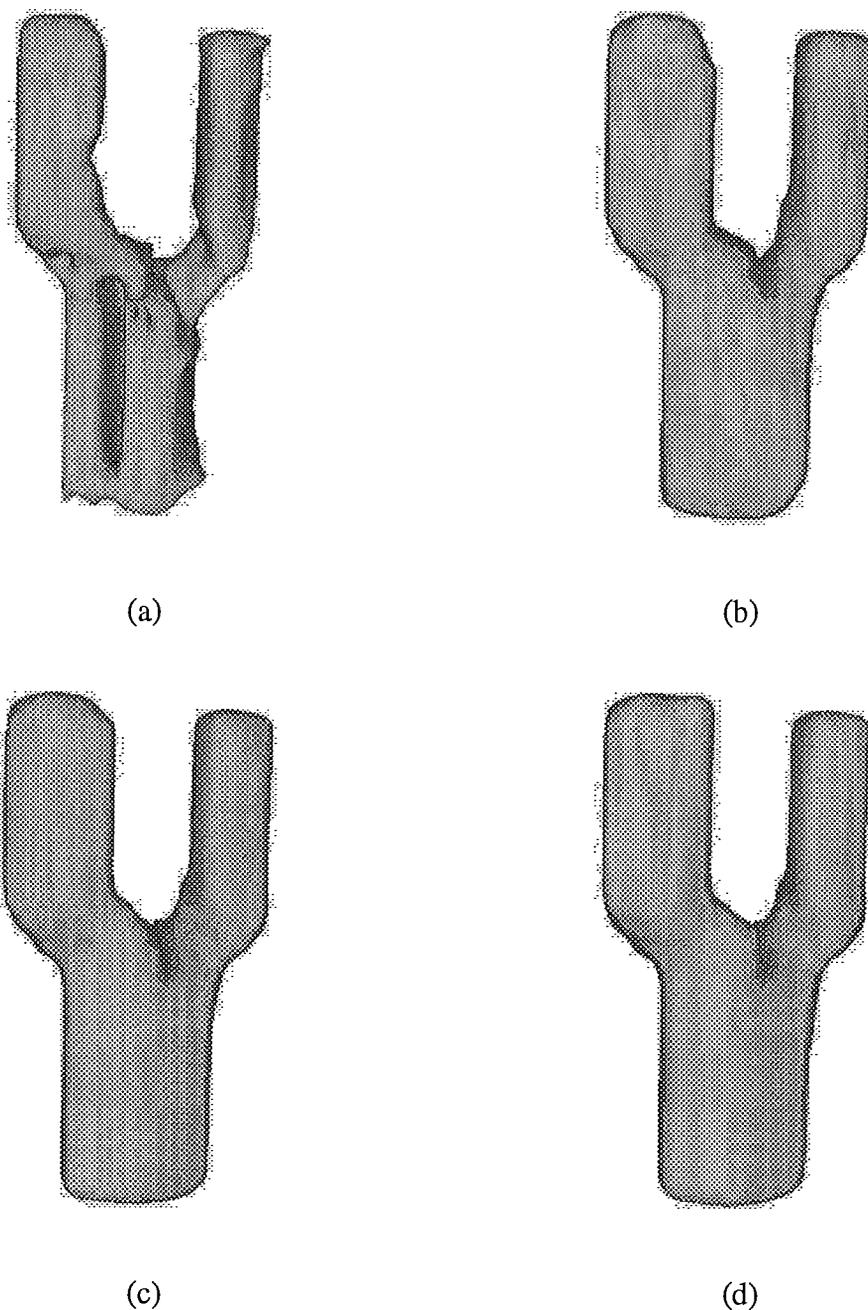


Figura 4.1: Evolução da T-Superfície através de inicialização via threshold.
(a) Primeira evolução. (b) Iteração 3. (c) Iteração 6. (d) Iteração 9 (final).

Devem-se tratar, separadamente, os casos onde há dois pontos ($P1$ e $P2$) de intersecção num mesmo eixo coordenado do *grid* em células vizinhas do nó candidato. Nesses casos, se a componente estiver expandindo-se, o eixo onde estão os elementos triangulares será descartado, caso contrário, o teste é realizado como visto anteriormente (Figura 4.3).

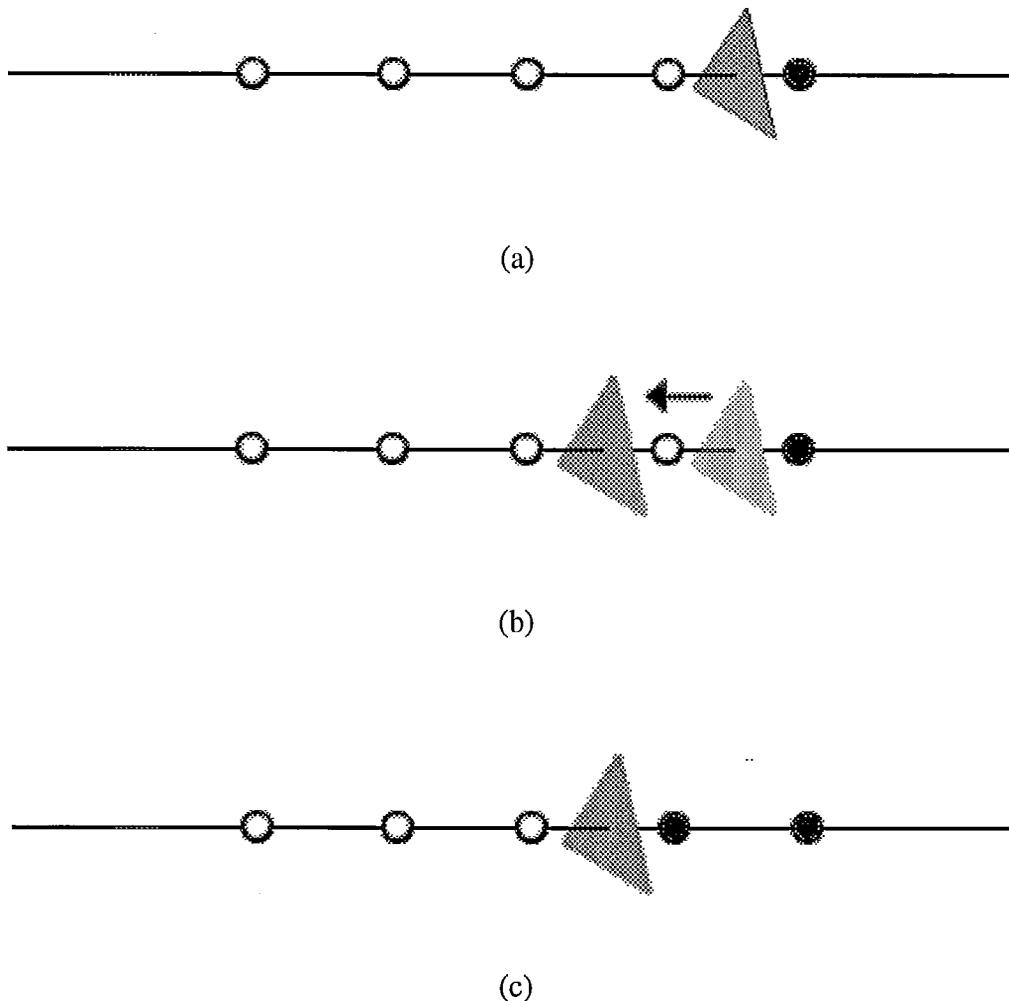


Figura 4.2: Visão do processo de atualização em um dos eixos coordenados. (a) Inicialmente, elemento triangular encontra-se na sua posição original. (b) Ocorre deformação do modelo e elemento triangular altera posição. (c) Vértice do *grid* por onde elemento triangular passou é queimado.

4.3 Filtragem por Difusão Anisotrópica

Em geral, as imagens utilizadas no método das T-Superfícies necessitam de uma etapa de pré-processamento antes de serem identificados os padrões de intensidade dos objetos / plano de fundo, principalmente em se tratando de imagens com ruído. Nesse trabalho, foi realizado um

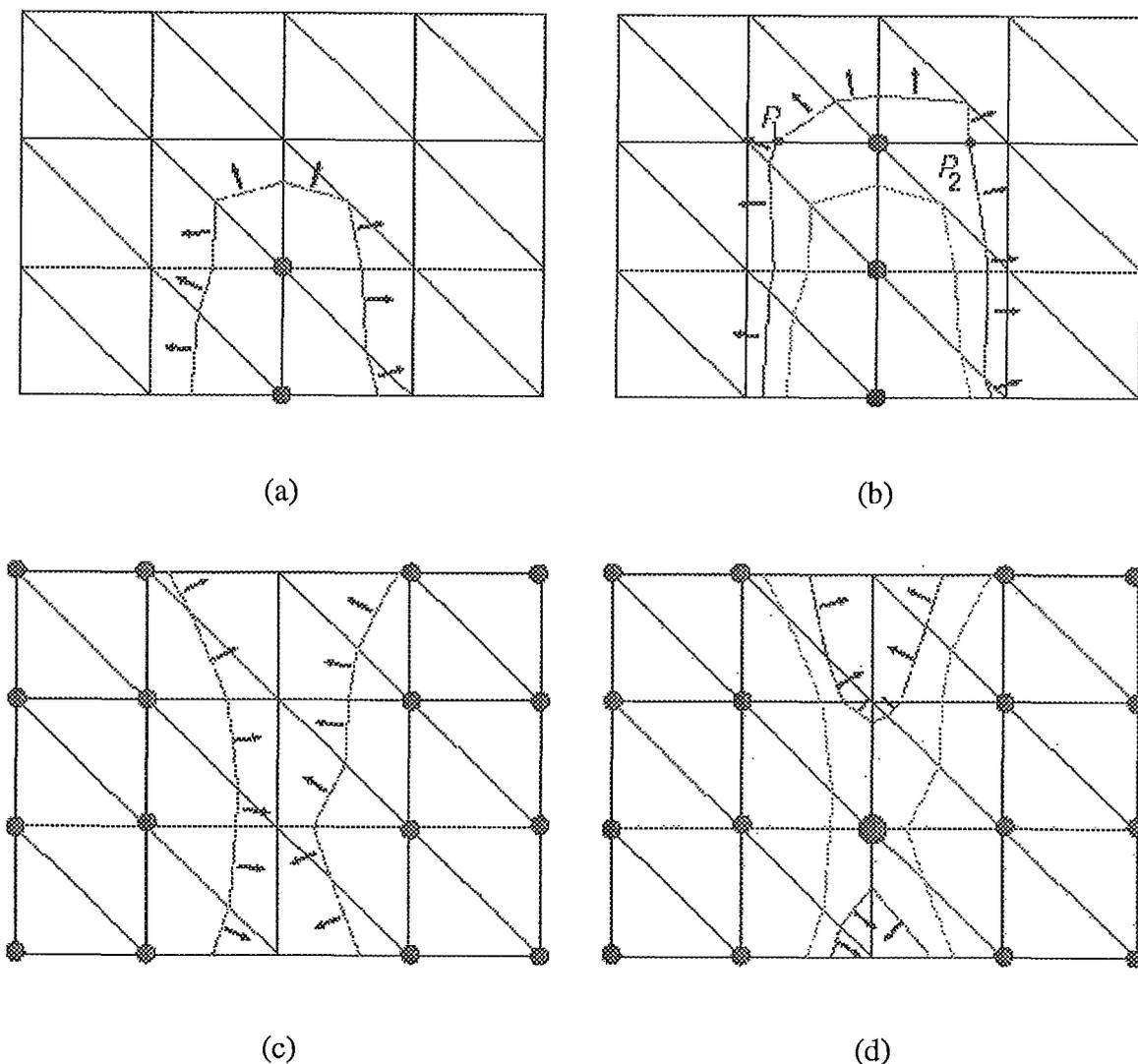


Figura 4.3: Atualização da Função Característica para o caso de 2 células vizinhas contendo elementos triangulares. (a) Caso de expansão. (b) Expansão após evolução. (c) Duas componentes se aproximam (d) Operação de Merge.

breve comparativo entre dois métodos de filtragem: Difusão Gaussiana e Difusão Anisotrópica (Perona & Malik, 1990).

O método de filtragem por difusão Gaussiana filtra a imagem de forma homogênea, fazendo com que as bordas dos objetos de interesse sofram o mesmo processo que os demais artefatos presentes na imagem. O método de Filtragem por Difusão Anisotrópica usado neste trabalho, é definido pela equação:

$$\frac{\partial I}{\partial t} = \text{div} \left(\frac{\nabla I}{(1 + \|\nabla I\|/K)^2} \right), \quad (4.1)$$

onde a constante K pode ser determinada através de um histograma de magnitude do gradiente. Esse método de redução de ruído é especialmente interessante quando aplicado ao método

das T-Superfícies pois possibilita “borrar” pequenas descontinuidades (magnitude do gradiente abaixo de K), assim como realçar as bordas (magnitude do gradiente acima de K), auxiliando desta forma no reconhecimento e posterior extração da superfície.

A figura 4.4 ilustra um teste com cada um dos métodos utilizando o mesmo volume de dados. Pode-se observar que a topologia da superfície inicial não é esférica, ou seja, contém alguns buracos (*holes*). As propriedades topológicas do modelo de T-Superfícies possibilitam corrigir esses defeitos.

Para o exemplo da Figura 4.4, foram necessárias 17 iterações para obter o objeto de interesse. Os parâmetros utilizados na T-Superfície foram: $c = 0.65$, $k = 1.32$ e $\gamma = 0.01$. A resolução do *grid* foi de $5 \times 5 \times 5$, ponto de congelamento igual a 15 e $T \in (120, 134)$.

4.4 Interação com o modelo de T-Superfícies

Os métodos descritos neste capítulo auxiliam na extração de objetos de interesse em um volume de dados, mas há casos em que se faz necessária uma intervenção manual.

O método de segmentação desenvolvido dá suporte a realização de operações de *split* manuais através de dois passos: (a) Definir um plano de corte; (b) Zerar os nós do *grid* que pertencem aos tetraedros interceptados pelo plano de corte e que são interiores à T-Superfície gerada.

Não haverá mais mudança de entropia nos nós do *grid* que foram zerados. Desta forma, evita-se a intersecção entre as componentes desconectadas. Assim, pode-se garantir eficientemente que as duas T-Superfícies envolvidas na operação de *split*, não serão conectadas após evoluir o modelo (Figura 4.5).

4.5 Offsets

Nesta seção, será abordada a proposta desenvolvida para tratar o problema de geração de *offset* para superfícies 3D. Inicialmente, assumi-se que temos uma Variedade Linear por Partes P (Subseção 2.2.1). Chama-se *n-offset* de P a superfície poligonal obtida após n interações da T-Superfície inicializada através de P com $\gamma = 0$ na Equação 3.6.

As auto-intersecções que podem ocorrer durante as deformações da superfície são facilmente tratadas pelo modelo das T-Superfícies. Desta forma, pode-se preservar a topologia da superfície P .

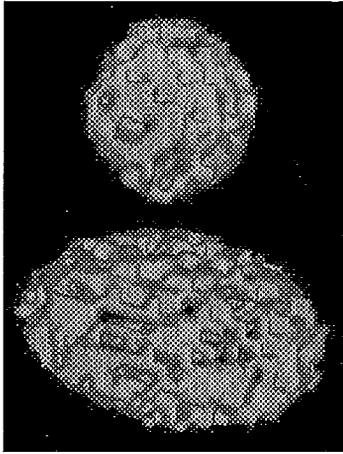
Após gerarmos o *offset*, a superfície criada é mais suave que a superfície inicial. A

deformação pode ser vista como um processo de difusão da curvatura no qual a velocidade de cada ponto da superfície depende da curvatura da mesma.

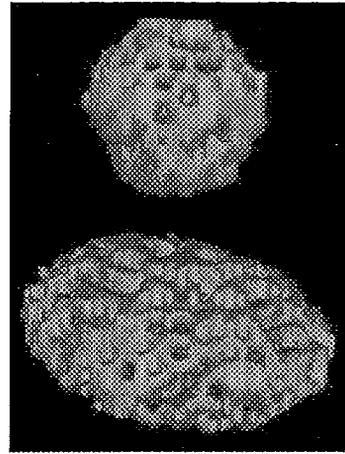
A superfície representada pelo *offset* é útil se for necessário reconstruir a “casca” externa dos objetos de interesse. Este *offset*, poderá ser usado como superfície inicial de uma T-Superfície em processo de contração. Desta maneira, a T-Superfície seria inicializada próxima a parede externa dos objetos que pretende-se reconstruir (Veja Capítulo 5).

4.6 Overview do método

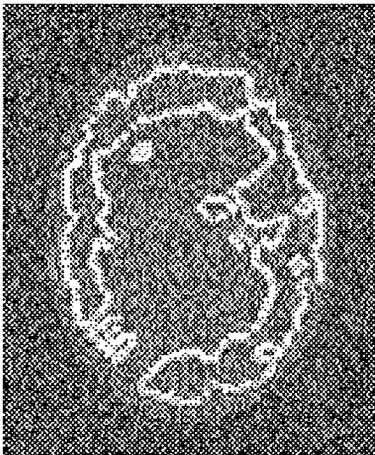
O diagrama apresentado na Figura 4.6 descreve as etapas propostas para a construção da T-Superfície, enfatizando-se as etapas de inicialização e evolução até o seu ponto de equilíbrio.



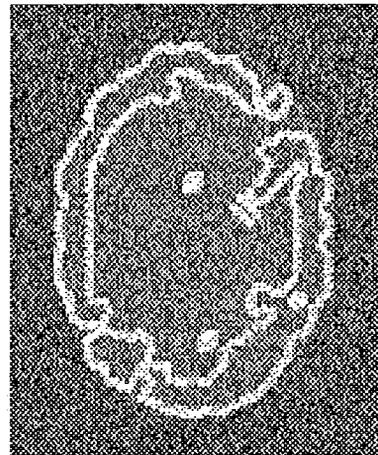
(a)



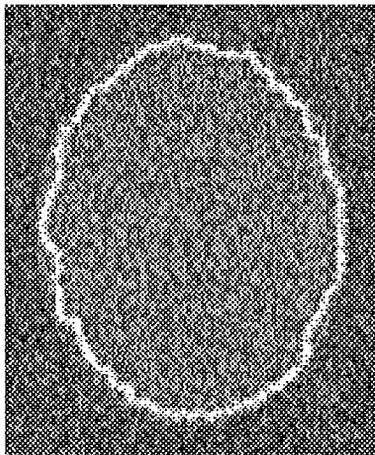
(b)



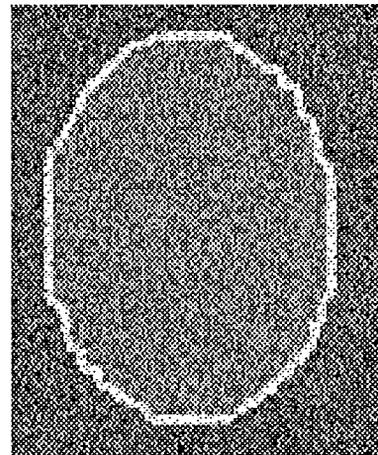
(c)



(d)

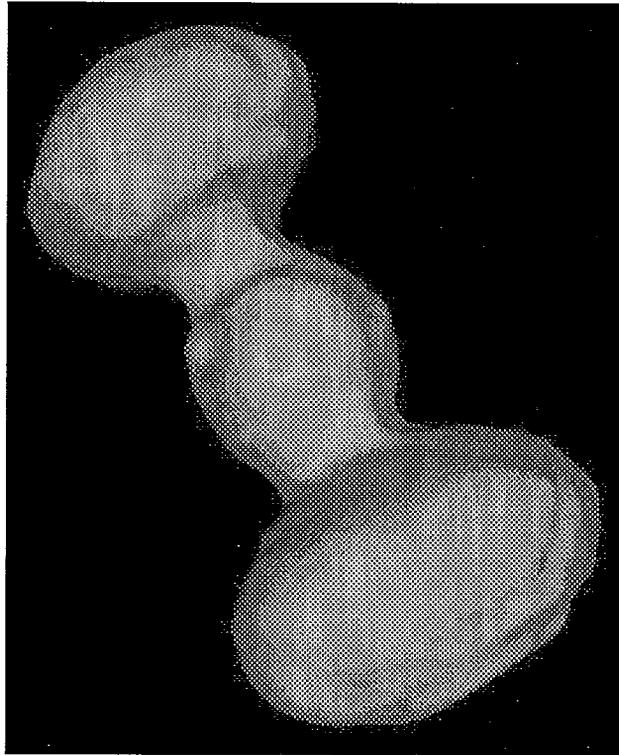


(e)

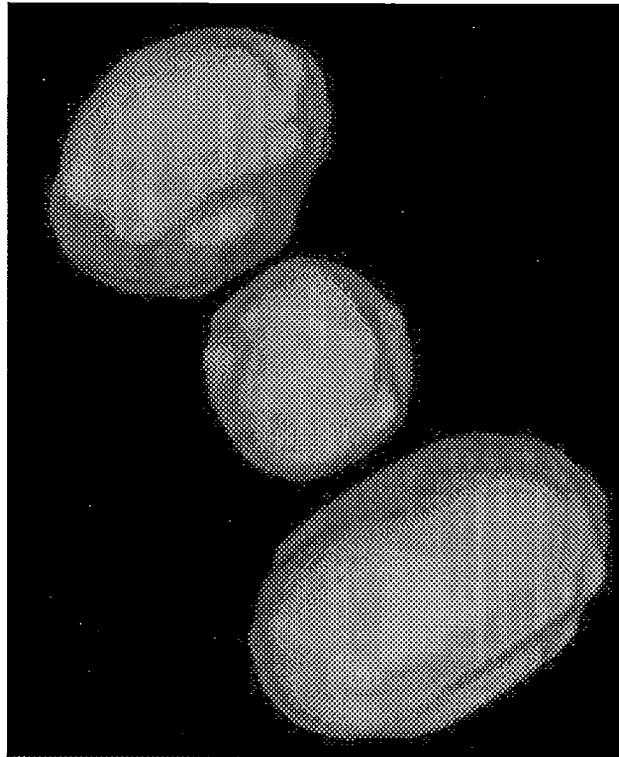


(f)

Figura 4.4: Em (a)-(b) temos o resultado da inicialização da via *threshold* com difusão Gaussiana e Anisotrópica, respectivamente. (c)-(d) Corte de (a),(b) para *slice* 40, respectivamente, sobre a imagem original. (d)-(e) Solução Final para (c),(d).



(a)



(b)

Figura 4.5: (a) T-Superfície antes e (b) após corte manual.

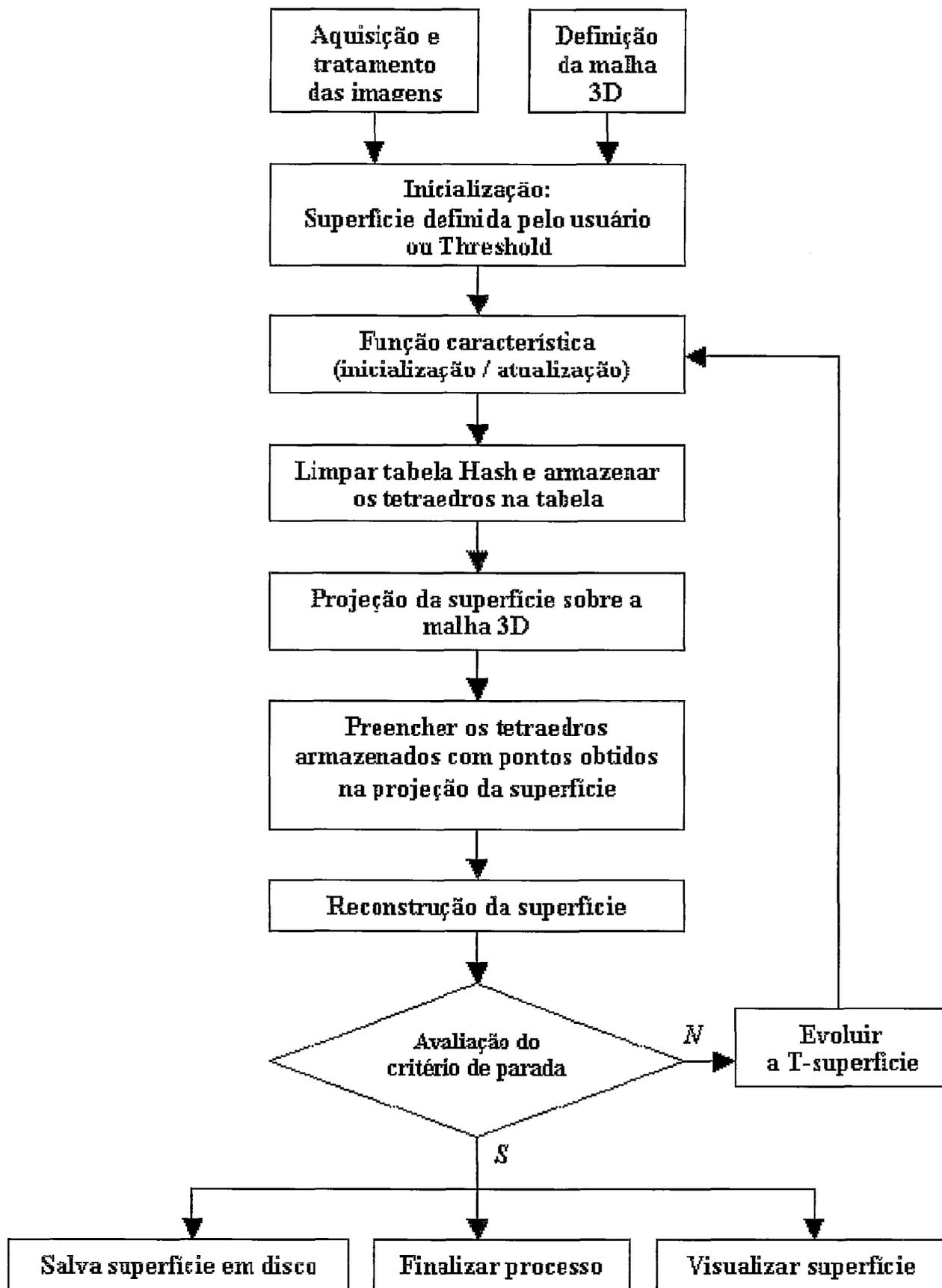


Figura 4.6: Diagrama das etapas da T-Superfície.

Capítulo 5

Análise dos Resultados

Neste capítulo, serão apresentados os resultados experimentais do método implementado, utilizando-se volumes de imagens sintéticas e reais. Deve ser observado que os valores atribuídos a parâmetros tais como intensidade de forças, resolução da malha, entre outros, foram definidos de forma empírica, através de tentativa e erro.

Inicialmente, aborda-se a precisão do método através de imagens sintéticas. A precisão foi calculada através do estudo da média do erro e de seu desvio padrão. Foram utilizados elipsóides e esferas e suas respectivas fórmulas matemáticas na avaliação dos cálculos.

Em relação aos métodos de inicialização, foi realizado um comparativo entre métodos tradicionais das T-Superfícies e a metodologia desenvolvida nesse trabalho baseada no estudo do *threshold*. Durante o processo de análise, foram considerados o número de iterações e o tempo de execução para um mesmo volume de imagens sintéticas.

O método das T-Superfície requer, na maioria dos casos, um pré-processamento do volume de imagens a ser utilizado. Será apresentado um exemplo onde a aplicação da filtragem por difusão anisotrópica possibilita a correta segmentação de objetos em um volume de imagens real.

A resolução da malha a ser utilizada no processo de segmentação está diretamente relacionada com o desempenho e a eficiência do método das T-Superfícies. Como a metodologia desenvolvida é multi-resolução, será apresentado um caso onde a utilização dessa técnica auxiliou na extração de objetos de interesse num volume de imagens real. Nesse exemplo, a filtragem não ajudou devido a fina espessura das extremidades das estruturas que foram reconstruídas.

Finalmente, será apresentado um exemplo de geração de *offset*, além uma proposta para sua utilização.

Deve ser observado que as análises numéricas apresentadas neste capítulo tem como base a noção de tempo em minutos e segundos, utilizando equipamento especificado na Seção A.4.

5.1 Precisão do Método

Para o caso de imagens 2D, a forma mais usual de avaliar a precisão é através de comparação visual. Mesmo utilizando imagens reais, essa forma de avaliação mostra-se simples e eficiente. No entanto, quando trabalha-se com volumes de imagens, a comparação visual pode ser ineficiente. Neste caso, é necessário realizar cortes transversais das superfícies geradas e projetar esses cortes sobre as imagens originais.

Para a análise da precisão do método desenvolvido, foi gerado um volume de imagens sintéticas compostas por um elipsóide e uma esfera, e calculou-se a precisão através da distância entre os pontos obtidos e os pontos reais dos objetos na imagem.

Baseando-se em uma escala métrica definida em *pixel* e *voxel*, o volume de dados criado foi de 150 x 150 x 150, contendo uma esfera de raio 30 e um elipsóide de eixos 45, 60 e 30. Os parâmetros utilizados na T-Superfície foram *grid* 5 x 5 x 5, $c = 0.65$, $k = 1.32$, $\gamma = 0.01$, *ponto de congelamento* = 10 e $T \in (145, 155)$. O resultado final foi atingido após 17 deformações. Calculou-se a média do erro e o desvio padrão para cada um dos casos, como pode-se observar na Tabela 5.1.

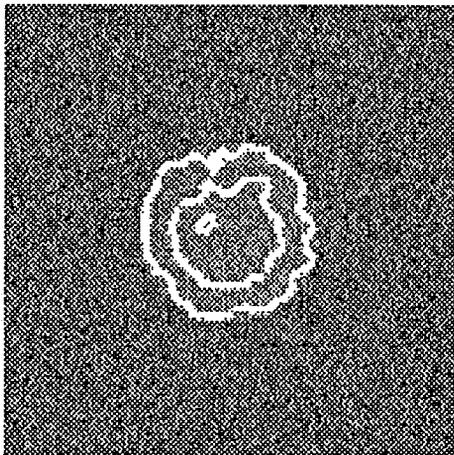
Tabela 5.1: Análise da precisão do método com imagens sintéticas.

	Esfera	Elipsóide
Pontos Analisados	13.080	28.500
Média do Erro para grid 5x5x5	0.933694	1.033619
Desvio Padrão do Erro para grid 5x5x5	0.755183	0.672980

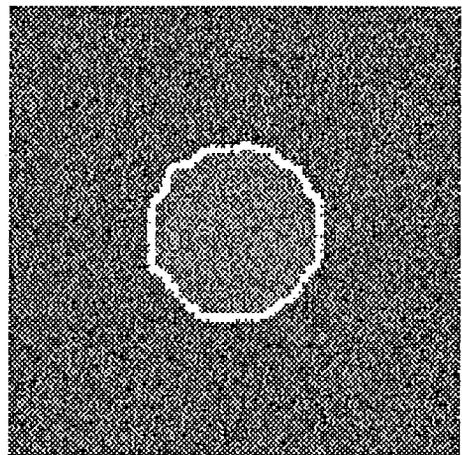
Numa segunda etapa, testou-se a precisão do método com o mesmo volume de dados imerso em um ruído uniforme especificado por uma intensidade de imagem variando de 0 a 150. Observando-se a Tabela 5.2, nota-se que mesmo nessas condições, a média do erro e o desvio padrão foram aceitáveis em relação ao *grid* especificado. A Figura 5.1 mostra cortes transversais das esferas imersas no ruído e as projeções da T-Superfície em sua fase inicial (Figura 5.1.a) e final (Figura 5.1.b).

Tabela 5.2: Análise da precisão do método com imagens sintéticas com ruído.

	Esfera	Elipsóide
Pontos Analisados	10677	24351
Média do Erro para grid 5x5x5	1.581309	1.996275
Desvio Padrão do Erro para grid 5x5x5	2.490322	2.3842078



(a)



(b)

Figura 5.1: Corte transversal das esferas imersas no ruído (a) Inicialização da T-Surface. (b) Resultado Final.

5.2 Comparação entre os métodos de Inicialização

O método de inicialização desenvolvido possui diversas vantagens em relação à metodologia tradicional. Será apresentado a seguir um comparativo entre os métodos de inicialização: (1) Superfície inicial descrita por uma semente de topologia esférica; (2) Análise do *threshold*.

A Figura 5.2 demonstra a eficiência do método em relação ao número de iterações necessárias para segmentarmos o objeto de interesse. Nesse exemplo, foi utilizado um volume de dados de dimensão $128 \times 128 \times 25$ e a superfície final, para ambos os casos, possuía aproximadamente 7000 elementos triangulares. Os dados utilizados na T-Superfície foram *grid* $6 \times 6 \times 2$, $c = 0.62$, $k = 1.45$, $\gamma = 0.05$, *ponto de congelamento* = 5 e $T \in (45, 55)$. Note que no primeiro caso, a esfera inicial (semente) teve o seu centro posicionado nas coordenadas (10,12,10), sendo o raio $r = 2$. Já na inicialização baseada no *threshold*, o resultado pôde ser obtido em menos tempo com um número menor de iterações, como mostra a Tabela 5.3.

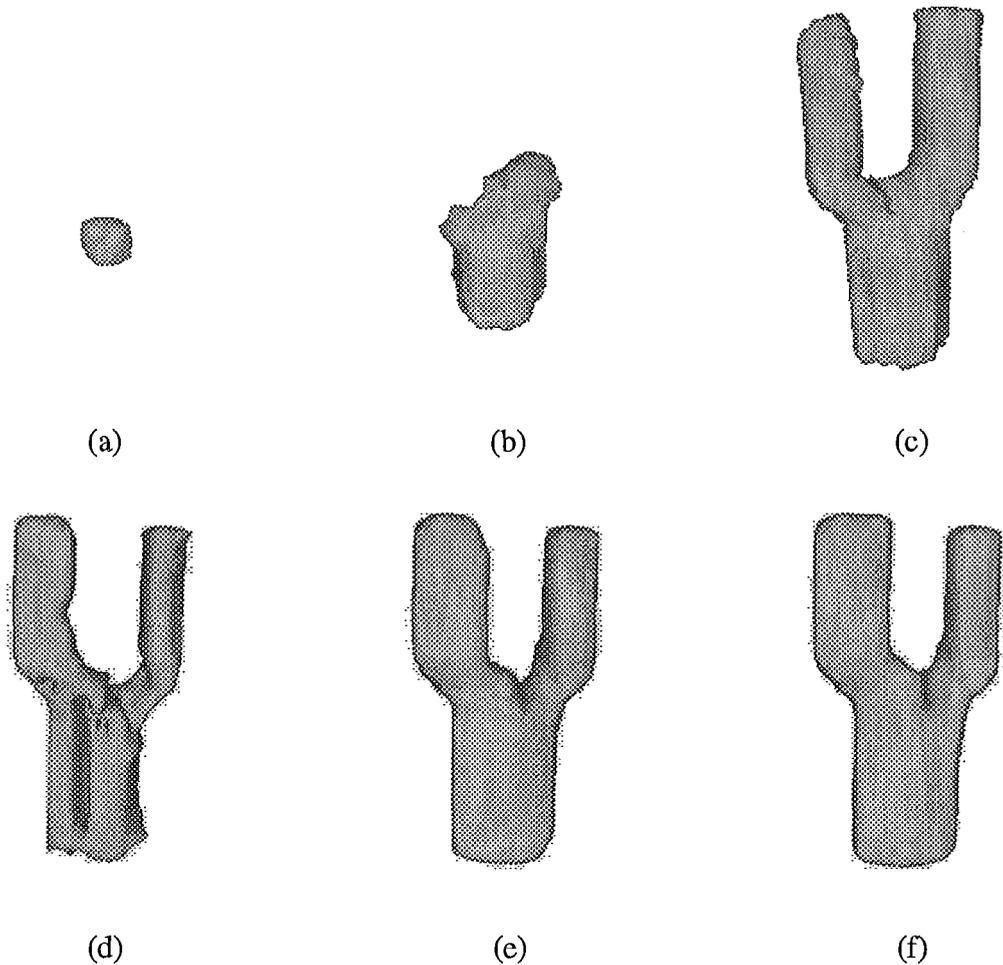


Figura 5.2: Comparativo entre os métodos de inicialização (a) Primeira evolução. (b) Iteração 32. (c) Iteração 49 (final). (d) Inicialização por *threshold*. (e) Iteração 6. (f) Iteração 9 (final).

Tabela 5.3: Análise comparativa da eficiência dos métodos de inicialização.

	Esfera	Threshold
Iterações	49	9
Tempo	9' 12"	33"

5.3 Aplicações da Filtragem por Difusão Anisotrópica

Quando trabalha-se com volume de imagens com ruído, o pré-processamento torna as intensidades de cinza das imagens mais homogêneas, facilitando a escolha do *threshold* que será utilizado como parâmetro para a inicialização do método das T-Superfícies.

Como visto na Seção 4.3, o método de filtragem através de difusão anisotrópica possibilita “borrar” pequenas discontinuidades bem como realçar as bordas, auxiliando desta forma no reconhecimento dos padrões de cinza que serão utilizados no método das T-Superfícies.

A Figura 5.3 apresenta um caso onde a T-Superfície não consegue segmentar o objeto de interesse de forma eficiente. Após efetuarmos a filtragem por difusão anisotrópica, as regiões onde a segmentação falhou foram corrigidas.

As imagens utilizadas nesse exemplo são do projeto “Homem Visível”. A artéria reconstruída foi extraída de um volume de imagens de $155 \times 170 \times 165$. Os parâmetros da T-Superfície foram $grid\ 4 \times 4 \times 4$, $c = 0.75$, $k = 1.12$, $\gamma = 0.35$, $ponto\ de\ congelamento = 10$ e $T \in (14, 26)$. O resultado foi obtido após 18 deformações, gerando uma superfície final com aproximadamente 13000 elementos triangulares. O tempo necessário para concluirmos a segmentação foi de 9 minutos e 37 segundos.

5.4 Multi-resolução

O método de multi-resolução desenvolvido nos dá a possibilidade de, quando necessário, usar uma malha de resolução mais grossa para obter uma primeira aproximação do objeto e, em seguida, refinar a malha nos locais onde há tetraedros transversos e melhorar a solução até atingir um resultado satisfatório.

A multi-resolução aplicada ao método de T-Superfícies foi útil para o caso ilustrado na Figura 5.4. Neste exemplo, inicializamos a T-Superfície num *grid* de resolução $3 \times 3 \times 3$. Como a artéria a ser extraída afinava em suas extremidades e a separação entre os ramos, em certas regiões, era mais fina que a resolução inicial, o resultado foi pouco satisfatório. Nesse caso, continuou-se o processo de evolução da T-Superfície com uma resolução de $1 \times 1 \times 1$ para atingir o resultado desejado.

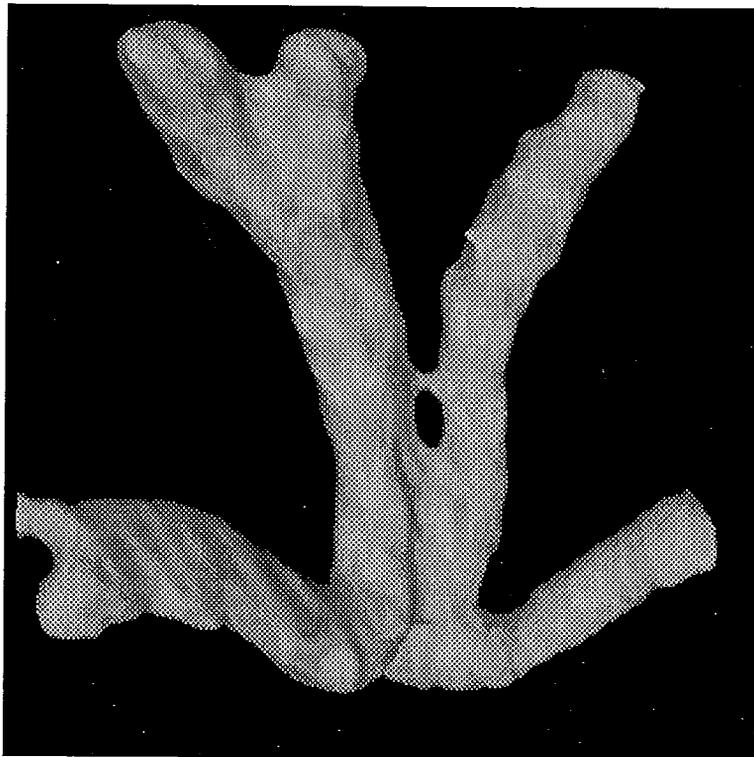
Os parâmetros das forças da T-Superfície utilizados nesse exemplo foram $c = 0.6905$, $k = 1.222$, $\gamma = 0.505$, $ponto\ de\ congelamento = 5$ e $T \in (28, 32)$ num volume de imagens de $86 \times 72 \times 100$. O resultado final foi obtido após 13 deformações num tempo de 1 minuto e 56 segundos.

5.5 Geração de *offset*

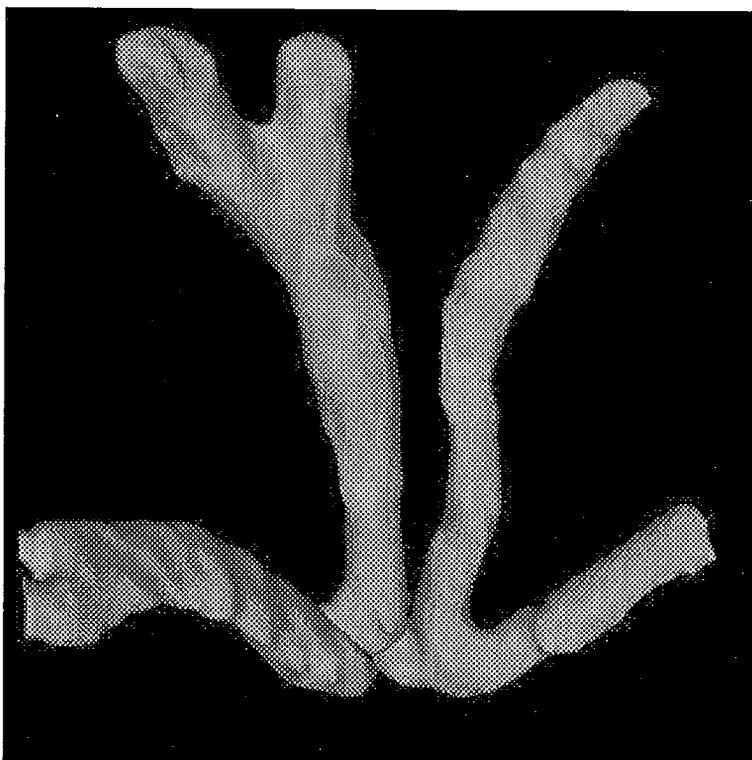
Na seção 4.5 discutiu-se a maneira como podemos gerar *offsets* a partir do modelo das T-Superfícies.

Testou-se o método para um volume de imagens de $155 \times 170 \times 165$ obtidos do projeto “Homem Visível”. Os parâmetros da T-Superfície foram $grid\ 4 \times 4 \times 4$, $c = 0.75$, $k = 1.12$, $\gamma = 0.300$, $ponto\ de\ congelamento = 10$ e $T \in (14, 26)$.

A Figura 5.5.b mostra o *5-offset* da superfície inicial. Observa-se que a topologia da superfície inicial foi preservada. Para reconstrução de geometrias em imagens 2D, o processo de geração de *offsets* tem sido usado no contexto de Modelos Duais de Contornos Ativos (Giraldi *et al.* , 2001). Para inicializar esse modelo, são necessários dois contornos, um interior à superfície e outro exterior. Através da expansão do contorno interior e da contração do contorno externo, reconstrói-se as paredes internas e externas de um objeto de interesse.

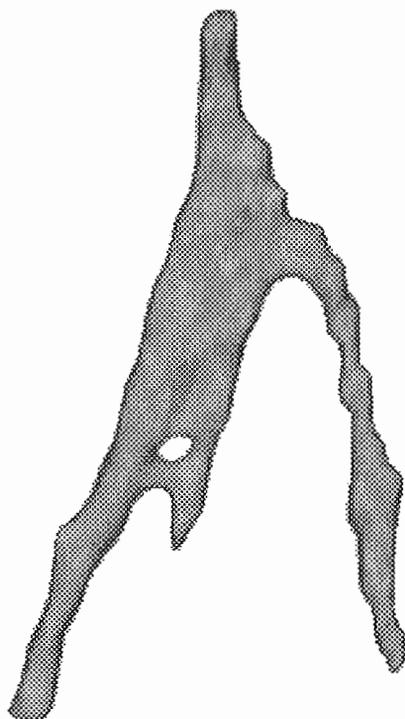


(a)

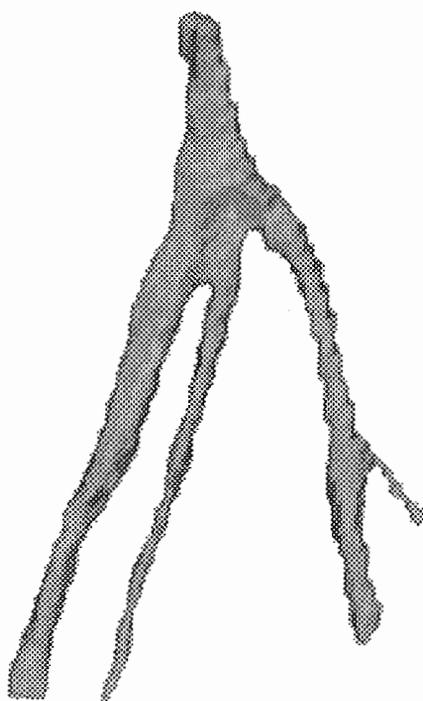


(b)

Figura 5.3: Aplicação da filtragem por difusão anisotrópica (a) Segmentação da artéria com volume de dados original (b) Segmentação após filtragem.

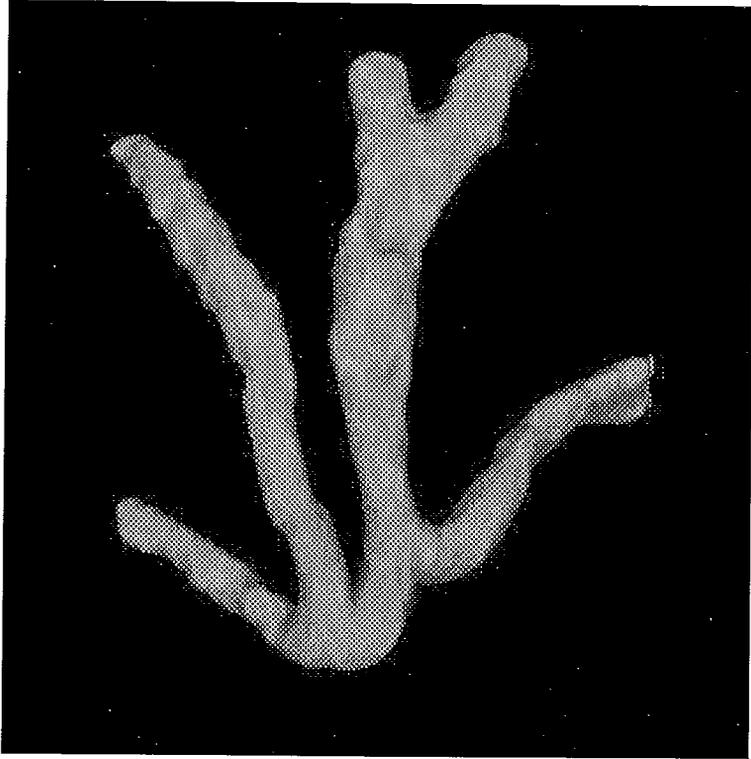


(a)

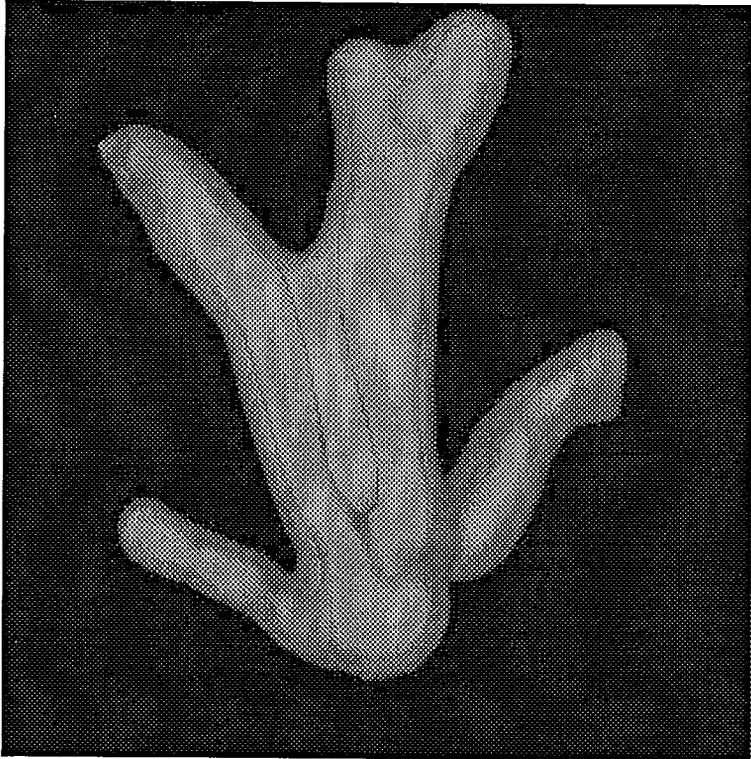


(b)

Figura 5.4: Aplicação da multi-resolução (a) Artéria inicialmente segmentada na resolução $3 \times 3 \times 3$ (b) Segmentação final na resolução $1 \times 1 \times 1$.



(a)



(b)

Figura 5.5: Geração de *Offsets* (a) Artéria Inicial (b) Artéria com *offset*.

Capítulo 6

Conclusões e Trabalhos Futuros

O enfoque deste trabalho foi no estudo de técnicas de multi-resolução aplicadas ao modelo de T-Superfícies com a finalidade de reconstruir superfícies de forma mais eficiente.

O “Algoritmo de Continuação 3D” (Seção 3.2.4) utilizado para reconstruir a superfície inicial compartilha os mesmos elementos utilizados em métodos de geração de iso-superfícies (Lorenson & Cline, 1987). Mas esses métodos, em geral, não incorporam as restrições de escala e topologia necessárias para identificar as estruturas de interesse. As habilidades topológicas das T-Superfícies possibilitam este tipo de identificação, bem como procedimentos interativos que possibilitam mudar a topologia de uma superfície.

A metodologia implementada preserva a topologia esférica da superfície gerada sem a necessidade de testes adicionais. Foi proposto a geração de *offsets* à partir da superfície final obtida com o método da T-Superfície e indicado suas possíveis utilizações.

No método das T-Superfícies, as imagens a serem utilizadas necessitam, na maioria das vezes, de uma etapa de pré-processamento. Foi proposta a utilização de métodos de filtragem por difusão anisotrópica, indicando suas vantagens em relação a outros métodos de filtragem. Foi apresentado um resultado, obtido com a utilização deste filtro.

O método das T-Superfícies foi escolhido para esse trabalho por ser um método simples e robusto. Outros métodos poderiam ter sido utilizados como será visto nas próximas seções.

6.1 Comparação com sistema dinâmico de partículas

Assim como o sistema dinâmico de partículas proposto por Szeliski (Székely *et al.*, 1996), as T-Superfícies podem ser entendidas como um sistema de partículas. Estes dois modelos compartilham outras características. Ambos utilizam um sistema de forças entre as partículas para manter a suavidade da superfície, assim como forças de imagem para atrair as partículas

até as áreas de interesse.

A T-Superfície utiliza seu *grid* simplicial para automaticamente inserir partículas (elementos do modelo) e realizar mudanças topológicas, sendo a conectividade das partículas mantida explicitamente durante a evolução do modelo. A técnica de Szeliski não faz uso de uma triangulação do espaço para inserir novas partículas durante a evolução do modelo. A idéia neste caso é adicionar partículas em regiões onde for considerado necessário (regiões de alta curvatura, por exemplo).

Por ter características de modelos implícitos e paramétricos, o modelo de T-Superfície tem como vantagem a possibilidade de poder utilizar restrições de topologia e forma para auxiliar na obtenção dos objetos de interesse. Uma vantagem do modelo proposto por Szeliski é a sua facilidade de tratar superfícies abertas, bem como superfícies fechadas.

6.2 Comparação com *Level Sets*

O modelo de *Level Sets* proposto por Osher e Sethian (Sethian, 1988) foi adotado por muitos pesquisadores para resolver o problema de reconstrução de superfícies. Nesta formulação, o modelo é definido pela interseção de uma função $w = F(XYZ, T)$ com o plano $w = 0$. A evolução da superfície é determinada por uma equação diferencial parcial, e não pela minimização de um funcional de energia.

Por ser um modelo de formulação estritamente implícita e, desta forma, necessitar de uma dimensão adicional, este modelo não é tão conveniente quanto um modelo paramétrico para possibilitar interação com usuário ou incorporar informações conhecidas à priori sobre a geometria dos objetos. A formulação da T-Superfície é mais simples pois envolve apenas um sistema dinâmico. O algoritmo é composto basicamente por operações algébricas e algumas estruturas de dados simples, como listas e tabelas de hash.

É importante ressaltar que, até onde conhecemos da literatura, não existe um trabalho com um comparação extensiva entre o método das T-Superfícies e *Level-Sets*.

6.3 Trabalhos futuros

Uma possibilidade de estendermos esse trabalho é através da generalização do método de interação com o usuário apresentado. Essa generalização seria obtida através da substituição do plano de corte por um bisturi virtual, permitindo ao usuário interagir de forma mais eficiente com a superfície.

Com relação a geração de *offsets*, uma possível aplicação seria na inicialização de T-Superfícies Duais (*Dual-T-Surfaces*). Neste caso, necessita-se de uma superfície interna, e uma superfície externa ao objeto. Estas superfícies seriam obtidas através de *offsets* da superfície inicial obtida pela método desenvolvido nesse trabalho.

Apêndice A

T-Surface Builder

Esse software, juntamente com as técnicas implementadas nesse trabalho, foi desenvolvido sob a co-orientação do Professor Gilson Antonio Giraldi, do LNCC (Laboratório Nacional de Computação Científica), no contexto do Projeto *Modelagem e Simulação Computacional do Sistema Cardiovascular Humano* daquele instituto.

Este projeto é constituído basicamente de 3 etapas: (1) Extração de geometrias de artérias; 2) Simulação numérica do fluxo sanguíneo; (3) Visualização dos dados gerados na simulação.

A principal finalidade do *T-Surface Builder* é na reconstrução de geometrias de artérias. Por outro lado, esse sistema pode auxiliar usuários com pouca experiência em modelos deformáveis na reconstrução de superfícies a partir de um volume de imagens, baseada em poucas etapas operacionais.

Basicamente, o usuário visualiza as imagens que formarão o volume de dados, verifica qual será o *threshold* a ser utilizado, completa a janela de parâmetros da T-Superfície, indica a forma como quer inicializar a T-Superfície, e inicia o processo. Caso o resultado não seja satisfatório, o usuário tem a opção de continuar o processamento a partir das superfícies geradas.

As seções a seguir apresentam as principais funções do aplicativo: Visualizador de Superfícies, Visualizador de Imagens e o Construtor de Superfícies.

A.1 Visualizador de Superfícies

Ao executar o programa com um dado volume de imagens, as superfícies intermediárias geradas durante o processo de deformação da T-Superfície não são descartadas. Todas essas superfícies são armazenadas para posterior avaliação. A opção de visualizar superfícies, nos permite carregar esses arquivos armazenados apenas para visualização (Figura A.1)

As superfícies armazenadas podem ser rotacionadas, transladadas, aproximadas e visuali-

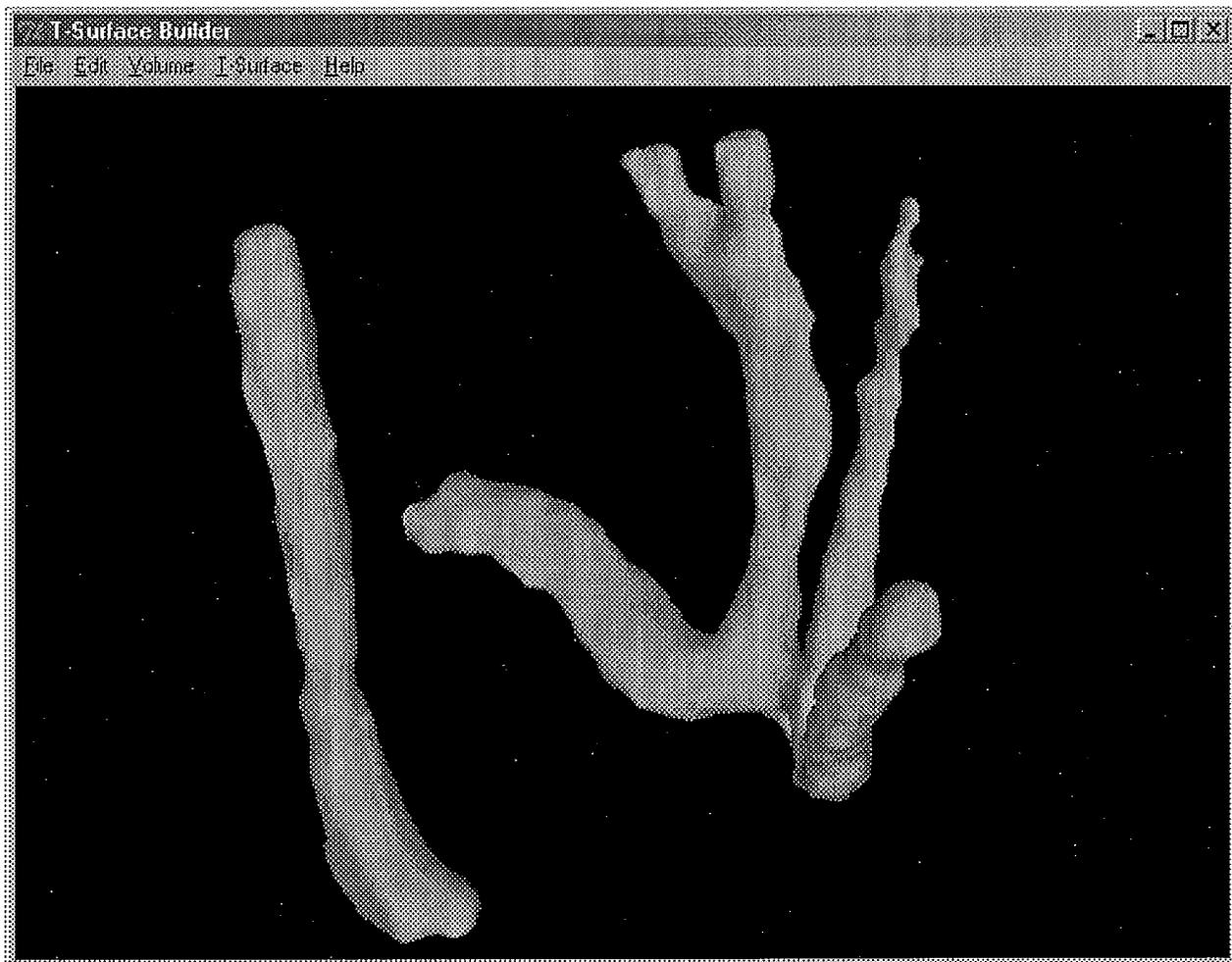


Figura A.1: Visualizador de superfícies apresentando duas artérias.

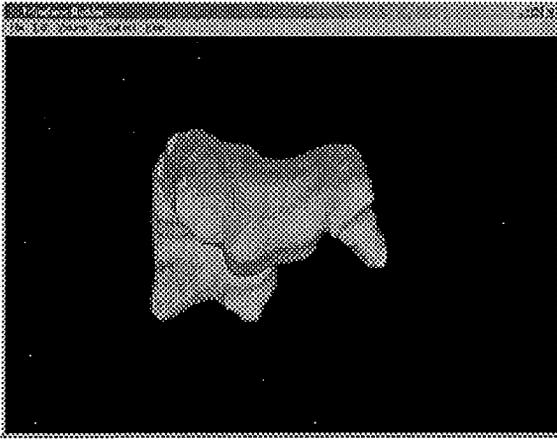
zadas como *wireframe* (Figura A.2). Pode-se, por exemplo, analisar o interior da superfície, como mostra a Figura A.3.

A.2 Visualizador de Imagens

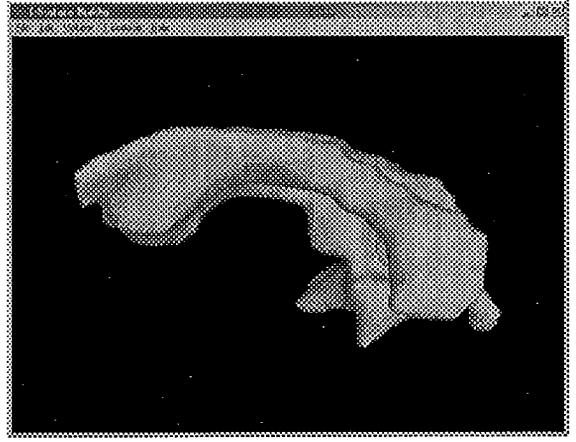
Antes de iniciarmos a seleção dos parâmetros da T-Superfície, é necessário verificar qual será o *threshold* a ser utilizado, ou mesmo o local onde será colocada a superfície inicial de topologia esférica, no caso de inicialização tradicional.

Para isso, introduziu-se no aplicativo um visualizador de imagens que possibilita vê-las, *slice* por *slice*, facilitando até mesmo a identificação do formato das estruturas antes mesmo de reconstruir a superfície (Figura A.4).

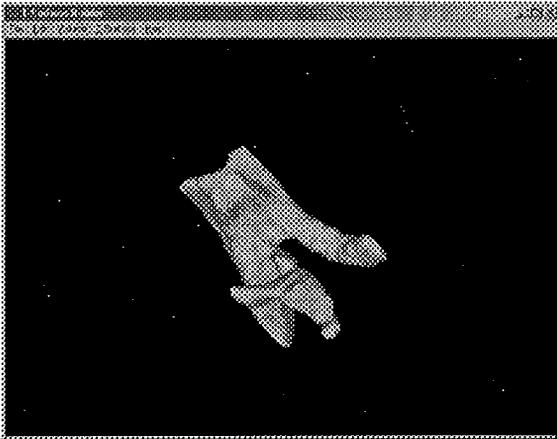
Pode-se recortar manualmente um bloco de imagens em uma certa região, caso a estrutura a ser reconstruída esteja confinada em alguma área do volume de imagens. Com esse processo, pode-se inicializar a T-Superfície com um volume de imagens menor, sem perder os objetos de



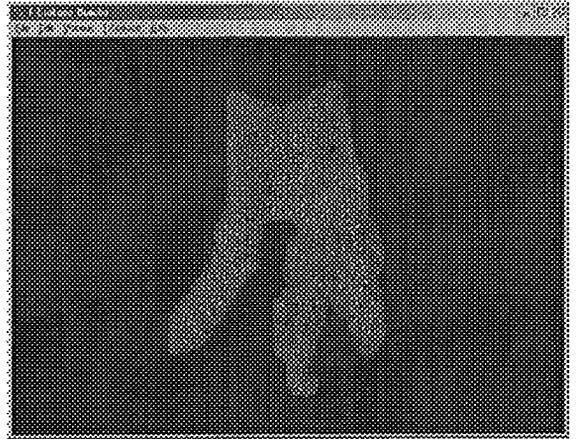
(a)



(b)



(c)



(d)

Figura A.2: Visualizador de superfícies. (a) Objeto visto de frente, (b) Objeto rotacionado, (c) Objeto afastado e (d) *Wireframe*.

interesse.

Outra possibilidade é a aplicação de filtragem por difusão anisotrópica a um grupo de imagens selecionadas. Deve-se indicar um nome do arquivo de saída diferente do nome do arquivo de entrada para que, desta forma, as imagens originais permaneçam sem alteração.

A.3 Construtor de Superfícies

Após verificar o volume de imagens e efetuar, se necessário, seu pré-processamento, pode-se ajustar os parâmetros da T-Superfície que irá reconstruir os objetos de interesse e iniciar o processamento. Este processamento só será interrompido se atingirmos o ponto de congelamento para todos os snaxels que compõem a T-Superfície ou se atingirmos o número de iterações determinado pelo usuário.

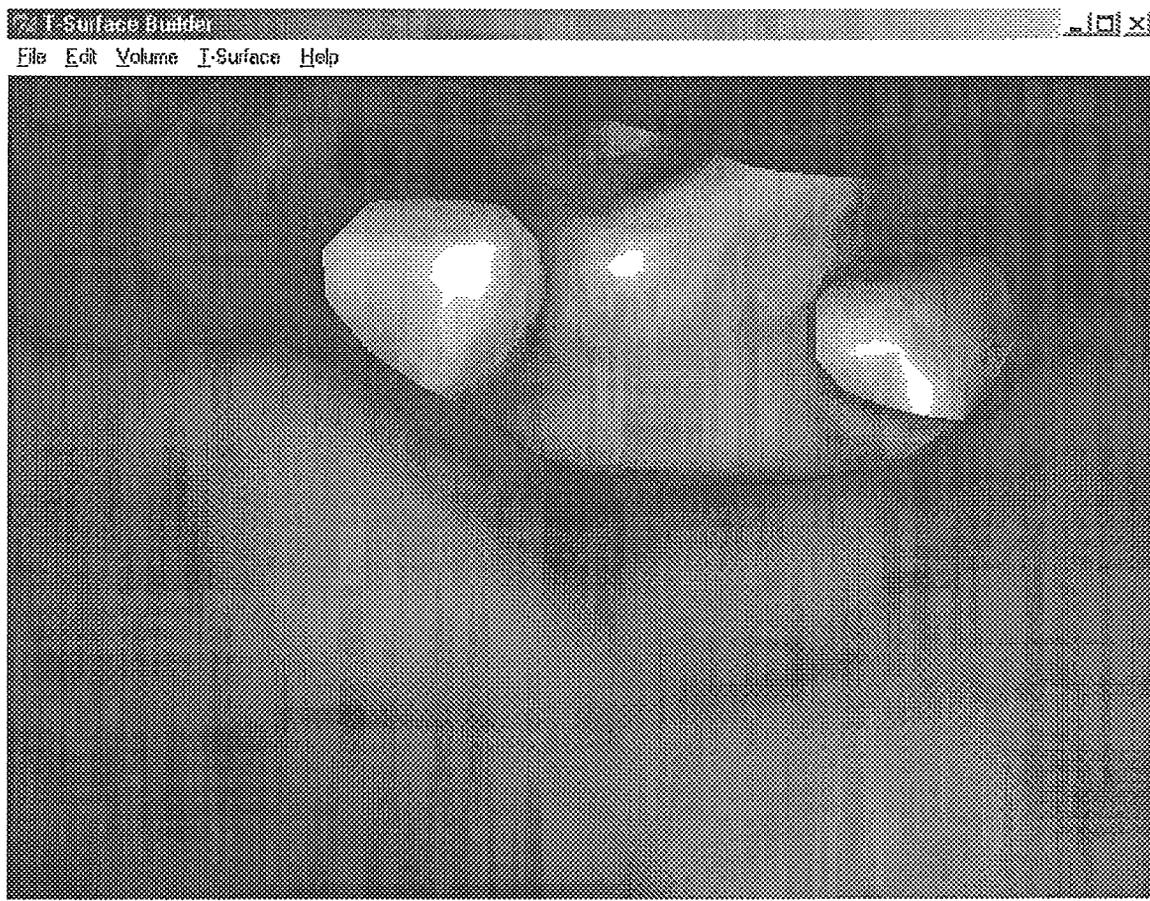


Figura A.3: Visualizador de superfícies sendo utilizado para mostrar o interior de um objeto.

A.3.1 Parâmetros da T-Superfície

A seleção dos parâmetros da T-Superfície pode ser complicada para usuários que nunca trabalharam com esse modelo deformável. Para facilitar esse processo de escolha de parâmetros, foi adicionado ao programa algumas opções *default*. Com os parâmetros previamente ajustados, o usuário terá apenas que acertar o conjunto de imagens que deseja utilizar no processo, a resolução da malha e o valor do *threshold* a ser utilizado, obtido anteriormente ao verificar as imagens (Figura A.5).

Durante a execução do programa, pode-se acompanhar o processo através de um painel que mostra informações de cada iteração que está sendo realizada, número de triângulos processados nessa iteração etc (Figura A.6).

Depois de ajustados e testados, os parâmetros podem ser salvos e utilizados em outros volumes de imagens com características semelhantes. Essa característica do programa facilita na segmentação de múltiplas componentes em um mesmo volume de dados.

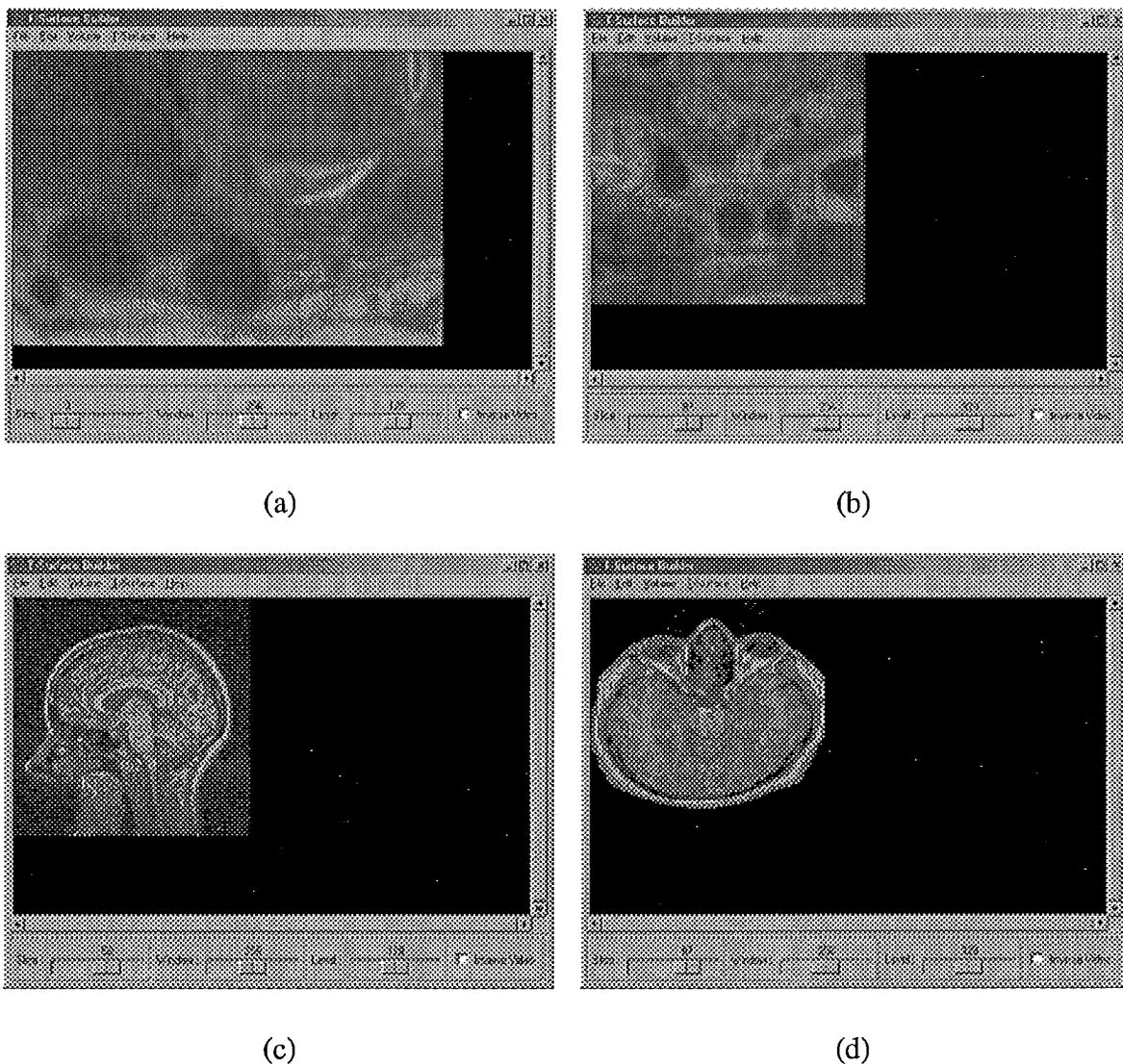


Figura A.4: Visualizador de imagens. (a) e (b) Artérias (*Visible Man*) (c) e (d) Interior da cabeça (MRI).

A.3.2 Métodos de Inicialização

Como já mencionado anteriormente, o sistema oferece diversas formas de inicialização do modelo de T-Superfícies: (1) usando-se um objeto de topologia esférica totalmente interior ou exterior ao objeto de interesse; (2) Utilizando superfície gerada através do *threshold* e (3) Utilizando superfícies intermediárias, geradas durante a evolução da superfície inicial.

Objeto com Topologia Esférica

No programa desenvolvido, foi escolhida a esfera como semente para inicializarmos a T-Superfície na forma tradicional. Deve-se indicar o raio e a posição em relação ao volume de dados da esfera (Figura A.7).

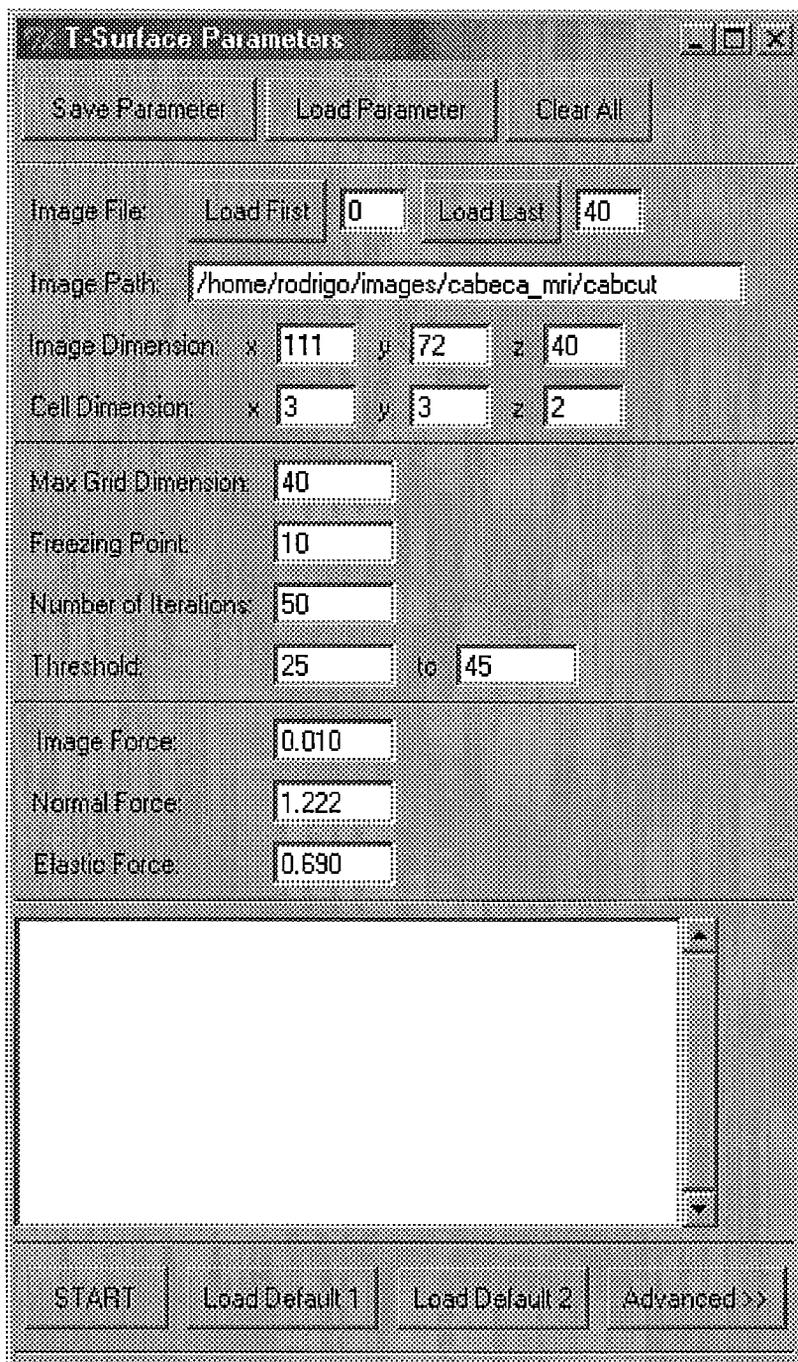


Figura A.5: Janela de parâmetro para inicializar processo segmentação.

Threshold

Ao inicializarmos a T-Superfície através de um *threshold* da imagem, foi dada a opção de escolhermos uma ou várias componentes. Esse opção é interessante quando o *threshold* do objeto de interesse está presente em diversas áreas do volume de imagem que não interessam. Para indicar a localização do objeto de interesse, deve-se informar a coordenada de um ponto interior ao objeto. Esse ponto pode ser obtido através do visualizador de imagens (Figura A.8).

Se for selecionada a opção de múltiplas componentes, todos os objetos que possuem *th-*

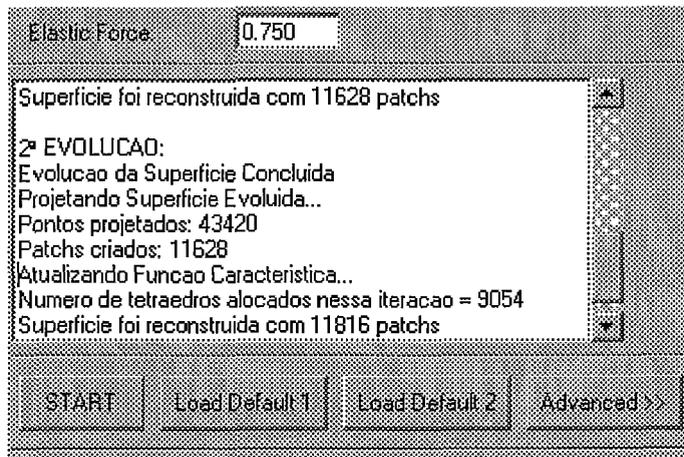


Figura A.6: Informações sendo mostradas durante execução da T-Superfície.

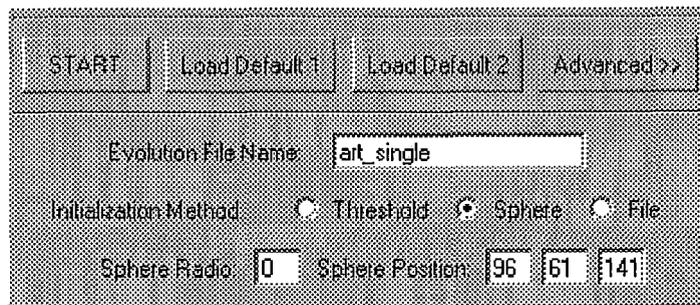


Figura A.7: Inicialização através de objeto esférico.

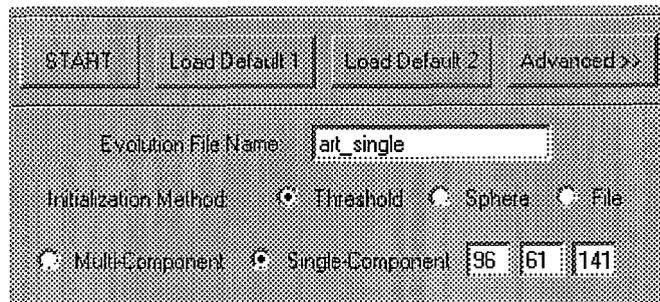


Figura A.8: Inicialização através de *threshold*.

reshold válido serão computados. Nesse caso, utilizando multi-resolução, pode-se descartar objetos de pequena dimensão em relação ao *grid* inicial, e aumentar a resolução conforme a necessidade.

Superfície Armazenada

A última opção de inicialização é através de uma superfície previamente armazenada. Nesse caso, deve-se apenas indicar o nome do arquivo a ser utilizado (Figura A.9).

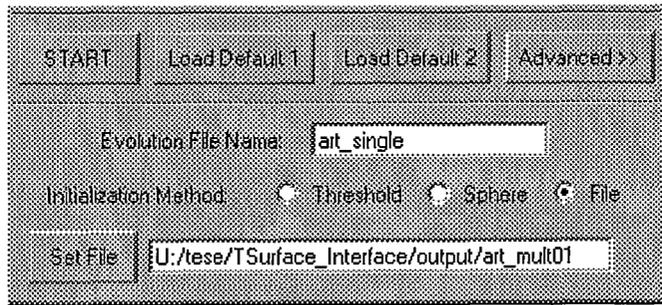


Figura A.9: Inicialização através de arquivo.

A.4 Informações Técnicas

A interface gráfica do programa foi desenvolvida em TCL/TK. O processamento é feito em Linguagem C, fazendo uso da biblioteca gráfica VTK. O programa é portátil, podendo ser executado tanto em ambiente Windows quando em ambiente LINUX, sem necessidade de alteração de código.

Todos os testes foram realizados em máquinas Intel Pentium III 866MHz com 512 MB de memória RAM.

Foi observado um nítido ganho de performance quando o programa foi executado em ambiente LINUX.

Referências Bibliográficas

- Adams, A. & Bischof, L. 1994. Seeded Region Growing. *IEEE Trans. on Pattern Analysis and Machine Intel.*, **16**(6).
- Allgower, E. L. & Georg, K. 1990. *Numerical Continuation Methods: An Introduction*. Springer-Verlag Berlin Heidelberg.
- Berger, M. J. 1986. Data Structures for Adaptive Grid Generation. *SIAM J. Sci. Stat. Comput.*, **7**, 904–916.
- Black, A. & Yuille, A. (eds). 1993. *Active Vision*. MIT Press.
- Canny, J. F. 1983. *Finding Edges and Lines in Images*. Tech. rept. MIT, <http://www.mit.edu/>.
- Cohen, L. D. 1991. On Active Contour Models and Balloons. *CVGIP:Image Understanding*, **53**(2), 211–218.
- Durikovic, R.; Kaneda, K. & Yamashita, H. 1995. Dynamic contour: a texture approach and contour operations. *The Visual Computer*, **11**, 277–289.
- Giraldi, G. A.; Strauss, E. & Oliveira, A. F. 2000. A Boundary Extraction Approach Based on Multi-resolution Methods and the T-Snakes Framework. *In: International Symposium on Computer Graphics, Image Processing and Vision (SIBGRAPI'2000 (to appear))*.
- Giraldi, G. A.; Strauss, E. & Oliveira, A. F. 2001. Improving the dual-t-snakes model. *In: In International Symposium on Computer Graphics, Image Processing and Vision (SIBGRAPI'2001)*.
- Gunn, S. R. & Nixon, M. S. 1997. A Robust Snake Implementation; A Dual Active Contour. *IEEE Trans. Pattern Anal. Mach. Intell.*, **19**(1), 63–68.
- Jain, A. K. 1989. *Fundamentals of Digital Image Processing*. Prentice-Hall, Inc.

- Jolion, J. M. & Montanvert, A. 1992. The Adaptive Pyramid: A framework for 2D Image Analysis. *CVGIP: Image Understanding*, **55**(3), 339–348.
- Jones, T. N. & Metaxas, D. N. 1998. Image Segmentation Based on the Integration of Pixel Affinity and Deformable Models. *In: Proceedings of CVPR'98*.
- Kass, M.; Witkin, A. & Terzopoulos, D. 1988. Snakes: Active Contour Models. *International Journal of Computer Vision*, **1**(4), 321–331.
- Lorensen, W. E. & Cline, H. E. 1987. Marching Cubes: A High Resolution 3D Surface Construction Algorithm. *Computer Graphics*, **21**(4), 163–169.
- Malladi, R.; Sethian, J. A. & Vemuri, B. C. 1995. Shape Modeling with Front Propagation: A Level Set Approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, **17**(2), 158–175.
- McInerney, T. & Terzopoulos, D. 1995 (June). Topologically Adaptable Snakes. *Pages 840–845 of: Proc. Of the Fifth Int. Conf. On Computer Vision (ICCV'95), Cambridge, MA, USA*.
- McInerney, T. & Terzopoulos, D. 1997 (March). Medical Image Segmentation Using Topological Adaptable Surfaces. *In: Proc. CVRMed'97*.
- McInerney, T. & Terzopoulos, D. 1999. Topology Adaptive Deformable Surfaces for Medical Image Volume Segmentation. *IEEE Trans. on Medical Imaging*, **18**(10), 840–850.
- McInerney, T. J. 1997. *Topologically Adaptable Deformable Models for Medical Image Analysis*. Ph.D. thesis, Department of Computer Science, University of Toronto.
- Mumford, D. & Shah, J. 1989. Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems. *Comm. Pure Appl. Math.*, **42**, 577–684.
- Perona, P. & Malik, J. 1990. Scale-Space and Edge Detection Using Anisotropic Diffusion. *IEEE Trans. on Patter Analysis and Mach. Intell.*, **12**(7), 629–639.
- Poggio, T.; Torre, Vincent & Koch, Christof. 1985. Computational Vision and regularization theory. *Nature*, **317**(26), 314–319.
- Rogers, David F. 1985. *Procedural Elements for Computer Graphics*. MacGraw-Hill Book Company.
- Sethian, S. Osher J. A. 1988. Fronts Propagation with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations. *Journal of Computational Physics*, **79**, 12–49.

- Snyder, M. A. 1991. On the Mathematical Foundations of Smoothness Constraints for the Determination of Optical Flow and for Surface Reconstruction. *IEEE Trans. on Pattern Analysis and Mach. Intell.*, **13**(11), 1105–1114.
- Székely, G.; Kelemen, A.; Brechbuhler, Ch. & Gerig, G. 1996. Segmentation of 2-D and 3-D objects from MRI volume data using constrained elastic deformations of flexible Fourier surface models. *Pages 82–87 of: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR-93)*.
- Vasilescu, M. & Terzopoulos, D. 1992. Adaptive Meshes and Shells: Irregular Triangulation, Discontinuities, and Hierarchical Subdivision. *Pages 829–832 of: Proc. IEEE Conf. Computer Vision and Pattern Recognition, CVPR*. Los Alamitos, California: IEEE Press.
- Xu, Gang; Segawa, E. & Tsuji, S. 1994. Robust Active Contours With Insensitive Parameters. *Pattern Recognition*, **27**(7), 879–884.
- Zhu, S. C. & Yuille, A. 1996. Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multibang Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, **18**(9), 884–900.