

SNAKES IMPLÍCITAS - APLICAÇÃO A IMAGENS OBTIDAS POR
ULTRA-SONOGRAFIA

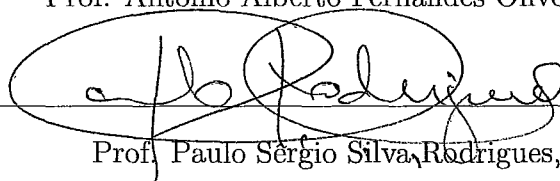
Diego Mazala Carvalho Rodrigues

DISSERTAÇÃO 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.

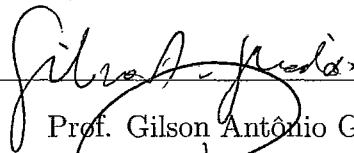
Aprovada por:



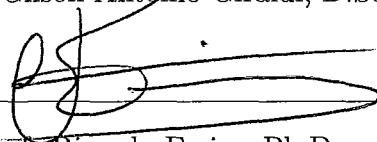
Prof. Antônio Alberto Fernandes Oliveira, D.Sc.



Prof. Paulo Sérgio Silva Rodrigues, D.Sc.



Prof. Gilson Antônio Giraldo, D.Sc.



Prof. Ricardo Farias, Ph.D.

RIO DE JANEIRO, RJ - BRASIL

JUNHO DE 2008

RODRIGUES, DIEGO MAZALA
CARVALHO

Snakes Implícitas - Aplicação a Imagens Obtidas por Ultra-sonografia [Rio de Janeiro] 2008

XIII, 94 p. 29,7cm (COPPE/UFRJ, M.Sc., Engenharia de Sistemas e Computação, 2008)

Dissertação – Universidade Federal do Rio de Janeiro, COPPE

1. Snakes Implícitas
2. Segmentação de Imagens
3. Level Sets

I. COPPE/UFRJ II. Título (série)

Ao meu pai, que se foi durante o curso deste trabalho e não teve tempo de participar da conclusão de mais esta etapa em minha vida.

Agradecimentos

Sei que serei incapaz de registrar meus agradecimentos a todos aqueles que de alguma forma contribuíram para a realização deste trabalho, mas não serei negligente ao ponto de desperdiçar esta oportunidade e pelo menos tentar prestar minha gratidão a quem é devida.

Primeiramente, quero agradecer à minha família. A cada um daqueles que jamais mediram esforços para me ajudar a caminhar. Em especial a minha mãe, minha amiga e conselheira. Estou certo de que suas orações e seu impecável exemplo sempre me sustentaram e ainda o fazem até hoje. Agradeço sua ajuda incondicional e peço desculpas pela ausência tão freqüente. Hoje, mesmo distante, a senhora é e continuará sendo meu grande exemplo de vida. Muito obrigado, mãe.

A minha amável esposa, por todo o apoio e compreensão na minha ausência. Você conseguiu namorar, noivar e casar comigo durante o mestrado. Incrível! Obrigado por me permitir dividir a vida com você. Te amo.

Aos meus amigos(irmãos) Algemiro e Bruno. Nós sabemos o quanto lutamos para que o mestrado fosse um sonho alcançável. Ainda estão na memória cada uma daquelas nossas conversas de apoio e incentivo mútuo. Devo muito a vocês. Obrigado pela oportunidade de trabalhar e estudar com vocês.

Ao Prof. Paulo Sérgio, por um dia ter acreditado em mim e me permitir ser seu aluno de iniciação científica, onde tudo começou. Agradeço por cada uma das aulas ministradas, pelos conselhos, pelos momentos de lazer, e por todo o tempo investido para que hoje eu pudesse escrever este texto.

Ao Prof. Antônio Oliveira, por toda atenção e inesgotável paciência explicando aquilo que lhe parecia tão trivial e que meu intelecto insistia em não compreender. Obrigado por me aceitar como aluno e me orientar durante todo este período.

Aos colegas de mestrado, pela agradável convivência, por todos os almoços e

cafezinhos que desfrutamos juntos, e por todo o aprendizado que pude obter com vocês.

Aos demais professores com quem convivi nos laboratórios e nas salas de aula, por seus exemplos e por todo empenho na ministração das aulas.

Devo um agradecimento, também, àqueles que por qualquer motivo disseram que eu não poderia ou não conseguiria entrar no mestrado, e após estar cursando, que não concluiria o curso. Agradeço pelas duras palavras mas que foram essenciais para me motivar e fortalecer nos momentos de maior dificuldade.

E por último, mas sem dúvida o mais importante, ao meu Senhor e Salvador Jesus Cristo. A quem plantou em meu coração este sonho e que foi responsável pela realização de mais este milagre. Senhor, estou certo de que não só minha carreira profissional como toda a minha vida é fruto da tua misericórdia e do teu amor para comigo. Se há alguma honra ou glória neste momento, que seja para Ti, o responsável pela conclusão de mais esta etapa em minha vida. Obrigado por realizar mais este sonho. Te amo, Senhor.

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

SNAKES IMPLÍCITAS - APLICAÇÃO A IMAGENS OBTIDAS POR ULTRA-SONOGRAFIA

Diego Mazala Carvalho Rodrigues

Junho/2008

Orientador: Antônio Alberto Fernandes Oliveira

Programa: Engenharia de Sistemas e Computação

Nesta dissertação é apresentado um estudo sobre os métodos de *snakes* implícitas aplicados à segmentação de imagens médicas, mais especificamente àquelas obtidas por ultra-sonografia. Primeiramente, é descrita a metodologia dos *level sets*, sobre a qual são formulados os modelos de *snakes* implícitas. Em seguida, são discutidas algumas aplicações envolvendo os *level sets* e, principalmente, sua utilização em segmentação de imagens médicas. Com o objetivo de testar e comparar as principais técnicas de segmentação em ultra-sonografias descritas neste trabalho, foi implementada uma aplicação e submetida à ela uma base de imagens para segmentação. Os resultados foram comparados com a segmentação manual realizada por dois especialistas no assunto.

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

IMPLICIT SNAKES - APPLICATION TO ULTRASOUND IMAGES

Diego Mazala Carvalho Rodrigues

June/2008

Advisor: Antônio Alberto Fernandes Oliveira

Department: Systems Engineering and Computer Science

In this dissertation is presented a study about the implicit snakes methods applied to medical image segmentation, more specifically to ultrasound images. Firstly, it is described the level set methodology, upon which the implicit snakes are based. Then, some applications using level sets and, mainly, its usage on medical image segmentation are discussed. In order to test and compare some of the main techniques in ultrasound segmentation described in this work, it was implemented an application, by which a set of images were segmented. The results were compared against the manual segmentations accomplished by two specialists on this subject.

Sumário

Lista de Figuras	xi
Lista de Tabelas	xiii
1 Introdução	1
1.1 Segmentação de Imagens	1
1.2 Algoritmos para Segmentação	1
1.3 Imagens de Interesse	2
1.4 Proposta	4
2 Level Sets	6
2.1 Idéia Inicial	7
2.2 Conceitos	7
2.3 Movimento	8
2.4 Movimento por Curvatura	9
2.5 Representação	10
2.6 Escolha da função de velocidade	13
2.7 Inicializando a curva	15
2.8 Cálculo da função distância	15
2.9 Vantagens do Level Set	17
2.10 Complexidade	18
2.11 Aplicações	18
3 Contornos Ativos	20
3.1 Snakes	20
3.2 T-Snakes	22

3.3	Variantes dos Modelos Implícitos	23
3.4	Level Sets sem Regularizadores	24
3.4.1	Com critério de parada baseado no gradiente	24
3.4.2	Com critério de parada baseado na resistência da aresta	25
3.4.3	Com critério de parada baseado na minimização da área	26
3.5	Level Sets com Regularizadores	26
3.5.1	Baseados em Clusterização	26
3.5.2	Baseados em Modelos Bayesianos	27
4	Segmentação de Ultra-sonografias	29
4.1	Classificação por nível de intensidades	30
4.2	Classificação baseada na variância da intensidade dos pixels	31
5	Implementação	34
5.1	Algoritmos	34
5.1.1	Entrada e saída	35
5.1.2	Criação da estrutura de dados	35
5.1.3	Atualização da estrutura de dados	37
5.1.4	Obtenção da curva final	37
5.2	Interface Gráfica	38
5.3	Ferramentas	39
5.4	Download	40
6	Resultados	41
7	Conclusão	50
A	Convertendo da formulação paramétrica para a implícita	53
A.1	Energia de Campo	54
A.2	Energia Interna	60
A.3	Energia Total	61
A.4	Evolução da Snake Segundo a Direção de Máximo Decréscimo	62
A.5	Formulação Implícita	64
A.6	Fórmula Geral	67

B Regiões de uma Curva Fechada	69
C Relação entre Curvatura e Divergente	72
Referências Bibliográficas	78

Lista de Figuras

1.1	Exemplo de imagem com uma região de interesse segmentada. Neste caso, a região é um tumor.	2
1.2	Classificação para os algoritmos de modelos deformáveis.	3
1.3	Amostra de algumas ultra-sonografias. Considere, neste exemplo, as regiões mais escuras como sendo de interesse. Note que estas não possuem suas bordas bem definidas.	4
2.1	Exemplos de curvas fechadas simples e não simples.	8
2.2	Nos pontos onde a curvatura é positiva a curva se movimenta para dentro. Onde ela é negativa a curva se move para fora.	10
2.3	Representação da frente de propagação.	11
2.4	Evolução da interface ao longo do tempo.	12
2.5	Representação gráfica da evolução de uma curva e os valores da função na grade.	16
3.1	Classificação dos modelos deformáveis.	24
4.1	Exemplo em que a curva ultrapassa a borda da região de interesse. . .	30
5.1	Fluxograma da implementação dos algoritmos de segmentação.	36
5.2	Interface gráfica para interação entre o usuário e os algoritmos de segmentação de imagens.	39
6.1	Resultados para ultra-sonografia 012.	43
6.2	Resultados para ultra-sonografia 030.	43
6.3	Resultados para ultra-sonografia 040.	44
6.4	Resultados para ultra-sonografia 080.	44

6.5	Resultados para ultra-sonografia 105.	45
6.6	Resultados para ultra-sonografia 130.	45
6.7	Resultados para ultra-sonografia 140.	46
6.8	Resultados para ultra-sonografia 170.	46
A.1	Representação do vetor \bar{N} .	59
B.1	Um curva fechada e suas regiões.	69

Lista de Tabelas

6.1	Medida de erro: Ultra-sonografia 012	43
6.2	Medida de erro: Ultra-sonografia 030	43
6.3	Medida de erro: Ultra-sonografia 040	44
6.4	Medida de erro: Ultra-sonografia 080	44
6.5	Medida de erro: Ultra-sonografia 105	45
6.6	Medida de erro: Ultra-sonografia 130	45
6.7	Medida de erro: Ultra-sonografia 140	46
6.8	Medida de erro: Ultra-sonografia 170	46
6.9	Tabela de parâmetros utilizados para testes da abordagem proposta por Caselles [1]. Ver seção 3.4.1. A primeira coluna é o parâmetro α descrito na Equação 3.6, a segunda é o número de iterações utilizado nos testes mostrados, e a última coluna é o raio da curva inicial em pixels.	47
6.10	Tabela de parâmetros utilizados para testes da abordagem proposta por Mora[2]. Ver seção 4.2. A primeira coluna é o termo multiplicador da curvatura, a segunda é um multiplicador do termo da imagem, a terceira é o número de iterações necessárias para o alcance da borda, e a última coluna é o raio da curva inicial em pixels.	48
6.11	Tabela de parâmetros utilizados para testes da abordagem proposta por Yang[3]. Ver seção 4.1. A primeira coluna é o termo multiplicador da curvatura, a segunda é um multiplicador do termo da imagem, a terceira é o número de iterações necessárias para o alcance da borda, e a última coluna é o raio da curva inicial em pixels.	48

Capítulo 1

Introdução

1.1 Segmentação de Imagens

Segundo Schenke [4], segmentar uma imagem significa particioná-la em regiões não-intersectantes, tal que cada região seja homogênea em relação a um determinado critério. No contexto relacionado às imagens médicas, a segmentação tem como objetivo delinear estruturas anatômicas ou quaisquer outras regiões de interesse, tais como órgãos e artérias [3] (veja Figura 1.1). Esta é uma tarefa importante, já que pode ser usada, em diversos campos da medicina, como por exemplo no acompanhamento de um feto durante a gravidez, na descrição das paredes de uma artéria ao longo de sua extensão, na separação de células em uma dada amostra ou no diagnóstico de um tumor. Logicamente, cada uma destas aplicações deve ser acompanhada de perto por um especialista que dê validade aos resultados obtidos pela segmentação automática.

1.2 Algoritmos para Segmentação

Embora estes algoritmos possuam os mesmos objetivos - identificar e representar de maneira eficiente uma determinada área de interesse em uma imagem - existem inúmeras abordagens e implementações propostas para que tais metas sejam alcançadas.

Dentre estas abordagens, podemos destacar dois princípios básicos sobre os quais classes diferentes de algoritmos se fundamentam: o da descontinuidade e o da simi-

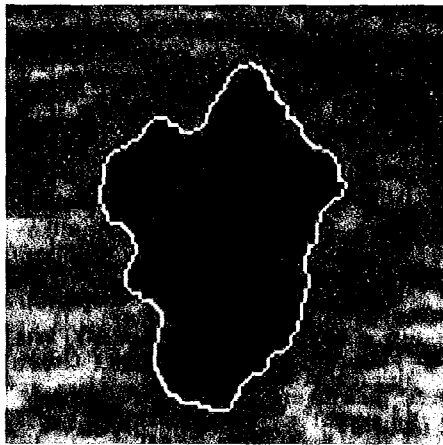


Figura 1.1: Exemplo de imagem com uma região de interesse segmentada. Neste caso, a região é um tumor.

laridade. O primeiro está relacionado à detecção das alterações abruptas nos níveis de intensidades dos pixels da imagem. O segundo conecta os pixels de uma região comum através da similaridade existente entre eles [5]. Embora possamos até fazer uso da similaridade, nosso foco estará sobre a primeira classe de algoritmos, na qual estão os procedimentos básicos empregando modelos deformáveis. Neste contexto, optou-se ainda pelos modelos contínuos com formulação implícita, onde se localizam os algoritmos baseados em curvas de nível (*level sets*). A Figura 1.2 mostra uma classificação para estes algoritmos.

1.3 Imagens de Interesse

A fim de restringir a área de pesquisa abrangida por este trabalho, decidiu-se por estudar a aplicação dos algoritmos de modelos deformáveis na segmentação de imagens médicas. Assim, restaria ainda observar em qual tipo de imagem médica: imagens de raios x, tomografia computadorizada, ressonância magnética, ultra-sonografia, dentre outras. Apesar da aplicabilidade de alguns dos algoritmos estudados em mais de uma destas classes, trabalhou-se principalmente com as imagens geradas por ultra-sonografia.

A ultra-sonografia, também conhecida como eco-grafia, é um método diagnóstico que aproveita o eco produzido pelo som para ver em tempo real as sombras produzidas pelas estruturas e órgãos do organismo. Os aparelhos de ultra-som em geral uti-

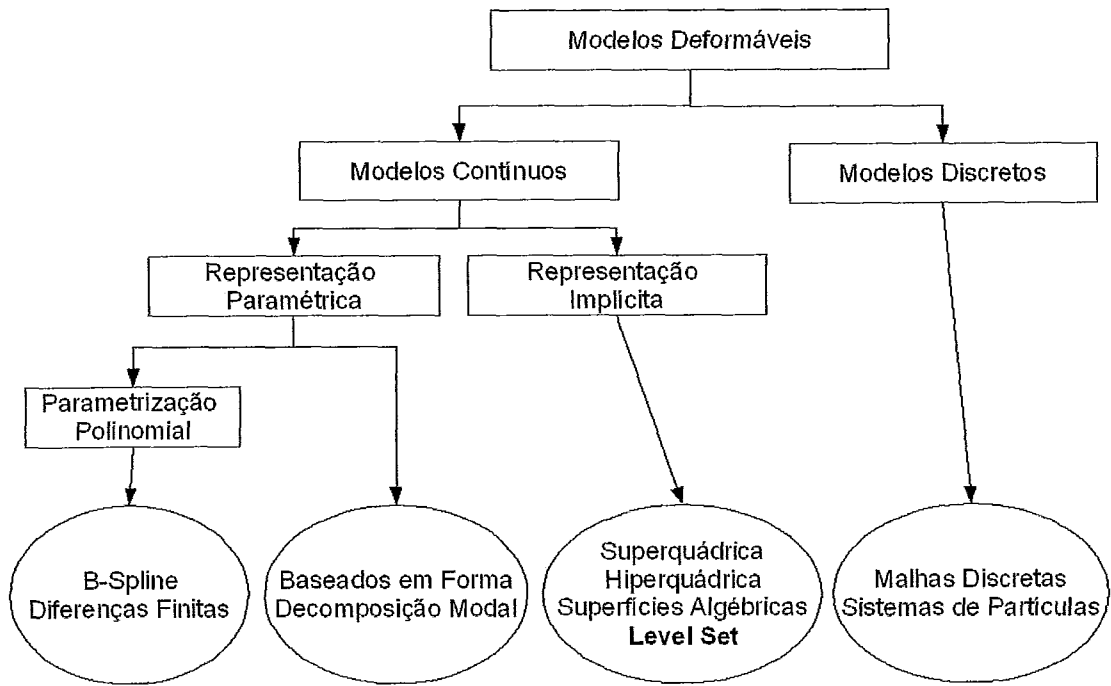


Figura 1.2: Classificação para os algoritmos de modelos deformáveis.

lizam uma frequência próxima de 1 MHz, emitindo através de uma fonte de cristal piezoelétrico que fica em contato com a pele recebendo os ecos gerados. Conforme a densidade e composição das estruturas, a atenuação e mudança de fase dos sinais emitidos varia, sendo possível a tradução em uma escala de cinza, que formará a imagem dos órgãos internos [6]. A ultra-sonografia é um dos métodos de diagnóstico por imagem mais versáteis e presentes na medicina, de aplicação relativamente simples e com baixo custo operacional.

Devido ao processo de aquisição, estas imagens possuem algumas características que as distinguem das demais. Dentre elas, podemos destacar duas principais: pouca definição das bordas e presença de *speckles* - ruídos multiplicativos localmente distribuídos. Considerando que os métodos de segmentação baseados em *snakes* fazem uso, geralmente, de informações extraídas do gradiente da imagem, as características destacadas acima tornam a segmentação de ultra-sonografias um trabalho relativamente distinto das demais aplicações em imagens médicas, já que o gradiente não seria suficiente para deter a evolução da curva nos pontos de bordas mal definidas.

A Figura 1.3 mostra alguns exemplos de ultra-sonografias. Repare a dificuldade

em definir com precisão a fronteira entre uma determinada região de interesse - área central mais escura - e o resto da imagem. Obviamente, a correta definição da fronteira da área de interesse deverá ser apontada por um especialista.

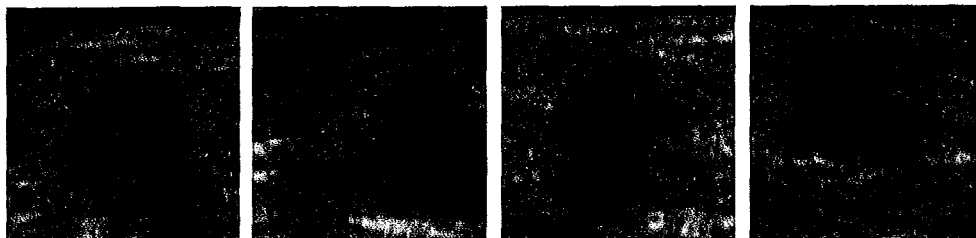


Figura 1.3: Amostra de algumas ultra-sonografias. Considere, neste exemplo, as regiões mais escuras como sendo de interesse. Note que estas não possuem suas bordas bem definidas.

1.4 Proposta

O objetivo desta tese é apresentar um estudo sobre os principais métodos de contornos ativos implícitos aplicados à segmentação de imagens médicas. Iniciamos com uma pesquisa descrevendo o modelo dos *level sets*, apresentados originalmente por Osher e Sethian em [7]. Nela, deverão ser abordados os conceitos de evolução da curva, desenvolvimento da função de velocidade, cálculo da curvatura e do vetor normal, e manipulação de mudanças topológicas, como divisões e uniões entre curvas.

Em seguida, será apresentado um estudo sobre alguns dos principais algoritmos de contornos ativos implícitos aplicados à segmentação de imagens médicas. Neste estudo, separaremos as abordagens em duas classe principais: os métodos que fazem uso dos regularizadores - agregam informações estatísticas da região ao modelo de evolução da curva - e os sem regularizadores - baseados somente nas informações de gradiente e curvatura.

Será apresentado ainda um estudo destes algoritmos aplicados, especificamente, à segmentação de ultra-sonografias. Os modelos abordados pertencem a classe de algoritmos de segmentação que fazem uso dos regularizadores. Esta estratégia é usada para prevenir um possível ultrapassagem da curva de evolução nos pontos onde a borda possui pouca definição. Além disso, estes modelos possuem ainda mecanismos de adaptação à *speckles*, característica comum às imagens geradas por

ultra-sonografias. Nos métodos de segmentação sem regularizadores, esses *speckles* podem deter a evolução da curva em regiões que não representam fronteiras, já que eles podem apresentar um valor de gradiente elevado.

Para validar, testar e comparar os métodos estudados, será realizada a implementação destes algoritmos, bem como a de uma interface gráfica que ofereça praticidade na interação com os modelos de segmentação. Esta aplicação deverá ainda ser multi-plataforma - permitir a execução em diferentes sistemas operacionais - e ser desenvolvida em sua totalidade com ferramentas gratuitas, evitando assim a geração de custos no desenvolvimento e na distribuição da aplicação.

Este trabalho está dividido da seguinte forma: o Capítulo 2 introduz a idéia dos *level sets* e mostra algumas aplicações fundamentais sobre esta abordagem; o Capítulo 3 descreve algumas das principais metodologias aplicadas à segmentação de imagens médicas utilizando os *level sets*; o Capítulo 4 se restringe às abordagens propostas para segmentação de ultra-sonografias; o Capítulo 5 descreve a implementação dos algoritmos e da interface gráfica desenvolvida para interação com o usuário; o Capítulo 6 ilustra e discute os resultados obtidos; e, finalmente, o Capítulo 7 apresenta algumas conclusões. Em adicional, nos apêndices podem ser encontradas algumas derivações matemáticas omitidas no corpo deste trabalho.

Capítulo 2

Level Sets

Entende-se como borda a curva que envolve um subconjunto regular do plano. Por meio dela é possível avaliar características inerentes a este subconjunto, tais como posição e área. Além disso, é possível ainda estimar o comportamento deste conjunto, tal como se ele está em movimento ou parado, se o movimento é de expansão ou contração e qual a velocidade desta movimentação, entre outras características.

Entretanto, desenvolver um modelo computacional capaz de descrever estas variáveis de maneira eficiente não é uma tarefa fácil. Foi pensando neste problema que Osher e Sethian, em 1987, desenvolveram a metodologia dos *level sets* [7].

O objetivo destes pesquisadores era fornecer um algoritmo capaz de descrever o movimento realizado por uma frente de propagação. Além disso, este modelo deveria ser robusto o suficiente para lidar com os estímulos físicos que poderiam alterar o estado desta frente de propagação. Alguns exemplos destes fenômenos físicos podem ser vistos quando o vento impulsiona as ondas do mar sobre a praia, quando espalha as chamas do fogo sobre a encosta de uma montanha, ou ainda, quando a gravidade e a viscosidade agem sobre a água ao ser derramada no chão. Nestes casos, as superfícies em movimento seriam a do mar, a da chama de fogo e a da água no chão. Os estímulos físicos seriam o vento, a gravidade e a viscosidade.

Como se estas variáveis já não fossem bastante para gerar suficiente complexidade ao problema, outros fenômenos complicadores deveriam ser administrados, tais como a divisão das suas bordas em duas ou mais curvas, ou ainda, o inverso, a união de duas superfícies distintas em uma única. Usando ainda as ilustrações anteriores, um exemplo poderia ser tomado se o vento dividisse a linha de frente do fogo em duas

componentes e as movimentasse em direções opostas. Em [8], Sethian usa inúmeras ilustrações para explicar a concepção e os problemas que devem ser solucionados pelo método dos *level sets*.

2.1 Idéia Inicial

Para tratar, em especial, este tipo de movimento entre bordas ou superfícies que determinam mudanças topológicas, Osher e Sethian publicaram seu primeiro trabalho sobre *level sets* em 1987 [7]. A idéia principal por trás de sua metodologia estava em adicionar uma dimensão ao problema, ou seja, a descrição do movimento de uma curva bidimensional seria feita num contexto tridimensional. Além disso, a abordagem era diretamente aplicável a problemas n-dimensionais.

A idéia de se adicionar uma dimensão ao problema foi contestada por alguns autores que diziam que a introdução de mais uma coordenada geraria alto custo computacional. Estes autores chegaram a descrever tal solução como um “ato de violência” [9, 8].

No entanto, desde o início das pesquisas, ocorridas no final da década de oitenta, até os dias de hoje, o modelo de *level sets* tem se tornado tema para muitos pesquisadores e área de grande interesse na literatura científica, oferecendo aplicabilidade em diversos ramos da ciência.

Nas próximas seções será explicada com detalhes toda estrutura necessária para se resolver um problema de descrição de bordas através do método de *level sets*. Alguns conceitos básicos inerentes ao assunto, como funções de velocidade, forma de evolução da interface de propagação e forças de parada, serão discutidos ao longo deste capítulo. O objetivo é fornecer o conhecimento necessário para se poder discutir algumas aplicações dos *level sets* em segmentação de imagens.

2.2 Conceitos

Como um exemplo hipotético, imagine um pedaço de corda com suas extremidades coladas, de maneira que ela seja uniforme. Disponha esta corda sobre o chão não permitindo que hajam partes sobrepostas. Uma curva fechada com esta topologia é dita **simplex**. Caso haja sobreposição de segmentos da corda, esta não será

uma curva fechada simples (veja Figura 2.1). Este conceito pode ser estendido ao espaço tridimensional, e assim uma superfície como uma esfera, que não possui auto-intersecções, seria chamada de **superfície fechada simples**.

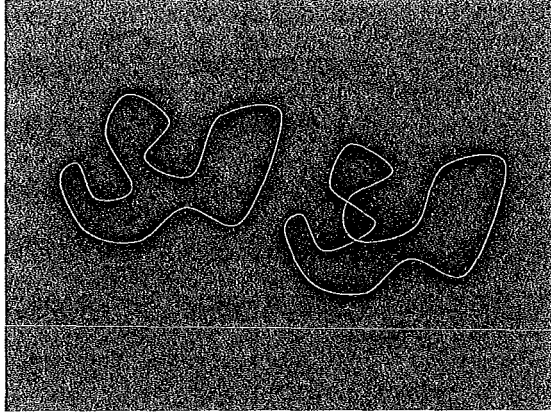


Figura 2.1: Exemplos de curvas fechadas simples e não simples.

Neste trabalho trataremos essencialmente de curvas e superfícies simples. Quando estes conceitos não se aplicarem, isto será indicado explicitamente.

Além disso, deve-se atentar ainda para o conceito de frente de propagação. No contexto desta pesquisa, uma frente deve ser entendida como uma curva ou superfície que separa dois meios que estão interagindo entre si, isto é, ela representa a borda ou fronteira entre duas regiões [10].

2.3 Movimento

Considere agora uma frente de propagação - seja uma curva no R^2 ou uma superfície no R^3 - separando duas regiões e se movendo em direção à normal, que em todo ponto da borda é orientada no mesmo sentido: ou para dentro ou para fora da borda.

A velocidade com a qual a borda se movimenta pode depender de inúmeros fatores. Uma forma de expressar a função velocidade pode ser:

$$F = F(L, G, I) \tag{2.1}$$

onde F é a função velocidade e os parâmetros L , G e I são os fatores dos quais

ela depende. Segundo Sethian [11], de uma maneira geral, estes parâmetros podem ser descritos da seguinte forma:

- L : são as propriedades locais. São aquelas determinadas, por exemplo, por informações geométricas locais, tais como curvatura e vetor normal;
- G : são as propriedades globais da interface. São aquelas que dependem da forma e da posição da interface;
- I : são as propriedades independentes, aquelas que não dependem da forma da interface.

Um dos maiores desafios na abordagem *level set* é produzir um modelo adequado para uma dada função de velocidade F . Este assunto será abordado mais detalhadamente na seção 2.6.

Dados a função velocidade F e a posição da interface, o objetivo agora é descrever o movimento realizado por ela.

2.4 Movimento por Curvatura

Para movimentar uma curva, são necessárias duas informações básicas: direção e velocidade. Como um exemplo de modelo de movimento, podemos tomar a direção como sendo indicada pelo vetor normal da curva e a velocidade como dependente da curvatura. A curvatura mede o quanto varia a inclinação da curva em um determinado ponto [12, 13, 14].

Para exemplificar, podemos pensar em um círculo. Neste caso, a curvatura é sempre constante, seja qual for o ponto considerado. Vale acrescentar que, quanto menor o círculo, maior é sua curvatura, dado que seu valor nesse caso será exatamente o inverso do raio.

O movimento da borda será perpendicular a ela, com velocidade proporcional à sua curvatura. Deve-se observar que, como a curvatura pode ser positiva ou negativa, algumas partes da curva podem ser mover para fora enquanto outras se movimentarão para dentro. A Figura 2.2 ilustra este aspecto.

Neste exemplo, as setas externas indicam que a curvatura é negativa e que portanto, a curva se move para fora. As setas internas indicam que a curvatura é

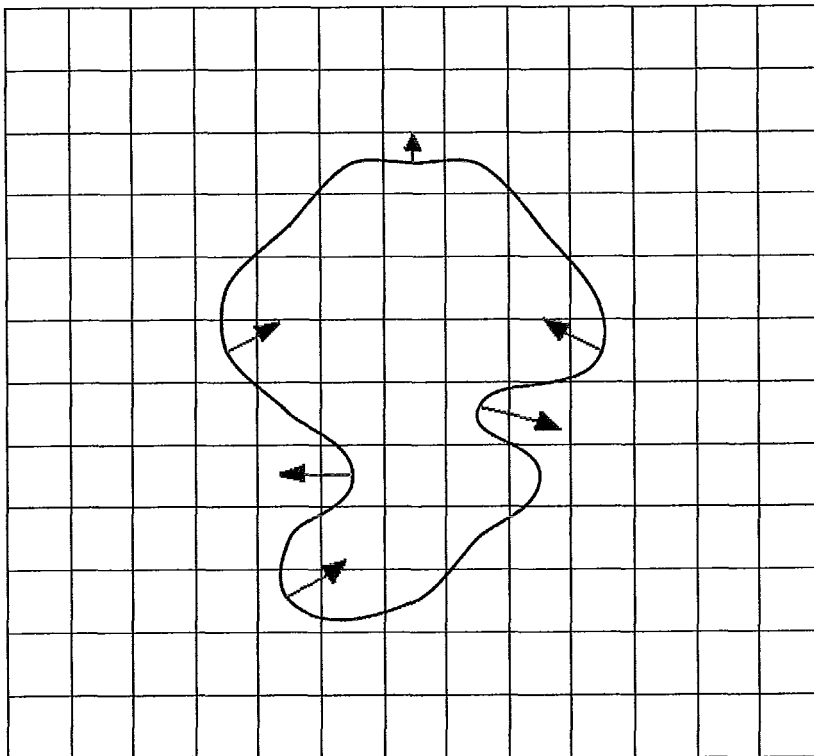


Figura 2.2: Nos pontos onde a curvatura é positiva a curva se movimenta para dentro. Onde ela é negativa a curva se move para fora.

positiva, movendo a curva para dentro. É importante notar que o tamanho das setas é proporcional ao valor da curvatura no referido ponto [15].

Considerando que o movimento com velocidade dependente da curvatura é um componente de muitos fenômenos físicos, existe um esforço contínuo em modelar mecanismos computacionais para descrever tais movimentos. Alguns trabalhos dedicados a modelagem destes fenômenos podem ser citados, tais como [16, 17, 18, 19, 14, 15, 7, 12].

2.5 Representação

Seja a função $z = \phi(x, y, t = 0)$, de forma que dado um ponto qualquer (x, y) ela retorne uma altura z referente a distância entre o ponto (x, y) dado e a frente de propagação no instante de tempo $t = 0$.

Desta forma, o gráfico de ϕ que intersecta o plano xy exatamente na frente de propagação em $t = 0$ é mostrado na Figura 2.3.

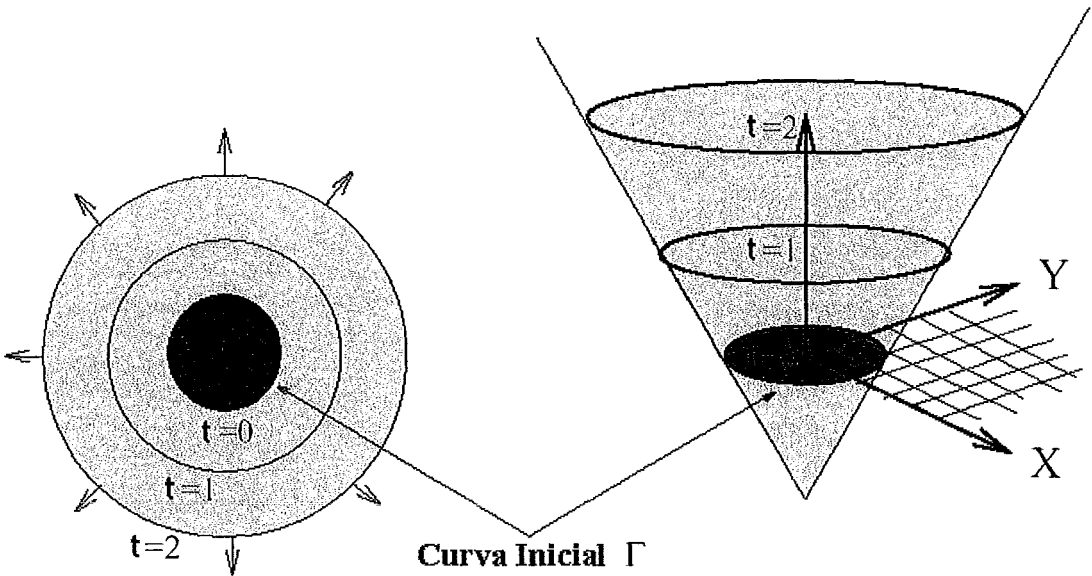


Figura 2.3: Representação da frente de propagação.

O gráfico de ϕ , representado em cinza claro, constitui a chamada “superfície *level set*”. A intersecção dessa superfície com cada plano horizontal delimita um *level set*.

A superfície em cinza claro é chamada de superfície *level set* [11, 8], já que aceita qualquer ponto como entrada e devolve um único valor como saída: a distância entre o ponto e a interface. A borda da interface em cinza escuro é chamada de nível zero, ou zero *level set*. Nela estão todos os pontos que possuem distância igual a zero.

Sendo assim, é possível ligar a evolução da função ϕ à propagação da interface em relação ao tempo. Em qualquer momento da evolução, a frente de onda da propagação é dada pelo nível zero da função ϕ (Figura 2.4).

Formalmente, dada uma curva de nível zero C_t no instante t , existem três valores possíveis para a função $\phi(\bullet, \bullet, t)$:

$$\phi(x(t), y(t), t) \begin{cases} > 0, \text{ se } (x, y) \text{ está fora de } C, \\ = 0, \text{ se } (x, y) \text{ está sobre de } C, \\ < 0, \text{ se } (x, y) \text{ está dentro de } C \end{cases} \quad (2.2)$$

Considerando um modelo unidimensional, nós podemos derivar a evolução da frente que é caracterizada pela equação

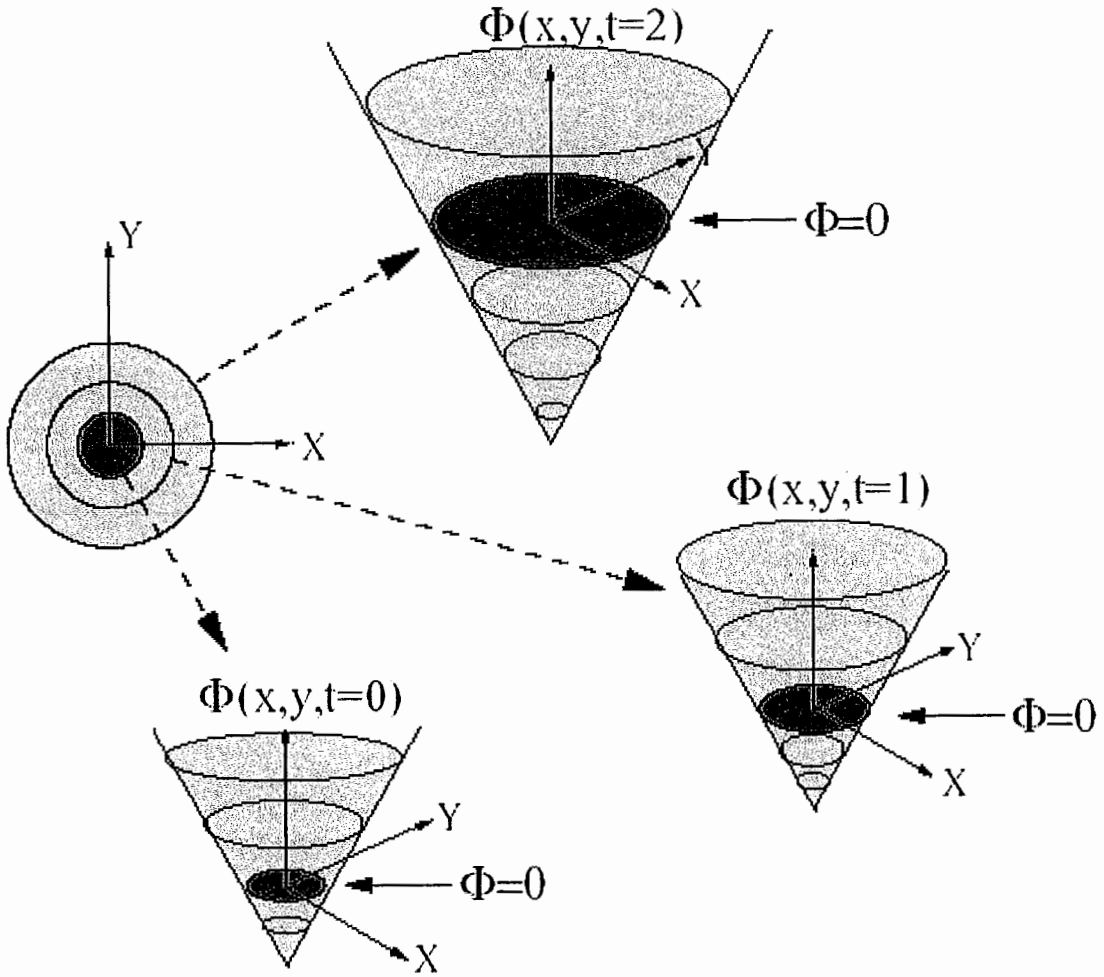


Figura 2.4: Evolução da interface ao longo do tempo.

$$\phi(x(t), t) = 0. \quad (2.3)$$

Usando a regra da cadeia, temos então que

$$\phi_t(x(t), t) + \nabla\phi(x(t), t) \cdot x'(t) = 0. \quad (2.4)$$

Supondo,

$$x'(t) \cdot n(x(t), t) = V_n(x(t), t), \quad (2.5)$$

então,

$$n(x(t), t) = \frac{\nabla\phi(x(t), t)}{\|\nabla\phi(x(t), t)\|} \quad (2.6)$$

e assim, a evolução de ϕ pode ser escrita como

$$\phi_t(x(t), t) + V_n(x(t), t) \|\nabla\phi(x(t), t)\| = 0. \quad (2.7)$$

2.6 Escolha da função de velocidade

A função de velocidade V_n depende de fatores relacionados às propriedades locais da curva, como curvatura, por exemplo, e de fatores relacionados às propriedades globais da curva, como forma e posição. Isto pode ser usado para controlar a curva de diversas maneiras. O método original de *level sets* propôs o uso de V_n como sendo a soma de dois termos

$$N_n = V_{n0} + V_{n1}(k) \quad (2.8)$$

onde V_{n0} é um termo constante de propagação e V_{n1} é uma função escalar sobre a curvatura k , que é dada por

$$k = \operatorname{div}\left(\frac{\nabla\phi}{\|\nabla\phi\|}\right) \quad (2.9)$$

O termo V_{n0} que é determinado por uma “*balloon force*” é independente da geometria da frente de propagação, a qual se movimentava uniformemente com velocidade V_{n0} , se contraindo ou se dilatando dependendo do sinal de V_{n0} [20].

Tomando como exemplo o caso em que o modelo dos *level sets* é usado para segmentação de imagens, é esperado que a curva cesse sua evolução se coincidir com a borda de um segmento. Isto pode ser realizado através da multiplicação do termo de velocidade por um termo dependente da imagem $g(x, y)$. Esse termo deve fazer a velocidade inversamente proporcional ao gradiente da imagem e pode ser definido da seguinte maneira:

$$g(x, y) = \frac{1}{1 + \frac{|\nabla I_\sigma(x, y)|^2}{\lambda^2}} \quad (2.10)$$

onde I_σ é a imagem I após um filtro Gaussiano definido por:

$$G_\sigma(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{|x^2| + |y^2|}{2\sigma^2}\right) \quad (2.11)$$

A Equação 2.10 é uma das usadas por Perona e Malik em seu trabalho de suavização de imagens usando equação de difusão [21]. No trabalho original, este termo foi usado para parar o processo de difusão próximo às bordas de um segmento. Através de sutis alterações no valor de λ , nós podemos controlar a velocidade nas bordas dos segmentos de forma que ela tende a zero quando se aproxima das bordas. Com a inclusão desse termo no cálculo de V_n a equação de evolução da frente de propagação pode ser escrita como:

$$\frac{\partial \phi}{\partial t} = g(x, y) \|\nabla \phi\| \left(\operatorname{div} \left(\frac{\nabla \phi}{\|\nabla \phi\|} \right) + V_n \right) \quad (2.12)$$

Modificações quanto ao termo de velocidade têm sido propostas por diversos autores. Por exemplo, Yezzi e seus colaboradores [22, 23] observaram que o termo de evolução dado pela Equação 2.10, em alguns casos, não faz a interface parar nas bordas. Eles propuseram então um termo adicional da forma $(\nabla g \cdot \nabla \phi)$ que faz com que a Equação 2.12 tome a seguinte forma:

$$\frac{\partial \phi}{\partial t} = g(x, y) \|\nabla \phi\| \left(\operatorname{div} \left(\frac{\nabla \phi}{\|\nabla \phi\|} \right) + F_0 \right) + \nabla g \cdot \nabla \phi \quad (2.13)$$

Neste modelo, o termo adicional possui função dupla quando a curva está próxima à borda de um segmento. Ele tem o efeito de atrair a frente de propagação quando ela se aproxima de uma borda e de empurrá-la de volta se ela passar da borda. Note que com a adição do termo $(\nabla g \cdot \nabla \phi)$, a Equação 2.13 torna-se equivalente ao modelo de *snake* geodésica que foi proposto simultaneamente por Caselles et al [1] e Kichenassamy et al [22].

2.7 Inicializando a curva

O passo chave em ambas as abordagens, implícita ou paramétrica, é a escolha do local onde colocar a curva inicial. Considerando que o contorno pode se mover para fora ou para dentro, seu local inicial determinará a segmentação que será obtida. Por exemplo, se existe um único objeto na imagem e o contorno é colocado externamente a este e com movimentação para dentro, a segmentação encontrará a borda externa do objeto. Entretanto, se o objeto tem um furo no meio, não será possível obter a borda deste furo. Para isso, a curva inicial deveria ser colocada dentro do furo e propagada para fora (expandindo) ou posicionada dentro do segmento e suficientemente mais próxima da borda interna do que da externa. É importante lembrar que mais de uma curva pode ser usada como inicialização do level set.

Neste trabalho foram abordados dois métodos para inicialização do contorno. Para obter bordas externas de segmentos de interesse na imagem, foi considerado um contorno qualquer que envolva o segmento. Esta curva é então propagada de fora para dentro (contração). Para obter as bordas internas do segmento, nós iniciamos o algoritmo com um pequeno contorno circular contido na região delimitada pela borda externa do segmento e a fazemos expandir. Se o contorno inicial estiver contido no próprio segmento e este não for simplesmente conexo, mudanças topológicas devem acontecer durante a evolução da *snake*.

Estes dois tipos de inicialização de curva são suficientes para analisarmos o comportamento das *snakes* implícitas. Vale ressaltar que, para que a segmentação de determinadas imagens atinjam resultados satisfatórios, a inicialização do contorno próximo da borda almejada pode ser essencial.

2.8 Cálculo da função distância

Uma vez inicializado o contorno, é preciso calcular a função ϕ inicial que conforme foi visto, será uma distância com sinal a este contorno (ver Figura 2.5). O cálculo pode ser feito através da resolução da equação Eikonal [11, 24].

Suponha $V_n(x(t), t) > 0$. Como a frente de propagação se move para fora (expansão), uma maneira de se caracterizar a evolução da frente é calcular o tempo de chegada $T(x, y)$ onde a curva cruza o ponto (x, y) . Esta função de chegada está

7	6	5	4	4	4	3	2	1	1	1	2	3	4	5
6	5	4	3	3	3	2	1	0	0	0	1	2	3	4
5	4	3	2	2	2	1	0	-1	-1	-1	0	1	2	3
4	3	2	1	1	1	0	-1	-2	-2	-2	-1	0	1	2
3	2	1	0	0	0	-1	-2	-3	-3	-2	-1	0	1	2
2	1	0	-1	-1	-1	-2	-3	-3	-2	-1	0	1	2	3
2	1	0	-1	-2	-2	-3	-3	-2	-1	0	1	2	3	4
2	1	0	-1	-2	-2	-2	-2	-1	0	1	2	3	4	5
3	2	1	0	-1	-1	-1	-1	-1	0	1	2	3	4	5
4	3	2	1	0	0	0	0	-1	-1	0	1	2	3	4
5	4	3	2	1	1	1	1	0	0	1	2	3	4	5
6	5	4	3	2	2	2	2	1	1	2	3	4	5	6

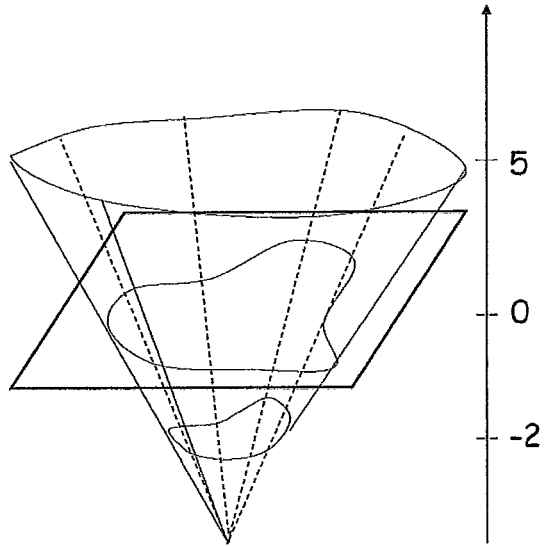


Figura 2.5: Representação gráfica da evolução de uma curva e os valores da função na grade.

relacionada com V_n através da equação

$$\| \nabla T \| V_n = 1, \quad (2.14)$$

e o contorno inicial é, logicamente, o lugar geométrico dos pontos onde T é nula. Quando a velocidade V_n depende somente da posição, a Equação 2.14 é referida como a equação Eikonal. A solução desta equação para uma velocidade constante unitária dá para cada instante t a função distância com sinal à curva de nível zero de t . O sinal de V_n depende da localização do pixel em relação ao contorno. Neste trabalho, foi usado sinal negativo para pixels internos ao contorno e positivo para os externos. Uma maneira eficiente para resolver a equação Eikonal é descrita em [25].

Com a evolução da frente de propagação, a função ϕ poderia, em princípio, ser atualizada apenas em uma vizinhança do contorno atual, dado que isto seria suficiente para determinar a evolução da curva. Entretanto, ao fazer isso a função ϕ deixaria de ser uma distância e, além disso, poderíamos não mais poder fazer a atualização correta dessa função, dado que partiríamos de valores incorretos.

Como resultado disto, a curva pode parar sua evolução próximo a uma borda sem atingí-la. Uma solução para este problema é reinicializar a função distância ϕ

periodicamente. Nós usamos a mesma equação Eikonal para esta finalidade. Entretanto, deve ser notado que isto pode adicionar um significativo custo computacional ao método level set se for feito freqüentemente.

2.9 Vantagens do Level Set

Existem algumas vantagens associadas à perspectiva de propagação da frente de onda utilizando o método dos *level sets*. Abaixo estão algumas.

- O método é diretamente aplicável a dimensões superiores.
- Alterações topológicas como divisão ou junção são manipuladas naturalmente. Em qualquer momento a borda é dada pelo nível zero da função, ou seja, a posição da frente no instante de tempo t é dado pelo conjunto solução de $\phi(x, y, t) = 0$.
- Propriedades geométricas são facilmente determinadas. Por exemplo, em qualquer ponto da frente de propagação o vetor normal é dado por

$$n = \frac{\nabla\phi}{|\nabla\phi|} \tag{2.15}$$

e a curvatura é dada pelo cálculo do divergente do vetor normal, isto é:

$$k = \nabla \cdot \frac{\nabla\phi}{|\nabla\phi|} = \frac{\phi_{xx}\phi_y^2 - 2\phi_x\phi_y\phi_{xy} + \phi_{yy}\phi_x^2}{(\phi_x^2 + \phi_y^2)^{3/2}} \tag{2.16}$$

- O método permite tratar apropriadamente os cantos da frente, dada uma adequada escolha de parâmetros e modelos.
- Controle de resolução através de parâmetros como Δh (espaçamento entre as células da grade) e Δt (passo de tempo)

2.10 Complexidade

Uma maneira interessante de se calcular o custo e a eficiência de um algoritmo é analisar sua complexidade teórica [26]. Para resolver um problema de fazer evoluir uma curva de nível num espaço de dimensão 2 discretizado em uma grade $n \times n$ pelo método dos *level sets*, são necessárias $O(n^2)$ operações por passo de tempo. A versão tridimensional do problema custará $O(n^3)$ [27, 7].

Algumas técnicas foram propostas visando diminuir o custo computacional relacionado ao cálculo da função ϕ . Métodos do tipo *Narrow Band* restringem a região de atualização da função ϕ às áreas próximas à interface. Já os métodos do tipo *Fast Marching* oferecem ganho de performance quando a evolução se restringe a somente uma direção [27].

2.11 Aplicações

Desde sua proposta original o método de *level sets* tem ganho grande popularidade, sendo usado nas mais diversas aplicações. Nesta seção estão relatadas algumas das áreas e aplicações desenvolvidas utilizando a abordagem *level set*.

Na área de *motion tracking*, por exemplo, onde as questões usuais estão em estimar o campo de movimento e segmentar as regiões onde ele é uniforme ou atende a alguma relação, alguns autores fazem uso dos modelos de *level sets* para a segmentação em função do movimento. Mansouri e Konrad, por exemplo, propuseram uma abordagem interativa para unir estimação do campo de movimento e segmentação baseada em *level sets* [28]. Seu algoritmo para segmentação é puramente baseado no movimento, diferentemente da abordagem feita por Paragios e Deriche [29], onde explicitamente há combinação de segmentação em função do movimento e da intensidade da imagem, causando a dependência das bordas estarem bem definidas. A metodologia proposta por Mansouri e Konrad apresenta um algoritmo baseado somente no movimento das imagens, sendo efetiva mesmo quando as bordas não estão bem definidas [30, 31].

Ainda em *motion tracking*, Catte, Dibos e Koepfler apresentaram um esquema para segmentação de movimento via cálculo de curvatura, que conjuga a utilização dos *level sets* e de morfologia matemática [18, 32].

Seguindo uma tendência dos últimos anos e visando um ganho de performance, iniciou-se a implementação dos modelos de *level sets* em placas gráficas. Um dos pioneiros foi o trabalho de Lefohn et al em 2003 [33]. Seu artigo aborda os problemas relacionados a segmentação tridimensional, focando no alto custo computacional e na dificuldade de parametrização. Os autores propõem o uso das placas gráficas para aceleração do processo de cálculo da superfície de evolução e da visualização dos resultados. Segundo os autores, suas principais contribuições são um sistema integrado capaz de fornecer controle interativo sobre o processo de segmentação e visualização, uma solução em GPU cerca de 15 vezes mais rápida do que os métodos anteriores, um novo e eficiente esquema de troca de mensagens entre GPU e CPU e a visualização tridimensional dos resultados em tempo real.

Além das já citadas, inúmeras outras áreas agregam as funcionalidades dos modelos *level sets*, tais como os trabalhos em geometria apresentados por Chopp e Sethian [14, 34], em geração de grades [19], em pesquisas sobre mecânica dos fluidos [35] e combustão [36], fabricação de dispositivos eletrônicos [37, 38], *morphing* [39, 40], visão estéreo [41], *shape from shading* [42], segmentação de imagens coloridas [43], cálculo do caminho mais curto [42] e classificação de imagens [44], entre outras.

Outro ramo da área de processamento de imagens onde se faz grande uso da abordagem *level set* é a segmentação de imagens médicas. Este assunto, entretanto, será abordado como mais detalhes nos capítulos 3 e 4.

Capítulo 3

Contornos Ativos

A maioria das estratégias usadas em contornos ativos visa minimizar um funcional de energia composto de um termo que é baseado na imagem e um outro termo relacionado com a energia interna ou modelo do contorno, o qual penaliza a falta de suavidade da curva. Os dois modelos principais são os paramétrico e o implícito, sendo este último implementado sob a formulação dos *level sets*.

Neste capítulo nos dedicaremos aos contornos ativos implícitos e às suas aplicações quanto à segmentação de imagens médicas. Serão abordadas algumas das metodologias propostas e suas variações, bem como algumas vantagens e desvantagens.

Antes de falarmos sobre os contornos ativos implícitos, cabe uma breve descrição sobre o modelo das *snakes* paramétricas.

3.1 Snakes

O método foi proposto originalmente por Kass, Witkin e Terzopoulos [45]. Os autores apresentaram um modelo de contorno definido no domínio da imagem, que pode se deslocar sob a influência de forças internas (definidas em função do próprio contorno) e externas, calculadas a partir de informações extraídas da imagem. As informações inerentes ao contorno devem manter sua suavidade durante sua deformação, enquanto que as informações vindas da imagem devem fazer o contorno se mover em direção às fronteiras da região que se deseja.

Segundo McInerney e Terzopoulos [46] as *snakes* são fruto da confluência da geometria, da física e da teoria da aproximação. Elementos geométricos são usados

para expressar a forma do objeto, a física impõe restrições à variação da forma ao longo do espaço e do tempo, e a teoria da aproximação permite formalizar os mecanismos que possibilitam o ajuste dos modelos aos dados.

Uma *snake* pode ser definida parametricamente da seguinte forma:

$$c(s) = (x(s), y(s)) \quad (3.1)$$

onde $x(s)$ e $y(s)$ são as coordenadas x e y ao longo do contorno, e o parâmetro $s \in [0, 1]$. O funcional de energia que deve ser minimizado pode ser escrito como:

$$E_{snake} = \int_0^1 E_{int}(c(s)) + E_{imagem}(c(s)) + E_{artificial}(c(s)) ds \quad (3.2)$$

em que E_{int} , E_{imagem} e $E_{artificial}$ representam a energia interna, as forças inerentes a imagem e a energia artificial inserida no modelo, respectivamente.

A energia interna da *snake* pode ser expressa como:

$$E_{int} = \alpha(s) \left| \frac{dv}{ds} \right|^2 + \beta(s) \left| \frac{d^2v}{ds^2} \right|^2 \quad (3.3)$$

onde $\alpha(s)$ e $\beta(s)$ representam a elasticidade e a rigidez do contorno. Enquanto que o primeiro termo representa uma resistência a tração ou compressão - variação da extensão - o segundo representa uma resistência a dobra - variação da curvatura.

O segundo termo da Equação 3.2 é obtido a partir da imagem e pode se constituir numa combinação complexa de funcionais de forma que a sua minimização atraia a *snake* para características de interesse na imagem. Como exemplo, se $f(x)$ for o nível de tons de cinza da imagem em (x, y) , o funcional correspondente a uma *snake* que é atraída por contornos com gradientes de imagem elevados pode ser:

$$E_{imagem} = -|\nabla f(x, y)|^2 \quad (3.4)$$

O terceiro termo é referente às componentes que não dependem nem da forma atual do contorno nem da imagem, mas que são introduzidas ou diretamente pela ação do usuário ou por um processo automático que pode levar em conta a situação

