Capítulo 5

Modelos Alternativos de Interesse

Neste capítulo descrevemos alguns modelos baseados nos elementos discutidos no capítulo anterior que procuram corrigir as limitações inerentes ao modelo original de snakes.

Dentre os métodos apresentados, o modelo Dual e das T-Snakes têm relação direta com nosso trabalho. O modelo das AI-Snakes nâo está diretamente relacionado com nosso método, mas constitui-se em um exemplo elegante onde o conceito de invariância tem de fato um papel central. O método dos Conjuntos de Níveis é um dos modelos que inspirou o desenvolvimento das T-Snakes, particularmente no que diz respeito à idéia de uma função que distingue o interior do exterior da snake e da política usada para atualizá-la. Por esta razão será apresentado neste capítulo.

5.1 Modelos Usando Invariância

O conceito de *invariância* por transformações, largamente empregado em visão computacional (Gool *et al.* , 1995), vem sendo usado no contexto dos modelos de contornos ativos discretos para remover os efeitos indesejados da componente normal da força interna ou para facilitar a incorporação de informações a priori (Gunn & Nixon, 1997; Ip & Shen, 1998).

Esta idéia vem sendo aplicada para contornos ativos fechados e consiste primeiramente na utilização de uma energia interna que possua propriedades adicionais de invariância além das usuais por translação e rotacão.

Esta energia interna pode ser obtida atravéz de um modelo de forma vinculado a um *protótipo* (Ip & Shen, 1998) ou simplesmente derivada de elementos que tenham as propriedades de invariância desejadas (Ip & Shen, 1998; Gunn & Nixon, 1997). Com isto, evita-se o colapso da curva, uma vez que a snake tem agora a tendência natural de adquirir a forma pré-estabelecida no modelo, limitando inclusive a maneira como a snake se deforma durante sua evolução. Ao mesmo tempo, a energia interna assim definida deve fornecer o termo de regularização (ou suavização) necessário.

5.1.1 Modelo Dual

O método Dual (Gunn & Nixon, 1997; Gunn, 1996a) é um exemplo do emprego do conceito de invariância para evitar as dificuldades com a força normal. Neste caso, as transformações de interesse são as transformações rígidas (translação e rotação) e transformações de escala. Assim, um snaxel ocupando uma posição V é transformado para uma posição U segundo:

$$U = s \begin{pmatrix} \cos\theta & -sen\theta \\ sen\theta & \cos\theta \end{pmatrix} V + \begin{pmatrix} a \\ b \end{pmatrix} = AV + B \tag{5.1}$$

onde s é o parâmetro de escala, θ é o ângulo de rotação e $(a,b)^T$ é o vetor de translação.

Este método consiste de um contorno que expande a partir do interior do objeto e de um outro contorno que contrai a partir do exterior.

É estabelecida uma correspondência entre os dois contornos o que permite que os mesmos sejam interligados para obter-se uma força adicional (*driving force*) a qual é usada para retirar os contornos de mínimos locais garantindo mais robustez.

Vejamos inicialmente a definição da energia interna invariante pelas transformações em questão.

Energia Interna

A necessidade de uma correspondência (*matching*) entre os pontos dos contorno interno e externo obriga que sejam impostas restrições para a forma das snakes pois do contrário a correspondência entre pontos das duas curvas pode se tornar complicada. Esta restrição de forma é incorporada ao modelo via um modelo de forma local que passaremos a descrever.

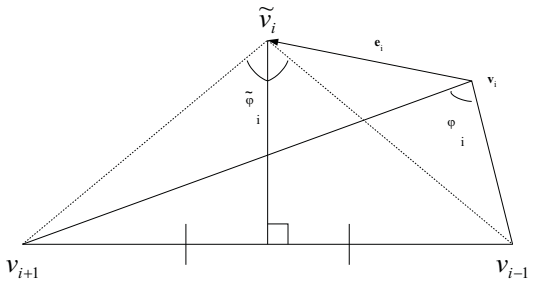

Figura 5.1: Modelo local de forma.

Na Figura 5.1, o vetor $\overrightarrow{e}_i$ é tal que

$$\|v_{i-1} - \widetilde{v}_i\| = \|v_{i+1} - \widetilde{v}_i\|\,, \tag{5.2}$$

isto é:

$$\overrightarrow{e}_i = \frac{1}{2}\left(v_{i-1} + v_{i+1}\right) - v_i + \frac{1}{2}\theta_i R\left(v_{i-1} - v_{i+1}\right), \tag{5.3}$$

onde R é uma matriz de rotação de 90^0 e θ_i está relacionado ao ângulo interno $\widetilde{\varphi}_i$ em v_i por:

$$\theta_i = \cot\left(\frac{\widetilde{\varphi}_i}{2}\right). \tag{5.4}$$

Com isto, o modelo de forma é estabelecido em (Gunn & Nixon, 1997) pela seguinte expressão:

$$E_{int} = \frac{1}{2}\sum_{i=0}^{N-1}\left(\frac{\|\overrightarrow{e}_i\|}{h}\right)^2. \tag{5.5}$$

Observe que E_{int} tem um mínimo global quando $\overrightarrow{e}_i = 0$, $i = 0, 1, ..., N-1$. Usando (5.2)-(5.4) pode ser mostrado que isto acontece quando:

$$\varphi_i = \pi\left(N-2\right)/N, i = 0, 1, ..., N-1, \tag{5.6}$$

os quais são ângulos internos do polígono regular com vértices dados pelos pontos v_i (Gunn & Nixon, 1997). Além disto, a energia (5.5) é invariante por transformações de rotação, translação e escala como desejado (Gunn & Nixon, 1997).

A energia total do modelo Dual é dada por:

$$E = \frac{\lambda}{N}E_{int}\left(v_i\right) + \left(1-\lambda\right)E_{ext}\left(v_i\right), \tag{5.7}$$

onde E_{ext} é uma forma discreta de (3.4):

$$E_{ext} = \sum_{i=0}^{N-1} P\left(v_i\right), \tag{5.8}$$

com potencial externo P dado pela expressão:

$$P\left(v_i\right) = -\left\|\nabla I\right\|^2. \tag{5.9}$$

O fator λ em (5.7) é um parâmetro de suavização tomado entre 0 e 1 (Gunn & Nixon, 1997).

Com a energia interna dada por (5.7) os problemas com a força normal interna são evitados uma vez que a curva tem agora a tendência natural a obter a forma de um polígono regular em vez de colapsar para um ponto.

Tal tendência tem também a propriedade de limitar a forma da curva durante a sua evolução. Em (Gunn & Nixon, 1997) os autores tomam vantagem desta limitação da forma das snake para estabelecer uma correspondência entre os pontos das mesmas. Tal correspondência é usada para definir uma força adicional usada para retirar uma snake de um mínimo local.

Vejamos esta força em detalhes.

Força Adicional e Mínimos Locais

Para compreendermos a utilidade desta força consideremos a Figura 5.2:

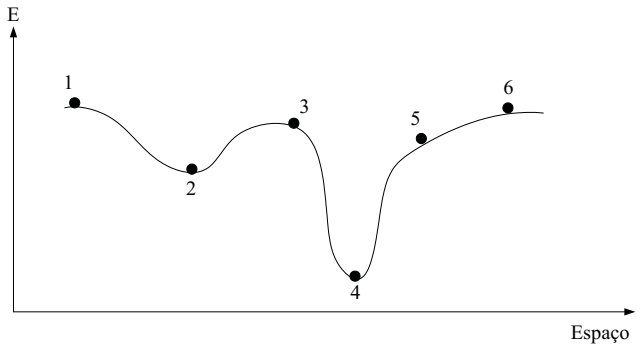

Figura 5.2: Perfil de energia.

Nesta figura temos um perfil de energia (aqui representada em 1D por simplicidade) da snake e vamos supor a mesma em uma configuração inicial com energia dada pela altura da posição 1.

Esta figura permite ilustrar os principais problemas associados aos métodos de minimização baseados em informações locais (stepest descent, Euler Lagrange, etc). Para qualquer um destes métodos, o resultado final partindo da posição 1 será uma configuração com energia dada pela posicão 2, embora o resultado desejado seja o correspondente ao ponto 4.

A dificuldade aqui está no fato que estes métodos de otimização, pela natureza local dos mesmos, não podem *saber* que na posição 4 há um vale mais profundo que em 2.

Aqui é possível apresentar a principal contribuição do método Dual: usar duas snakes e com isto estabelecer um balanço entre posição e energia, a fim de ter garantias da globalidade da solução encontrada.

Assim, se inicializarmos uma snake na posição 1 e outra na posição 6 da Figura 5.2, quando a primeira estabilizar seu movimento na posição 2 poderemos evoluir a segunda snake e quando esta atingir o repouso comparar a posição e a energia das snakes. Neste caso, a segunda snake atingirá a posição 4 indicando diretamente que a posição 2 era um mínimo local mas não global.

Tal análise é ilustrativa, mas ao mesmo tempo grosseira se lembrarmos que a energia (5.7) está definida em espaço com dimensão $2N$, onde N é o número de snaxels.

Neste caso, podemos ter duas snakes parcialmente ajustadas ao contorno, mas em posições de equilíbrio (mínimos local da energia). Certamente, uma vez que as snakes têm o mesmo número de snaxels, podemos identificar que as mesmas estão distantes e que portanto ao menos uma está em um mínimo local. Escolhida uma snake para evoluir via algum critério a ser especificado, o próximo problema é como saber automaticamente qual snaxel deve ser movimentado?

Para abordar esta questão propõe-se em (Gunn & Nixon, 1997; Gunn, 1996a) um algoritmo guloso: movimenta-se inicialmente os snaxels que aumentam menos a energia. Rigorosamente, perde-se a garantia do ótimo com este procedimento. Na prática, advogam os autores em (Gunn & Nixon, 1997; Gunn, 1996a), que os resultados têm sido satisfatórios. A utilização de algoritmos gulosos em snakes foi estudada também em (Bulpitt & Efford, 1996).

Para movimentar os snaxels de uma snake retirando-a de um mínimo local é necessário usar uma força extra a qual denominamos *Força Adicional*. O efeito desta força pode ser ilustrado pela figura 5.2. Uma vez concluído que 2 não é ótimo global, esta força extra teria a finalidade de *puxar* a snake para fora do vale correspondente ao ponto 2 até que a mesma atinja a posição 3 quando então esta terá condições de evoluir naturalmente em direção ao ponto 4.

A força adicional é dada por:

$$F_{adicional} = g\left(t\right) \frac{u_i - v_i^t}{\left\| u_i - v_i^t \right\|}, \tag{5.10}$$

onde v_i^t é um snaxel do contorno sendo processado no instante t, u_i é o contorno em repouso neste instante e $g\left(t\right)$ é a intensidade da força.

O critério de parada adotado em (Gunn & Nixon, 1997) é do tipo dado pela expresão (3.48):

$$\max_i \left\| v_i^{t+1} - v_i^t \right\| < \delta, \tag{5.11}$$

onde δ é um parâmetro pré-estabelecido.

A aquação de evolução do método Dual é finalmente dada por:

$$v_i^{t+1} = v_i^t + \frac{1}{2}\left(\lambda\frac{e_i}{h} + (1-\lambda)\,F_i\right) + g\,(t)\,\frac{u_i - v_i^t}{\|u_i - v_i^t\|}, \tag{5.12}$$

onde F_i é dada pelo gradiente do potencial externo.

O método Dual aqui descrito é também denominado *Contorno Ativo Dual Baseado em Evolução* para diferenciar do Dual baseado em programação dinâmica que apresentaremos a seguir.

O algoritmo correspondente ao método Dual descrito nesta seção pode ser sumarizado da seguinte forma.

Algoritmo

Inicialização: Interação com o usuário para definir snakes interna e externa,

Passo 1: Selecionar a snake com energia (5.7) mais elevada,

Passo 2: Se o movimento desta snake satisfazer a condição de terminação aumentar a força de arrasto (5.10) até que o movimento da snake ultrapasse o parâmetro δ em (5.11),

Passo 3: Quando a energia da snake começar a decrescer, remover a força de arrasto e evoluir a snake até seu equilíbrio,

Passo 4: Se as snakes ocupam a mesma posição, parar a execução. Caso contrário voltar ao passo 1.

Para comparar a posição das snakes no passo 4 acima é usado uma norma no espaço dos Descritores de Fourier das curvas; isto é, dados dois contornos u e v a distância entre eles é definida por:

$$d\,(u,v) = \left[\sum_{n=-M}^{M} |\mu_n - \nu_n|^2\right]^{\frac{1}{2}},$$

onde μ_n e ν_n são os Descritores de Fourier para as curvas u e v, respectivamente (Jain, 1989).

5.1.2 Metodologia do Dual

O método Dual evolucionário tem sua metodologia fundamentada em três pontos:

1) Balanço entre a energia da snake interna e da externa para escolher a snake a ser processada,

2) Critério para decidir quais snaxel de uma snake estabilizada pode ser movido,

3) Critério final de parada: snakes próximas.

Destes ítens, o mais crítico é o segundo. Isto porque, se este critério não for bem definido, porções de uma snake já acomodadas sobre a borda desejada podem ser retiradas desta posição, o que implica em perda de eficiência e performance.

5.1.3 AI-Snakes: Modelo de Forma Ativo

O modelo Dual da seção 5.1.1 incorpora um modelo de forma local.

Nesta seção, analisaremos um caso onde o modelo de forma é global. Nosso ponto de partida será o conhecimento a priori de um *protótipo* do objeto, e o fato de que a borda procurada é uma deformação deste protótipo.

Assim, é natural que o modelo de contorno ativo incorpore tanto a fidelidade ao protótipo quanto seja invariante pelas transformações que geram as possíveis deformações do objeto.

Em (Ip & Shen, 1998) a invariância é por transformações afins sobre a curva (translação e transformações lineares não-singulares). Neste método é desenvolvida uma noção de curvatura afim-invariante empregada como elemento de suavização da solução e uma energia de modelo que vincula a forma da snake ao *protótipo* em questão.

Para que o vínculo entre o protótipo e a snake seja estabelecido é necessário uma correspondência entre os pontos do protótipo e da snake, o que aliás é uma característica comum dos *modelos de forma ativa* (Ip & Shen, 1998).

Este método é apropriado para segmentação de cenas onde objetos com formas previamente conhecidas foram deformados por transformações afins ou que possam ser aproximadas por uma seqüência de transformações deste tipo.

Seja, por exemplo, uma seqüência de fotos de um mesmo objeto tiradas de diferentes posições. As transformações no caso são *projetivas* as quais podem ser aproximadas por uma seqüencia de transformações afins. O objeto fotografado (mais exatamente, sua fronteira) nos fornece o protótipo.

Neste caso, a aplicação da noção de invariância tem a finalidade de manter fidelidade ao *protótipo* e às transformações correspondentes. Ao mesmo tempo a energia interna deverá garantir também a suavidade necessária para a solução final.

Iniciemos pela análise da noção de curvatura inerente ao modelo original de snakes (seção 3.1).

Curvatura

Inicialmente, vejamos porque a curvatura gaussiana não é invariante por transformações afins. Vamos retomar a descrição continua para curvas.

Seja então uma curva suave

$$c(\tau) = (x(\tau), y(\tau)), \quad \in \Re.$$

Neste caso, a curvatura gaussiana $k(\tau)$ é dada por:

$$k(\tau) = \frac{\dot{x}(\tau)\,\ddot{y}(\tau) - \ddot{x}(\tau)\,\dot{y}(\tau)}{\left[\left(\dot{x}(\tau)\right)^2 + \left(\dot{y}(\tau)\right)^2\right]^{3/2}} \tag{5.13}$$

onde:

$$\dot{x}(\tau) = \frac{dx}{d\tau}, \ddot{x}(\tau) = \frac{d^2x}{d\tau^2},$$

idem para y.

Seja s o comprimento de arco correspondente à curva c. Então nós teremos:

$$ds^2 = (dx)^2 + (dy)^2,$$

Assim a expressão (5.13) pode ser reescrita como:

$$k(\tau) = \frac{\begin{vmatrix} \dot{x}(\tau) & \ddot{x}(\tau) \\ \dot{y}(\tau) & \ddot{y}(\tau) \end{vmatrix} \cdot (d\tau)^3}{(ds)^3} = \frac{ang}{dist} \tag{5.14}$$

onde:

$$ang = \begin{vmatrix} \dot{x}(\tau) & \ddot{x}(\tau) \\ \dot{y}(\tau) & \ddot{y}(\tau) \end{vmatrix} \cdot (d\tau)^3 \text{ e } dist = (ds)^3.$$

Vejamos como ang e $dist$ se comportam sob transformações afins. Seja então:

$$\begin{pmatrix} u(\tau) \\ v(\tau) \\ 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x(\tau) \\ y(\tau) \\ 1 \end{pmatrix} \tag{5.15}$$

a curva c transformada pela transformação afim definida pela matriz quadrada de ordem 3 do lado direito desta igualdade. Então, um cálculo direto mostra que:

$$ang = \begin{vmatrix} \dot{u}(\tau) & \ddot{u}(\tau) \\ \dot{v}(\tau) & \ddot{v}(\tau) \end{vmatrix} \cdot (d\tau)^3,$$

é transformado da seguinte forma:

$$ang = \begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} \begin{vmatrix} \dot{x}(\tau) & \ddot{x}(\tau) \\ \dot{y}(\tau) & \ddot{y}(\tau) \end{vmatrix} (d\tau)^3 = (a_{11}a_{22} - a_{12}a_{21}) \cdot ang. \tag{5.16}$$

Esta expressão mostra que o valor de ang é simplesmente alterado por um fator de escala linear dado pelo determinante da matriz que define a transformação. Para $dist$ teremos:

$$\begin{pmatrix} du \\ dv \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \cdot \begin{pmatrix} dx \\ dy \end{pmatrix}, \tag{5.17}$$

e assim teremos:

$$dist = \left((du)^2 + (dv)^2\right)^{3/2} = \left((a_{11}dx + a_{12}dy)^2 + (a_{21}dx + a_{22}dy)^2\right)^{3/2}, \tag{5.18}$$

o que mostra que o valor de $dist$ é alterado de forma não linear por transformações afins e assim $dist$ não pode ser tornado afim-invariante exceto em situações especiais. Logo, fica demonstrado que a curvatura Gaussiana não é afim-invariante, e conseqüentemente o modelo original de snakes (seção 3.1) também não (veja que a curvatura Gaussiana aparece explicitamente na equação (3.8)).

Área obtida da curvatura é afim-invariante

Vamos mostrar agora que uma versão discreta para $(1/2)\,ang$ é a área de um triângulo definido por três vértices da curva c. Seja então as aproximações das derivadas em (5.13) por diferenças finitas:

$$\begin{aligned}
\dot{x}\,(\tau) &\doteq \frac{x\,(\tau) - x\,(\tau - \Delta\tau)}{\Delta\tau}, \\
\dot{y}\,(\tau) &\doteq \frac{y\,(\tau) - y\,(\tau - \Delta\tau)}{\Delta\tau}, \\
\ddot{x}\,(\tau) &\doteq \frac{x\,(\tau) - 2x\,(\tau - \Delta\tau) + x\,(\tau - 2\Delta\tau)}{(\Delta\tau)^2}, \\
\ddot{y}\,(\tau) &\doteq \frac{y\,(\tau) - 2y\,(\tau - \Delta\tau) + y\,(\tau - 2\Delta\tau)\,.}{(\Delta\tau)^2}
\end{aligned} \tag{5.19}$$

Usando estas aproximações podemos escrever o numerador da expressão (5.14) da seguinte forma:

$$\frac{1}{2}ang \doteq \frac{1}{2}\begin{vmatrix} x\,(\tau - 2\Delta\tau) & x\,(\tau - \Delta\tau) & x\,(\tau) \\ y\,(\tau - 2\Delta\tau) & y\,(\tau - \Delta\tau) & y\,(\tau) \\ 1 & 1 & 1 \end{vmatrix}, \tag{5.20}$$

o qual é a área do triângulo formado pelos vértices:

$(x\,(\tau - 2\Delta\tau)\,, y\,(\tau - 2\Delta\tau))\,, (x\,(\tau - \Delta\tau)\,, y\,(\tau - \Delta\tau))\,, (x\,(\tau)\,, y\,(\tau))\,.$

Vejamos como se comporta a área dada por (5.20) sob transformações afins. Neste caso teremos:

$$area_{\tau_1\tau_2\tau_3} = \frac{1}{2}\begin{vmatrix} x(\tau_1) & x(\tau_2) & x(\tau_3) \\ y(\tau_1) & y(\tau_2) & y(\tau_3) \\ 1 & 1 & 1 \end{vmatrix}, \tag{5.21}$$

Denotemos por $area^{transf}_{\tau_1\tau_2\tau_3}$ esta área transformada de acordo com a expressão (5.15); isto é:

$$area^{transf}_{\tau_1\tau_2\tau_3} = \frac{1}{2}\begin{vmatrix} u(\tau_1) & u(\tau_2) & u(\tau_3) \\ v(\tau_1) & v(\tau_2) & v(\tau_3) \\ 1 & 1 & 1 \end{vmatrix}, \tag{5.22}$$

Expressando agora 5.22 acima em função da área original $area_{\tau_1\tau_2\tau_3}$ obteremos:

$$\begin{aligned} area^{transf}_{\tau_1\tau_2\tau_3} &= \frac{1}{2}\left|\begin{pmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ 0 & 0 & 1 \end{pmatrix}\right| \begin{vmatrix} x(\tau_1) & x(\tau_2) & x(\tau_3) \\ y(\tau_1) & y(\tau_2) & y(\tau_3) \\ 1 & 1 & 1 \end{vmatrix} = \\ & (a_{11}a_{22} - a_{12}a_{21})\, area_{\tau_1\tau_2\tau_3}. \end{aligned} \tag{5.23}$$

Esta expressão mostra que a área de qualquer triângulo, quando apropriadamente normalizada, é invariante por transformações afins. Com isto estamos em condições de definir uma energia interna invariante por estas transformações.

Energia Interna

A energia interna da AI-snake é dada pela seguinte expressão:

$$E_{int} = E_{suav} + E_{modelo}, \tag{5.24}$$

onde E_{suav} e E_{modelo} são dados como segue.

Para definir o primeiro termo retomemos o resultado (5.23) e através dele definamos uma noção de curvatura baseada em área e que seja afim-invariante.

Algumas considerações iniciais devem ser feitas. Seja inicialmente a Figura 5.3.

Uma vez que tencionamos usar área para definir curvatura e por sua vez o termo de suavizacão E_{suav} em (5.24) devemos estar conscientes que minimizar este termo de energia implicará em minimizar área e portanto devemos nos precaver quanto às situações na Figura 5.3.(c). Felizmente, esta possibilidades ficam descartadas uma vez que o efeito dos outros termos de energia e da direção de busca mostrada na Figura 5.3.(a) impedem que a mesma ocorra (Ip & Shen, 1998).

Baseados nestas idéias o termo de suavização será definido por:

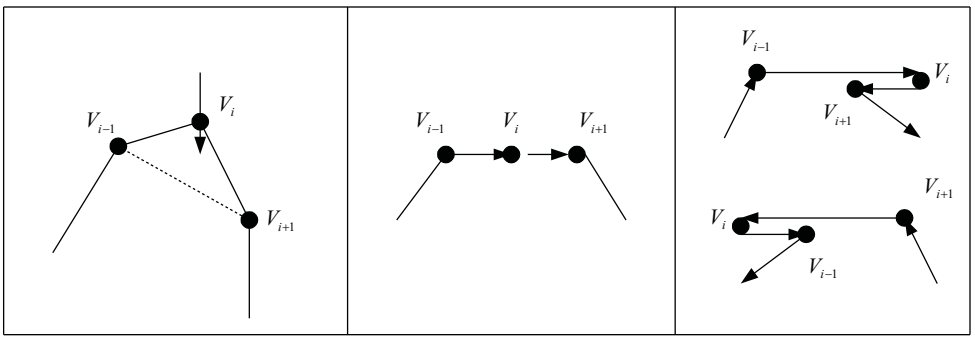

Figura 5.3: Configurações possíveis para vértices vizinhos.

$$E_{suav} = \sum_{i=0}^{N-1} e_{suav}\left(v_i\right), \tag{5.25}$$

onde:

$$e_{suav}\left(v_i\right) = \alpha \frac{|area_i|}{AREA_{now}} \frac{AREA_{modelo}}{\frac{1}{2}d^2}, \tag{5.26}$$

sendo α um parâmetro, $AREA_{now}$ a área da snake na presente interação, a qual pode ser calculada por:

$$AREA_{now} = \frac{1}{2} \sum_{i=0}^{N-1} \begin{vmatrix} u_i & u_{i+1} & u_{cent} \\ v_i & v_{i+1} & v_{cent} \\ 1 & 1 & 1 \end{vmatrix}$$

onde (u_{cent}, v_{cent}) são as coordenadas do centróide da snake dadas por:

$$\begin{pmatrix} u_{cent} \\ v_{cent} \end{pmatrix} = \frac{1}{N} \sum_{i=0}^{N-1} \begin{pmatrix} u_i \\ v_i \end{pmatrix}. \tag{5.27}$$

$AREA_{modelo}$ é a area do protótipo (modelo) e d é uma constante representando a média das distâncias entre os snaxels no protótipo. Pelo resultado (5.23) segue diretamente que E_{suav} é afim-invariante.

O termo E_{modelo} em (5.24) é definido para garantir a fidelidade ao protótipo em questão. Este protótipo, denominado $c_{protótipo}$, é definido por uma poligonal fechada dada pelo conjunto de pontos $\{W_i = (x_i, y_i) \,|i = 0, ..., N-1\}$.

A snake c_{snake} pode ser vista como uma versão deformada e afim-transformada do protótipo $c_{protótipo}$. Uma vez determinado a correspondência entre os pontos de

$c_{protótipo}$ e os pontos de c_{snake} é possível comparar o protótipo transformado com a snake atual (Figura 5.4 a seguir).

O protótipo transformado é denominado $c_{alinhado}$ e é obtido pela expressão:

$$\begin{pmatrix} x_i^{alinhado} \\ y_i^{alinhado} \end{pmatrix} = A \begin{pmatrix} x_i \\ y_i \end{pmatrix}, \tag{5.28}$$

onde a matriz A representa uma transformação afim.

A idéia central é obter uma energia de modelo que aumente caso a *forma* da snake se afaste daquela do protótipo transformado segundo a expressão (5.28).

Uma vez que a área de um triângulo pode ser convenientemente normalizada para torna-se afim-invariante define-se a energia E_{modelo} baseada também em área.

Da Figura 5.4 nós vemos que o processo de tornar V_i próximo de $W_i^{alinhado}$ é equivalente a fazer as áreas totais dos triângulos definidos pelos pontos $\left(W_{i-1}^{alinhado}, V_i, W_i^{alinhado}\right)$ e $\left(W_i^{alinhado}, V_i, W_{i+1}^{alinhado}\right)$ próximo de zero. A energia E_{modelo} pode portanto ser definida por:

$$E_{modelo} = \sum_{i=0}^{N-1} e_{modelo}\left(V_i\right), \tag{5.29}$$

tal que:

$$e_{modelo}\left(V_i\right) = \beta \frac{\left|S_{W_{i-1}^{alinhado}, V_i, W_i^{alinhado}}\right| + \left|S_{W_i^{alinhado}, V_i, W_{i+1}^{alinhado}}\right|}{AREA_{now}} \frac{AREA_{modelo}}{d^2}, \tag{5.30}$$

onde $S_{W_{i-1}^{alinhado}, V_i, W_i^{alinhado}}$ e $S_{W_i^{alinhado}, V_i, W_{i+1}^{alinhado}}$ são as areas dos triângulos definidos pelos pontos $\left(W_{i-1}^{alinhado}, V_i, W_i^{alinhado}\right)$ e $\left(W_i^{alinhado}, V_i, W_{i+1}^{alinhado}\right)$ respectivamente e β é um parâmetro de suavização (o valor para d e $AREA_{modelo}$ foram definidos acima).

Energia Externa

Uma vez que nos propomos manter uma certa fidelidade entre a snake e o protótipo devemos definir a energia externa em conformidade com esta proposta. Isto pode ser feito usando-se informações direcionais do gradiente. Para cada snaxel, a energia externa será definida pelo produto interno do vetor normal à snake com a direção do gradiente, pesado pela magnitude do gradiente em questão. Assim teremos:

$$E_{ext} = \sum_{i=0}^{N-1} e_{ext}\left(v_i\right), \tag{5.31}$$

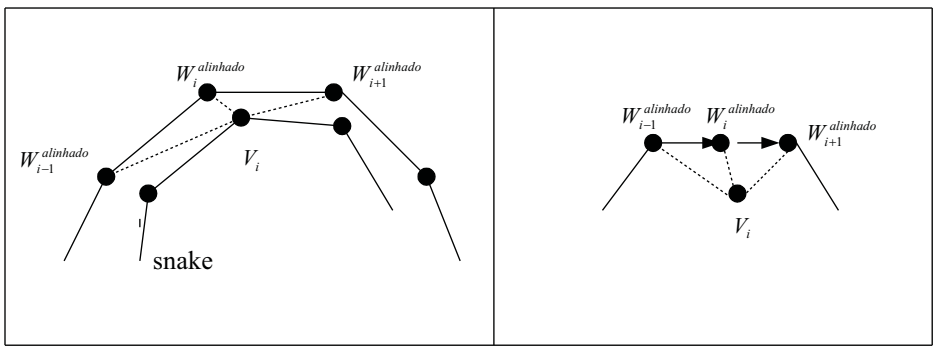

Figura 5.4: Comparação entre a snake e o protótipo alinhado.

tal que:

$$e_{ext}\left(v_i\right) = \gamma\left(1 - \left|\nabla I\left(v_i\right)\right| \left|n\left(v_i\right) \cdot h\left(v_i\right)\right|\right), \tag{5.32}$$

onde $n\left(v_i\right)$ é a normal à snake no ponto v_i tomada com sentido para o interior da snake.

Energia Total

A energia total usada no AI-Snake é finalmente dada por:

$$\begin{aligned} E_{snake} \;=\; & \alpha \sum_{i=0}^{N-1} \frac{|area_i|}{AREA_{now}} \frac{AREA_{modelo}}{\frac{1}{2}d^2} + \\ & \beta \sum_{i=0}^{N-1} \frac{\left|S_{W_{i-1}^{alinhado}, v_i, W_i^{alinhado}}\right| + \left|S_{W_i^{alinhado}, v_i, W_{i+1}^{alinhado}}\right|}{AREA_{now}} \frac{AREA_{modelo}}{d^2} + \\ & \gamma \sum_{i=0}^{N-1} \left(1 - \left|\nabla I\left(v_i\right)\right| \left|n\left(v_i\right) \cdot h\left(v_i\right)\right|\right), \end{aligned} \tag{5.33}$$

onde α, β e γ são os parâmetros do modelo, a serem estabelecidos a priori.

O problema de correpondência entre a snake e protótipo é tratado em detalhes em (Ip & Shen, 1998). Não abordaremos este problema nesta apresentação.

Uma vez determinado esta correspondência, é imediato obter a transformação afim que melhor relaciona as duas curvas.

Sem perda de generalidade, vamos supor que o ponto $W_{i+j_0} = (u_{i+j_0}, v_{i+j_0})$ do protótipo corresponde ao ponto $v_i = (x_i, y_i)$ da snake. Então, procuremos uma matriz A que minimize o erro ε_A dado por:

$$\varepsilon_A = \|Q' - AQ\|^2 , \tag{5.34}$$

onde A é a matriz que representa a transformação afim procurada e:

$$Q = \begin{pmatrix} x_1 & x_2 & & x_N \\ y_1 & y_2 & & y_N \\ 1 & 1 & & 1 \end{pmatrix} , \tag{5.35}$$

$$Q' = \begin{pmatrix} u_{1+j_0} & u_{2+j_0} & & u_{N+j_0} \\ v_{1+j_0} & v_{2+j_0} & & v_{N+j_0} \\ 1 & 1 & & 1 \end{pmatrix} , \tag{5.36}$$

são as matrizes dos pontos da snake e do protótipo respectivamente. A solução de (5.34) fornece a transformação afim que melhor relaciona a snake e o protótipo segundo o critério de mínimos quadrados. Esta solução é dada por:

$$A = \left(Q'Q^T\right)\left(QQ^T\right)^{-1} .$$

5.2 Modelo Dual Baseado em Programação Dinâmica

Na seção 5.1.1 descrevemos um método Dual baseado na equação de evolução (5.12). Nesta seção discutiremos uma versão do método Dual desenvolvida em (Gunn, 1996b) a qual é baseada nos conceitos de programação dinâmica discreta apresentados na seção 4.1 do capítulo anterior.

Consideremos inicialmente uma situação como aquela da Figura 4.1 onde supomos que a borda de interesse está no interior da região limitada pelas duas curvas destacadas.

Procedendo exatamente como na seção 4.1 vamos montar o espaço de busca estabelecendo inicialmente uma correspondência entre N pontos em cada curva e em seguida conectando estes pontos com segmentos de retas discretizados em M pontos (Figura 4.2). O espaço de busca final contém NM pontos.

O algoritmo de busca que passamos a descrever é denominado Viterbi (Gunn, 1996b; Bamford & Lovell, 1995). Este problema tem as características dos *problemas de decisão finitos em multi − estágio* abordados na seção 4.1. A única dificuldade é que procuramos agora contornos fechados e não abertos como foi o caso da discussão no capítulo anterior.

Para lidar com contornos fechados a solução adotada é usar duas buscas (Gunn, 1996b). Na primeira, escolhem-se dois segmentos adjacentes no espaço de busca e

então faz-se uma busca (equação (4.8)) para obter o contorno aberto ótimo para estes segmentos (Figura 5.5.(a)).

Em seguida, escolhem-se dois pontos no meio da primeira solução para serem os pontos inicial e final para a próxima busca. Aplica-se a expressão recursiva (4.8) novamente para então obter a solução final (Figura 5.5.(b)).

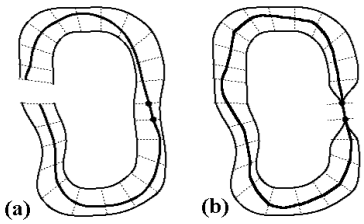

Figura 5.5: Programação Dinâmica para contornos fechados.

O funcional de energia para os contornos abertos é do tipo daquele apresentado em (4.2) e tem a seguinte forma (Gunn, 1996b):

$$E_{snake} = \sum_{i=0}^{N-3} E_i \tag{5.37}$$

onde:

$$E_i = \alpha E_{int}\left(v_i, v_{i+1}, v_{i+2}\right) + \beta E_{ext}\left(v_{i+1}\right) + \lambda E_{linha}\left(v_{i+1}\right), \tag{5.38}$$

com E_{int}, E_{ext} e E_{linha} dados por:

$$E_{int}\left(v_i, v_{i+1}, v_{i+2}\right) = \left(\frac{v_{i+2} - 2v_{i+1} + v_i}{\left\|v_{i+2} - v_i\right\|}\right)^2, \tag{5.39}$$

$$E_{ext}\left(v_i\right) = -\left\|\nabla I\left(v_i\right)\right\|^2, \tag{5.40}$$

$$E_{linha}\left(v_{i+1}\right) = \pm I\left(v_{i+1}\right). \tag{5.41}$$

Como já dito acima, a relação de recorrência correspondente a ser usada tem a forma clássica dada por (4.8).

5.3 Métodos envolvendo mudanças Topológicas

Analisaremos nesta seção dois modelo alternativo que têm a finalidade de corrigir as limitações topológicas do modelo original.

5.3.1 Método dos Conjuntos de Níveis

A idéia central nesta formulação, chamada *Conjunto de Níveis (Level Sets)* (Malladi *et al.* , 1995), é representar a snake como o conjunto de nível $\{x \in \Re^2 | G(x) = 0\}$ de uma função $G : \Re^2 \to \Re$.

Esta idéia tem sua origem nos trabalhos envolvendo propagação de interfaces em física de materiais onde uma frente (modificação do meio) se desloca alterando o mesmo (Sethian, 1988). Por este motivo a terminologia usual desta formulação (implícita) para contornos ativos usa o termo *frente* em lugar de snake.

Sejam então uma função G, a qual assumiremos agora dependente do espaço e também do tempo:

$$G : \Re^2 \times \Re^+ \to \Re, \tag{5.42}$$

tal que a frente em questão, no instante $t = 0$ seja dada por uma curva c:

$$c(s, t = 0) = (x | G(x, t = 0) = 0). \tag{5.43}$$

O próximo passo será encontra a partir destes elementos uma formulação Euleriana para o movimento da frente c.

Seguindo o trabalho de Setian (Malladi *et al.* , 1995) vamos supor que a frente se propaga ao longo da sua direção normal com velocidade $\overrightarrow{F}$, onde $\overrightarrow{F}$ pode ser uma função de vários elementos incluindo curvatura, direção normal, etc.

A Figura 5.6 ilustra o problema para o caso onde a função G tem a forma inicial de um parabolóide e a curva definida pela expressão (5.43) tem a forma de uma circunferência.

Nestas condições, necessitamos de uma equação para a evolução da função $G(x, t)$ a qual contem o movimento de $c(s, t)$ como o conjunto de nível dado por:

$$c(s, t) = (x | G(x, t) = 0). \tag{5.44}$$

Seja então $x(t)$, $t \in \Re^+$ a trajetória de um ponto da frente que se propaga. Isto é, $x(t = 0)$ é um ponto na frente inicial $c(s, t = 0)$ e $x_t = \overrightarrow{F}(x, t)$ com o vetor x_t normal à frente no ponto $x(t)$. Uma vez que a função G é sempre zero sobre a frente, nós devemos ter:

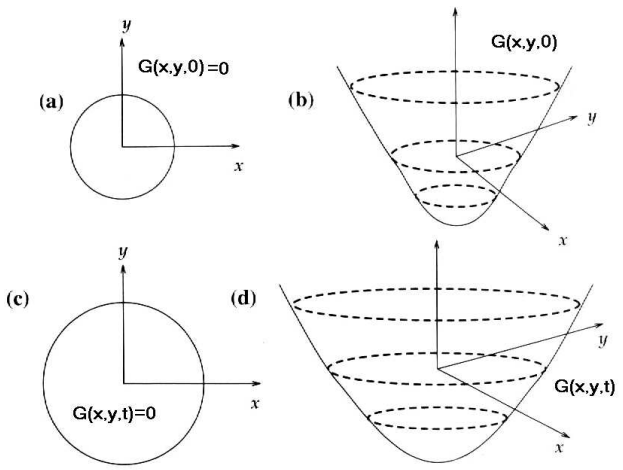

Figura 5.6: Método dos Conjuntos de Níveis.

$$G\left(x\left(t\right),t\right)=0. \tag{5.45}$$

Supondo G diferenciável tendo zero como um valor regular, o Teorema da Função Implícita nos permite derivar (5.45) com relação ao tempo para obter:

$$G_t+\sum_{i=1}^{N}G_{x_i}x_{i_t}=0, \tag{5.46}$$

onde x_i é a ith componente de x. Lembrando que o somatório na última expressão é exatamente o produto escalar entre o gradiente (com relacão a x) de G e o vetor velocidade podemos escrever:

$$\sum_{i=1}^{N}G_{x_i}x_{i_t}=\nabla G\cdot\overrightarrow{F}=\left\|\nabla G\right\|\left\|\overrightarrow{F}\right\|, \tag{5.47}$$

obtemos finalmente a equação procurada:

$$G_t+F\left|\nabla G\right|=0, \tag{5.48}$$

onde $F=\left\|\overrightarrow{F}\right\|$.

A equação (5.48) tem como condição inicial a função $G\left(x,t=0\right)$ a qual deve ser definida a priori. Para certas escolhas da função de velocidade F esta equação

assume a forma de uma equação de Hamilton-Jacobi, bem conhecidas em mecânica clássica (Goldstein, 1981).

O condição inicial $G(x, t = 0)$ pode ser definida naturalmente pela expressão

$$G(x, t = 0) = \pm d, \tag{5.49}$$

onde d é a distância de x a $c(s, t = 0)$, e o sinal indica se o ponto é interior (-) ou exterior (+) à curva inicial.

Existem quatro vantagens desta formulação Euleriana baseada na equação de Hamilton-Jacobi acima. A primeira provém do fato de que a funcão $G(x, t)$ sempre permanece uma função diferenciável desde que F seja suave (Malladi *et al.* , 1995). Contudo, o conjunto de nível $\{G(x, t) = 0\}$, e portanto a frente $c(s, t)$ pode mudar a topologia, dividindo-se ou agregando-se, durante a evolução da função $G(x, t)$ (Malladi *et al.* , 1995), incorporando operações topológicas naturalmente.

A segunda vantagem diz respeito à análise numérica. Se $G(x, t)$ permanece uma função diferenciável durante sua evolução, nós podemos usar diferenças finitas para discretizar a equação (5.48) tanto no tempo quanto no espaço.

Contudo, a técnica correta para aproximar as derivadas espaciais na equação (5.48) vem de uma condição de entropia apropriada para frentes propagando-se, a qual é discutida em detalhes em (Sethian, 1996). Uma condição de entropia comumente usada é a seguinte: *uma vez que um ponto foi queimado, ele permanece queimado*; isto é, se um ponto se tornou interior à frente então ele permanecerá interior durante toda a evolução da mesma (Malladi *et al.* , 1995).

Para obter um esquema numérico para discretizar a equação (5.48) que respeite esta condição de entropia usa-se técnicas em *leis de conservação hiperbólicas*. Em (Sethian, 1996) encontramos uma discussão detalhada a respeito.

A terceira vantagem desta formulação implícita é que a geometria intrínseca da frente pode ser facilmente determinada a partir da função G. Por exemplo o vetor normal à frente e a curvatura da mesma são facilmente obtidos pelas expressões:

$$\overrightarrow{n} = \nabla G,$$

$$K = \nabla \cdot \left(\frac{\nabla G}{\|\nabla G\|} \right),$$

respectivamente, onde o gradiente e o operador divergente $(\nabla \cdot)$ são todos calculados em relação a x.

Por fim a última vantagem é que não existe diferença significativa em seguir frentes em 2D ou 3D. Para isto basta extender as estruturas de dados e os operadores

diferenciais. Assim, superfícies em propagação são facilmente formuladas através de uma equação análoga a (5.48).

Extração de Bordas

Nesta seção, nós descrevemos como a formulação acima pode ser usada para extração de bordas.

Primeiramente, notemos que a frente é uma curva que evolui no tempo. Uma vez que a idéia é extrair bordas em uma dada imagem, a frente deve ser forçada a parar na vizinhança da borda desejada, analogamente ao que apresentamos para as snakes paramétricas já discutidas. Assim, o problema está em como definir adequadamente a função velocidade F na equação (5.48). Em (Sethian, 1996) encontramos uma discussão detalhada a respeito.

Uma possibilidade interessante para F é discutida em (Sapiro, 1997) sendo dada pela expressão a seguir:

$$\begin{aligned} \frac{\partial G}{\partial t} &= \|\nabla G\| \ \nabla \cdot \left(g_{cinza}\left(I\right) \frac{\nabla G}{\|\nabla G\|} \right) \\ &= g_{cinza}\left(I\right) \|\nabla G\| \ \nabla \cdot \left(\frac{\nabla G}{\|\nabla G\|} \right) + \nabla g_{cinza}\left(I\right) \cdot \nabla G, \end{aligned} \tag{5.50}$$

onde $g_{cinza}\left(r\right) \to 0$ se $r \to +\infty$.

Nesta expressão, o segundo termo da segunda igualdade dirige a frente para as fronteiras dos objetos, isto é, para os vales de g_{cinza}. O primeiro termo incorpora explicitamente a curvatura (veja equação 5.3.1) tendo o papel de suavizar o resultado final (Malladi *et al.* , 1995; Sethian, 1996).

5.3.2 T-Snakes

O método dos conjuntos de níveis da seção anterior tem em sua base teórica elementos de geometria e topologia diferencial (geodésicas, Teorema da Função Implícita, etc).

O método das T-Snakes (McInerney & Terzopoulos, 1995; McInerney, 1997) que passaremos a apresentar agora aborda as questões topológicas num outro contexto. A idéia fundamental é converter o problema para o contexto da Topologia Combinatória a fim de obter uma formulação paramétrica com habilidades topológicas para as snakes.

No caso dos conjuntos de níveis inicializamos o método definindo a função G a qual tem a snake inicial como conjunto de nível e zero como um valor regular.

No presente caso, vamos construir uma função linear por partes χ definida sobre uma triangulação do plano e cuja solução para $\chi\left(x\right) = 0$ corresponda à snake.

Tendo como base o Princípio Door-In-Door-Out da seção 4.3, poderemos aplicar o algoritmo do final da seção 4.3.2 (Algoritmo Door-In-Door-Out) para obter cada uma das componentes conexas da snake resolvendo assim as mudanças topológicas (splits e merges).

A função linear por partes em questão é denominada *função caracterírstica* em (McInerney & Terzopoulos, 1995; McInerney, 1997) podendo ser definida por:

$$\chi : D \subset \Re^2 \rightarrow \{0, 1\} \tag{5.51}$$

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

Para construir esta função o método usado consiste primeiramente em projetar as curvas (snakes) sobre a triangulação usada como mostra a Figura 5.7 a seguir.

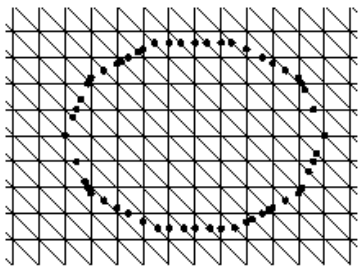

Figura 5.7: Snake projetada sobre uma triangulação do tipo Coxeter-Freudenthal.

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 scan line (Rogers, 1985). A Figura 5.8 é um exemplo de como a função χ e o algoritmo da seção 4.3 fornecem naturalmente o merge entre duas curvas.

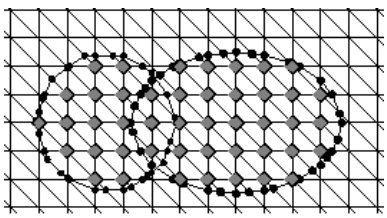

Figura 5.8: Merge entre duas curvas. Nós marcados são aqueles onde χ tem valor 1.

Nesta figura, os nós marcados são aqueles onde a função característica tem valor 1 e os demais onde esta função tem valor 0. Aplicando o algoritmo Door-In-Door-Out, obteremos ao final a seqüência 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.

Observe que não precisamos determinar precisamente a solução de $\chi\,(x) = 0$. O que precisamos é a segurança de que a seqüencia aresta-triângulo-aresta obtida pelo algoritmo é cíclica e finita, e assim, que o método é consistente. Estas propriedades foram apresentadas no final da seção 4.3 e são fundamentais para o método das T-Snakes.

Além da triangulação do espaço (no caso, o $\Re^2$), da projeção da snake e da função característica, um outro componente do método das T-Snakes é um modelo discreto de contorno ativo, o qual passaremos a definir agora.

Modelo de Snake

A T-Snake, assim como o contorno ativo Dual descrito acima, é definida por um conjunto de N pontos (snaxels) cujas posições $\{v_i = (x_i, y_i)\,, i = 0, ..., N-1\}$, são conectadas em seqüência formando uma curva fechada.

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

Os pontos da snake são supostos 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 a expansão ou compressão somente quando a distância entre snaxels $\|r(t)\| = \|v_{i+1} - v_i\|$ é maior ou menor que l_i, respectivamente.

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

$$\alpha_i = a_i e_i r_i(t) - a_{i-1} e_{i-1} r_{i-1}(t). \tag{5.52}$$

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 $t - \Delta t$. Isto fornece ao modelo o comportamento de um material com viscoelasticidade não nula (McInerney, 1997).

Em adição à força (5.52) 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:

$$\beta_i = b_i \left(v_i - \frac{1}{2}\left(v_{i-1} + v_{i+1}\right) \right), \tag{5.53}$$

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:

$$F_i = sig\,(v_i)\, k n_i, \tag{5.54}$$

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 expressão 5.54, n_i é a normal ao vértice v_i e k é fator que define a intensidade da força. Esta é uma força tipo balloon (seção 3.4), 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 $sig\,(v_i)$:

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

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

As forças dadas (5.52)-(5.54) são forças internas.

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:

$$P = -\left\|\nabla I\right\|^2, \tag{5.55}$$

ou de uma normalização do tipo daquela usada no Balloon (seção 3.4):

$$\begin{aligned} f &= 0 \text{ se } \nabla P < T_F, \\ f &= \lambda \frac{\nabla P}{|\nabla P|} \text{ se } \nabla P > T_F, \end{aligned}$$

onde T_F é um limiar previamente estabelecido e λ é um fator de escala para a força.

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

$$v_i^{(t+\Delta t)} = v_i^t + \frac{\Delta t}{\gamma}\left(\alpha_i^t + \beta_i^t + F_i^t + f_i^t\right), \tag{5.56}$$

Durante a evolução da T-Snake alguns nós da malha tornam-se interiores à mesma (supondo expansão). Estes nós são denominados *nós queimados*, em analogia ao método dos conjuntos de níveis da seção 5.3.1, o qual inspirou o presente método (McInerney, 1997).

Para evitar os problemas conhecidos para evolução de curvas na direção normal (desenvolvimento singularidades) o método das T-Snakes adota uma condição de *entropia* do tipo da usada no método dos conjuntos de níveis (McInerney, 1997): *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 (daí o nome entropia). Neste sentido, primeiramente define-se uma *temperatura* para cada snaxel a qual é o número de deformações que o triângulo em que o elemento de 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. No entanto, esta condição limita o movimento da snake o que pode trazer dificuldades como veremos a seguir.

Uma vez que a T-Snake atingiu o equilíbrio, a malha correspondente à divisão simplicial pode ser descartada e a snake evoluir como uma snake discreta segundo a equação (5.56)

Atualização da Função Característica

Ao final de cada passo de evolução da snake, devemos atualizar a função característica definida em (5.51). Isto é equivalente a determinar o conjunto de nós da malha que foram queimados durante a evolução da snake. Passaremos agora a descrever o algoritmo correspondente (McInerney, 1997).

Cada elemento de modelo (par de snaxels consecutivos) pode ter passado sobre nenhum apenas um ou vários nós durante sua evolução. Inicialmente, para cada elemento de modelo nós formamos um polígono usando a atual posição e a posição anterior. Este polígono permite facilmente determinar os nós que podem ter sido queimados (Figura 5.9).

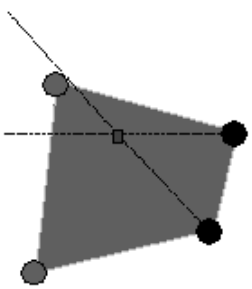

Figura 5.9: Polígono contendo nó interior (queimado).

Para cada nó no interior do polígono nós particionamos a imagem em 4 subes-

paços através das semi-retas indicadas na Figura 5.10:

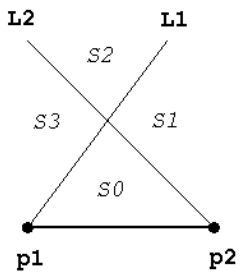

Figura 5.10: Subespaços usados para classificação de nós.

Estas linhas são denominadas $L1$ e $L2$ na figura acima 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 5.10. Portanto, temos um total de 16 possibilidades a considerar mostradas nas Figuras 5.11.(1)-5.11.(16).

Os quatro pontos p_1, p_2, p_{1n} e p_{2n} formam o polígono fechado Q que pode ser convexo ou não (Figura 5.11). Vejamos duas definições importantes (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 *aresta 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..

De acordo com estas definições, vemos que os casos (1), (2), (7) da Figuras 5.11 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 5.10 fornece 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

Até que o ponto de congelamento seja atingido:

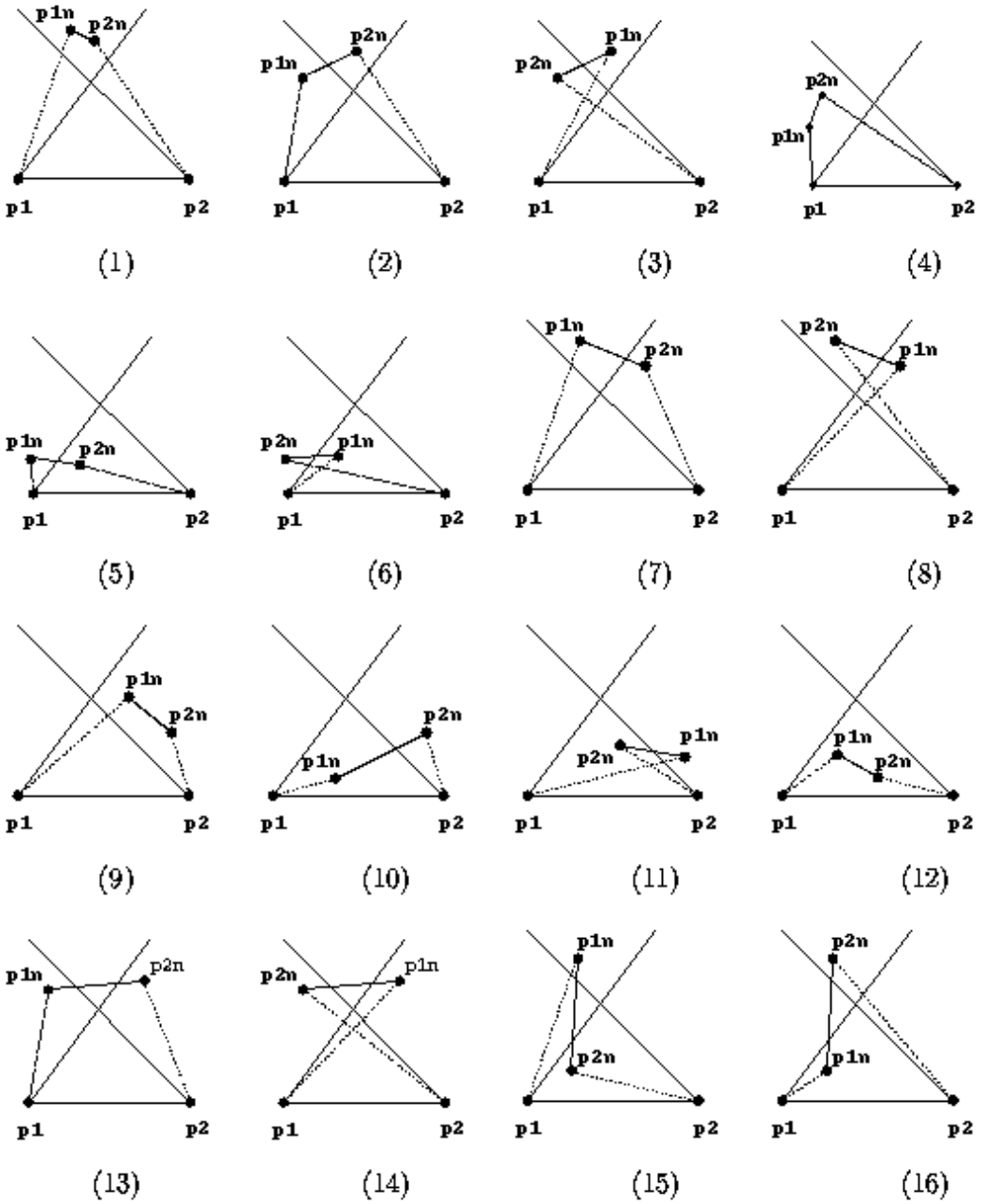

Figura 5.11: Possíveis casos para classificação.

1. Computar as forças internas e externas e atualizar a posição dos snaxels de acordo com a equação (5.56).

2. Computar as interseções entre a malha e os elementos de modelo.

3. Atualizar a função característica definida pela equação (5.51).

4. Usando a função característica determinar o correspondente conjunto de triângulos de borda usando o algoritmo Door-In-Door-Out.

5. Para cada aresta da seqüencia de arestas obtida escolher um ponto dentre os calculados no ítem 2 para ser um novo snaxel. Se não houver nenhum, criar um ponto na aresta em questão. Descartar todas as outras interseções.

6. Atualizar temperatura dos snaxels.

Apesar de suas habilidade topológicas, o método das T-Snakes tem várias limitações. Uma vez que este método é central para nosso trabalho faremos a seguir uma discução destas limitações.

Limitações das T-Snakes

Em (McInerney, 1997) algumas limitações da T-Snake são discutidas e algumas soluções 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 expanção ou apenas contração, mas não ambos, o que pode dificultar a interação com o usuário caso esta se torne necessária.

Uma possível solução para este problema seria reverter a direção de evolução ocasionalmente 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 pudecemos saber automaticamente onde a malha necessita ser refinada, o que na prática é difícil. Na seção 7.5 indicaremos uma possível solução para este problema baseada em uma metodologia que parte de uma malha mais grossa, refinando-a até a resolução desejada.

Outra limitação da T-Snake relacionada com nosso trabalho é o fato do método de reparametrização da snake (projeção sobre a malha) ser extrínseco, isto é, 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 adotamos 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.

Outro aspecto relacionado à decomposição do domínio se refere ao viés, na direção da diagonal secundária, de uma triangulação como a da Figura 5.7 . Isto provém da observação de que a snake terá menos snaxels em regiões aproximadamente paralelas à diagonal, o que vai se refletir nas forças internas correspondentes (expressões (5.52)-(5.54)).

Uma forma de evitar este problema seria usar uma triangulação do tipo J_1 como na Figura 5.12 abaixo.

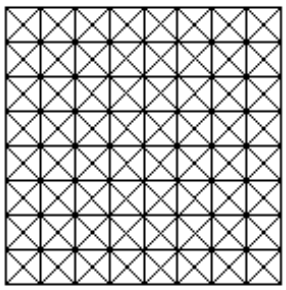

Figura 5.12: Triangulação do tipo J_1.