

Segmentação de Dados Tridimensionais Baseada em Crescimento por Volume

DENISE GULIATO

Escola Politécnica da Universidade de São Paulo
LSI - Laboratório de Sistemas Integráveis
Div. de Sist. Digitais - Grupo de Computação Gráfica
Av. Prof. Luciano Gualberto, travessa 3, n.158
05508-900 - São Paulo, SP, Brasil
guliato@lsi.usp.br

Abstract. This paper focuses the topic of segmentation of medical images, propose a new recursive algorithm based on volume growing and present its results.

1. Segmentação de Imagens

A idéia da segmentação tem sua origem na teoria Gestalt, que estuda a preferência dos seres humanos em agrupar ou organizar conjuntos de formas encontrados em um determinado campo visual. Os princípios de Gestalt mostram certas preferências de agrupamento, geralmente baseadas em características de proximidade, similaridade e descontinuidade [3]. O agrupamento de partes homogêneas de uma imagem com relação a uma ou mais destas características resulta numa imagem segmentada [2].

Os métodos de segmentação, geralmente, são baseados nas propriedades dos valores dos níveis de cinza e podem ser divididos em duas categorias: *métodos de segmentação baseados em descontinuidade* e *métodos de segmentação baseados em similaridade*.

Nos métodos baseados em descontinuidade, a segmentação é baseada nas mudanças abruptas dos níveis de cinza. Nos métodos baseados em similaridade a segmentação baseia-se na homogeneidade dos níveis de cinza pertencentes ao objeto de interesse. A homogeneidade varia de acordo com um critério previamente estabelecido [1,2].

Até agora, a maior concentração de esforços tem sido no desenvolvimento de técnicas para segmentar imagens 2D. No entanto, as novas técnicas de aquisição de dados, em particular, para diagnóstico médico, têm exigido métodos que operem diretamente sobre volume. Alguns trabalhos têm sido apresentados para segmentar volumes, baseando-se na mudança abrupta dos níveis de cinza [4, 5, 6, 7, 8, 9, 10]. Já os segmentadores de volume baseados em similaridade têm sido menos explorados, dado, provavelmente, à dificuldade de estabelecer critérios de aceitação gerais e independente do método de aquisição [11].

Este trabalho propõe um algoritmo recursivo para segmentar dados 3D, obtidos por tomografias computadorizadas. O método é denominado *crescimento por volume*, sendo uma extensão direta do método de segmentação baseado em crescimento por região [1, 2]. O critério de aceitação usado é a diferença de níveis de cinza entre voxels 26-conectados.

2. Algoritmo Recursivo para Segmentação baseada em Crescimento por Volume

A filosofia para segmentar volumes é a mesma que para segmentar imagens, exceto pelas implicações resultantes da inclusão de uma terceira dimensão. O volume a ser segmentado é representado por uma matriz 3D, onde cada elemento da matriz $f(i,j,k)$ é referido como um voxel (elemento do volume). A cada voxel é associado um nível de cinza cujo valor representa uma certa propriedade do modelo real.

Uma maneira de se obter a segmentação 3D, é estender a técnica convencional de crescimento por região que segmenta dados 2D. Neste caso continuam válidas as restrições que o método impõe para o caso bidimensional. Nesta seção descreveremos uma implementação para o método crescimento por volume usando técnicas de recursão. Antes, introduziremos o conceito de *máscara de visualização*, recurso usado para a implementação do método de segmentação proposto.

2.1. Máscara de Visualização

O método de segmentação aqui proposto utiliza um recurso que chamaremos *máscara de visualização*. Essa máscara é uma matriz com a mesma dimensão do volume 3D original, que tem como objetivo determinar quais voxels da imagem estarão visíveis durante a etapa de visualização e quais estarão invisíveis. Por exemplo, o voxel da posição (i,j,k) será visível se a máscara de visualização na posição (i,j,k) for igual a 1 e invisível caso contrário. A máscara de visualização permite, ainda, alguma sofisticação, como por exemplo, ao invés de determinar apenas a visibilidade ou não de um voxel, ela pode associar a visibilidade a um valor referente a uma entrada numa tabela de cores, atribuindo, assim, uma cor ao voxel visível.

A execução de uma operação lógica AND entre a máscara de visualização e o volume de dados resulta num terceiro volume que representa a imagem segmentada. Essa operação pode ser executada antes ou durante a etapa de visualização. No último caso há um ganho de tempo e de espaço de armazenamento.

2.2. Descrição do Algoritmo Recursivo para a Segmentação baseada em Crescimento por Volume

2.2.1. Entrada e Saída de Dados

Neste método, o resultado da segmentação é um volume 26-conectado cujos voxels satisfazem a condição de aceitação, dada pela diferença de intensidade entre voxels adjacentes.

O algoritmo tem como entrada de dados:

- a. a imagem tridimensional representada por uma matriz 3D
- b. a posição do voxel semente
- c. a condição de aceitação
- d. o número que representa uma entrada na tabela de cores. (para simplificar, usaremos o valor 1 para indicar que o voxel pertence ao segmento e 0 caso contrário).

Como saída, o algoritmo gera a máscara de visualização, com a mesma resolução espacial da matriz de entrada e com resolução de profundidade de 8 bits.

2.2.2. Descrição do Algoritmo

A seguir descreveremos, sem detalhes de implementação, o algoritmo para o procedimento recursivo de segmentação baseado em crescimento por volume.

a. Programa Principal

```
main ()
{
  inicializa máscara;
  inicializa variáveis;
  lê volume;
  lê voxel_semente;
  lê condição_de_aceitação;
  cresce_volume(volume, máscara,
voxel_semente,0,1,condição_de_aceitação);
}
```

b. Procedimento cresce_volume

```
cresce_volume(volume, máscara, voxel, 0, 1, condição)
{
  máscara[voxel] = 1;
  valor_intensidade_voxel = volume[voxel];
  para todos os 26 vizinhos do voxel faça
  {
    vizinho = determina_vizinho(voxel);
    se (vizinho é um voxel válido)
    então
    {
```

```
      valor_intensidade_vizinho =
volume[vizinho];
      se (abs(valor_intensidade_voxel -
valor_intensidade_vizinho)
<= condição)
      então cresce_volume(volume, máscara,
voxel, 0, 1, condição)
      senão máscara[vizinho] = borda;
    }
  }
}
```

A máscara de visualização é criada e inicializada com um valor NÃO_EXAMINADO, de tal forma a impedir, que, durante o processo de recursão, voxels sejam analisados mais de uma vez.

É condição básica que o voxel_semente pertença ao volume de interesse, portanto é atribuída à máscara de visualização, na posição correspondente ao voxel_semente, a cor do segmento. O próximo passo é analisar um dos vizinhos 26-conectados ao voxel_semente.

Para cada um dos 26 vizinhos do voxel_corrente, o algoritmo verificará:

- a. O vizinho está dentro dos limites do volume 3D?
- b. O vizinho não foi analisado em uma etapa anterior do algoritmo?
- c. A condição de aceitação foi satisfeita?

Se as três condições acima forem verdadeiras significa que o vizinho é um voxel que pertence ao volume de interesse. Neste momento o algoritmo entra num procedimento de recursão. O voxel corrente é empilhado, o voxel vizinho é incorporado ao volume de interesse e passa a ser tratado como o voxel corrente e cada um dos seus 26 vizinhos é analisado. O processo recursivo pára quando nenhum voxel vizinho válido satisfizer à condição de aceitação. Se um vizinho não satisfaz à condição de aceitação, significa que ele representa o limite entre o volume de interesse e o resto da imagem. Portanto podemos rotular esse voxel como sendo uma *borda*.

Conforme vimos, a condição de aceitação do segmentador apresentado é baseado na diferença de intensidade entre voxels vizinhos. Pode acontecer, especialmente para grandes estruturas, que, embora a diferença entre voxels vizinhos satisfaça à condição de aceitação, a diferença entre o voxel sendo analisado e o voxel semente seja bastante significativa. Numa inspeção visual, auxiliada pelo histograma do volume, pode ficar óbvio que o voxel analisado não pertence ao volume de interesse. Uma maneira de resolver este problema é aplicar, à imagem, um algoritmo de limiarização 3D (veja seção 2.3), de tal maneira a eliminar previamente da imagem estruturas facilmente

identificadas como não pertencentes ao volume de interesse, veja figura 1.b.

2.3. Limiarização 3D

A limiarização 3D é uma extensão direta da Limiarização 2D. A partir do histograma do volume de dados, o usuário seleciona o intervalo de "threshold". A máscara de visualização é alocada e adequadamente inicializada. Todos os voxels da imagem que pertencerem ao intervalo dado serão considerados visíveis e à sua posição correspondente na máscara de visualização será atribuído o valor 1, caso contrário, o valor 0.

Algumas modificações devem ser feitas no algoritmo apresentado na seção 3.2 para fazer com que a máscara de visualização, contendo informações da Limiarização 3D, seja entrada de dados para o segmentador. Agora, somente os voxels do volume original que tiverem seu correspondente, na máscara de visualização, com uma indicação de visibilidade, serão analisados, não mais todo o volume original de dados como no algoritmo apresentado. O resultado da aplicação da limiarização 3D pode ser visto na figura 1.b.

3. Resultados

O resultado da aplicação do algoritmo proposto sobre uma imagem obtida por Tomografia Computadorizada por raio X pode ser vista nas figuras 1.a, 1.b, 1.c.

A figura 1.a apresenta a imagem a ser segmentada, cuja estrutura de interesse é o coração. A figura 1.b mostra o resultado da aplicação do algoritmo de limiarização 3D, eliminando da imagem todos os voxels cujos valores, sabia-se por análise do histograma, não pertencerem à estrutura de interesse. Nota-se que o resultado é o coração com algum artefato. Isto significa que os voxels, tanto da estrutura de interesse quanto do artefato, possuem níveis de cinza dentro do mesmo intervalo de limiarização. Aplicando-se sobre esta imagem o algoritmo de segmentação aqui proposto, usando como condição de aceitação uma diferença entre valores de voxels 26-conectados igual a 30, o resultado obtido é o apresentado na figura 1.c.

4. Conclusões e Trabalhos Futuros

O segmentador 3D baseado em crescimento por volume é bastante eficiente na exploração espacial existente entre as diversas estruturas de um volume. Uma restrição ao método implementado de forma recursiva é a exigência de plataformas poderosas, tanto a nível de processamento, quanto de armazenamento.

Apesar dos resultados satisfatórios, o critério de aceitação, conforme determinado, não é adequado para volumes de dados muito complexos. Estamos estudando

critérios mais elaborados que possam servir para um conjunto mais extenso de dados.

Referências Bibliográficas

- [1] Gonzalez, R.C.: Digital Image Processing. Reading Mass, Addison Wesley 1987.
- [2] Ballard, D.H; Brown, C.M. Computer Vision. Prentice Hall, Inc. 1982.
- [3] Attneave, F. Some Informational aspects of visual perception: Psychological Review 61, 1954.
- [4] Lui, H.K. Two and Three Dimensional boundary Detection. Comput. Graphics Images Process. 6, 1977. pp 123-134
- [5] Zacker, S.W.; Hummel, R.A.. A Three Dimensional Edge Operator. IEEE Trans. Pattern Anal. Machine Intelligence. PAMI-3, 1981.
- [6] Hueckel, M.T. An operator which locates edges in digitized pictures. J. Assoc. Comput. Mach. 18, 1971. pp 113-125.
- [7] Hummel, R.A. Texture detection using basis function. Comput. Graphics Image Process. 9, 1979. pp 40-55.
- [8] Mongenthaler, M.; Rosenfeld, A.. Multidimensional edge detection by hypersurface fitting. IEEE Trans. Pattern Anal. Machine Intelligence. PAMI-3, 1987.
- [9] Canny, J.F. Finding Edges and Lines in Images. Technical Report TR 720, MIT, June 1983.
- [10] Monga, O.; Deriche, R. 3D Edge Detector Using Recursive Filtering: Application to Scanner Images. CVGIP: Image Understanding 53, 1, Jan. (1991), pp 76-87.
- [11] Liou, S.; Jain, R.C. An Approach to Three-Dimensional Image Segmentation. CVGIP: Image Understanding. 53, 3 May (1991) pp 237-252.

