A new algorithm of brain volume contours segmentation*

WU Jian-ming(吴建明)†, SHI Peng-fei(施鹏飞)

(*Institute of Image Processing and Pattern Recognition, Shanghai Jiaotong University, Shanghai 200030, China*)

†E-mail: wjm010@sjtu.edu.cn

Received July 6, 2002; revision accepted Sept. 29, 2002

Abstract: This paper explores brain CT slices segmentation technique and some related problems, including contours segmentation algorithms, edge detector, algorithm evaluation and experimental results. This article describes a method for contour-based segmentation of anatomical structures in 3D medical data sets. With this method, the user manually traces one or more 2D contours of an anatomical structure of interest on parallel planes arbitrarily cutting the data set. The experimental results showes the segmentation based on 3D brain volume and 2D CT slices. The main creative contributions in this paper are: (1) contours segmentation algorithm; (2) edge detector; (3) algorithm evaluation.

Key words: CT slices, Contours segmentation, Edge detector
Document code: A **CLC number:** TP 242.6

INTRODUCTION

Segmentation, a central issue of computer vision, is a fundamental processing step in most systems that support medical diagnosis or planning of surgical operations and radiation treatments (Wyatt *et al.*, 2000; Winterer *et al.*, 2002; Bao *et al.*, 1998).

Contour-based segmentation (Marcondes *et al.*, 1999; Masato *et al.*, 1996; Luca Foresti *et al.*, 1998) methods are generally computationally efficient, but when applied to real images, are often not robust and quite sensitive to noise and data variability. The reliability of such methods can be improved using techniques for segmentation and data representation based on data-driven elastic models, such as snakes and deformable surface models, which have been applied successfully also in the field of medical imaging.

In this paper, we introduce a 3D generalization of the approach for segmenting 3D object boundary (Shen *et al.*, 2001; Ge *et al.*, 2001) which further reduces the time spent by the user in segmentation. In a 2D slice, for two specified points (pixel vertices) on the boundary of the object, the best boundary segmentation is the minimum-cost path between the two points, described as a set of oriented pixel edges. A complete 2D boundary is identified as a set of consecutive boundary segmentations forming a "closed", "connected" contour. In the strategy of the 3D extension (Flyun, 1990; Feng, 2002), users first specify contours on a few slices that are orthogonal to the natural slices of the original scene. If these slices are selected strategically, then we have a sufficient number of points on the 3D boundary of the object to subsequently trace optimum boundary segments automatically in all natural slices of the 3D scene. A 3D boundary may define multiple 2D boundaries per slice.

To segment a slice, we apply the edge detector (Poli *et al.*, 1992) to the image. The edge detector is based on a one-dimensional second-order polynomial filter.

This paper explores brain image segmentation technique and some related problems, including contours segmentation algorithms, edge detector, algorithm evaluation and experimental results. The main creative contributions in this paper are: (1) contours segmentation algorithm; (2) edge detector; (3) algorithm evaluation.

* Project(No.69931010) supported by the National Natural Science Foundation of China

GENERAL THEORY OF IMAGE SEGMENTATION

Image segmentation divides an image into several parts according to unanimous theory so that every part of the image meets the unanimous theory, but the combination of every two adjacent parts of the image do not meet the unanimous theory. If I represents an image, P represents the logical properties in the combination of adjacent parts of the image. Divide an image I into N disconnecting parts: $R_1, R_2, \ldots, R_N$ which meet the following condition:

$$I = \bigcup_{i=1}^{N} R_i \tag{1}$$

$$R_i \cap R_j = \phi \quad \text{for } i \neq j \tag{2}$$

$$P(R_i) = \text{TRUE} \tag{3}$$

$$P(R_i \cup R_j) = \text{FALSE} \quad \text{for } i \neq j,\ R_i \text{ near } R_j \tag{4}$$

Segmentation is the process of classifying a given pixel as the background or the structure of interest. There are a large number of classification methods ranging from simple pixel classification based on intensity to region based techniques. Once classified, each region must be labeled with a unique number to allow interaction with individual structures. In order to take advantage of the regularity of the data, we devised the following sequence of steps to perform classification and labeling:

1. Detection and numbering local maxima
2. Boundary formation
3. Three dimensional connected component analysis

Step 1 and Step 2 are performed on individual two dimensional sections of the three dimensional image. Step 3 uses boundary data on all sections.

Local maxima detection is performed to determine the center of each replication site. A hill climbing algorithm is applied to equally distributed set of points which would find the local intensity maxima in the neighborhood of each point. By selecting every third pixel as a starting point, the algorithm is assured of finding all significant intensity maxima. A useful byproduct of this algorithm is that the peaks are labeled.

During the boundary formation step, each local maxima is taken as the center of the particle and intensity cross sections are obtained in several radial directions. For each radial, the boundary is taken as the point with the highest intensity gradient. The set of vertices obtained by this method is processed to give an 8-connected set of boundary points.

The final task of the segmentation process is to form three dimensional objects from individual contours for each section of the 3D image. A unique label for each contour is generated by taking the section number as the two MSB's and the contour number within the section as the two LSB's of a 32 bit number. During the first pass, labels of each contour in one section are propagated to all overlapping contours of the next section. If a contour inherits two or more labels from the previous section, the lesser of them is assigned to it and an equivalence relation is recorded among all the competing labels. During the second pass, each label is examined for any equivalent labels if one exists. During this process, various statistics gathered for each section are accumulated for each object.

CONTOURS SEGMENTATION ALGORITHM

We think of a 3D scene in a Cartesian coordinate system (x, y, z) as a set $C^{(m)} = \{C_1, C_2, \cdots, C_m\}$ of m consecutive and parallel 2D scenes C_i in the xy-plane, called natural slices, for each fixed z coordinate. Given a natural slice C_i, $1 \leqslant i \leqslant m$, of $C^{(m)}$, the boundary of the 3D object in C_i may be represented by a set of 2D boundaries.

In each natural slice, the 3D object may be represented by multiple connected 2D boundaries. To trace these 2D boundaries using the points which fall in a given natural slice, we should be able to identify the set of points that belong to each connected 2D boundary in this natural slice.

Suppose there are six objects O_1, O_2, O_3, O_1', O_2', O_3' in a scene domain, B_0, B_1, B_2, B_3, B_4, B_5 are boundaries, as in Fig. 1, where in a binary scene representation is given. The boundary B_0 of the scene domain is considered to be an internal boundary and forms the root of the

containment tree. It contains immediately inside two other boundaries B_1 and B_2, both of which are external boundaries. This is indicated by two arcs connecting B_1 and B_2 to B_0. B_1 does not contain any boundaries and hence is a leaf node. B_2 contains B_3 and B_4 (both internal boundaries) and B_4 contains B_5 (external boundary). Note that the levels in the tree alternate between internal and external. A node corresponding to an external boundary together with its children constitutes an object. For example, B_2 with B_3 and B_4 constitutes O_2. Analogously, a node corresponding to an internal boundary together with its children constitutes a background component. The purpose of all this description is to point out that a boundary can be identified with a unique component, each external boundary with an object component and each internal boundary with a background component, following a bottom up order in the containment tree.

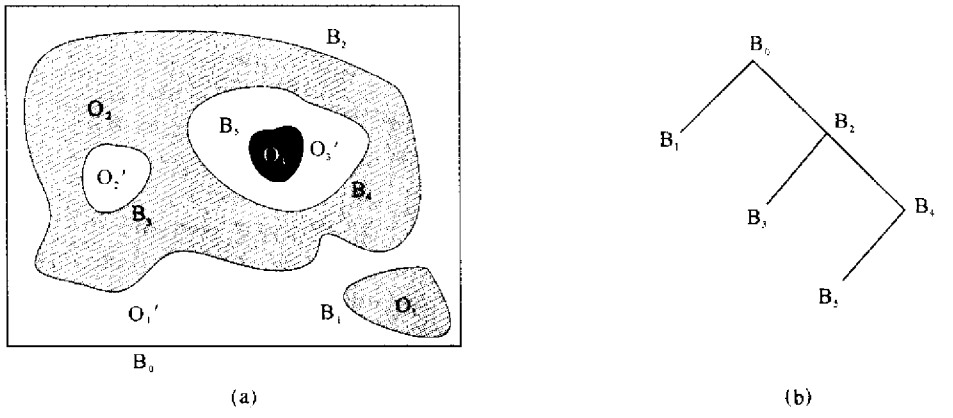

Fig.1 A binary scene representation of contour segmentation
(a) O_1, O_2, O_3 and its containment tree; (b) B_0, …, B_5 are the boundarites

We divide $C^{(m)}$ into contiguous slabs of constant object topology with the user's help, and within each slab, identify all boundaries of the 3D object O. A slab I of constant object topology (or slab for short) is a set of consecutive natural slices C_i and C_{i+1} in I if all of the following conditions are satisfied: (1) the number of 2D connected components of O is the same; (2) each 2D component of O in C_i is adjacent to exactly one 2D component of O in C_{i+1}. Our approach will be to take the user's help in identifying slabs in $C^{(m)}$ and, for each slab, in selecting a structure/co-structure for uniquely indicating a boundary to be created.

The complete segmentation process consists of five main steps: (1) slab definition; (2) structure/co-structure selection; (3) selection of orthogonal slices; (4) ordering of points; (5) boundary detection.

In Step (1), given a 3D scene $C^{(m)}$ and a 3D object of interest in $C^{(m)}$, the user initially recognizes all structures of the 3D object by observing all natural slices of $C^{(m)}$ in a montage display and then specifies the first natural slice of each slab. The slabs are contiguous so the first natural slice of a current slab determines the last natural slice of the previous slab determines the last natural slice of the previous slab, and $C^{(m)}$ is the last natural slice of the last slab by default.

In Step (2), to select a structure/co-structure S in a given slab I, the user chooses any natural slice of I and specifies a point c in the interior of S in this natural slice. The point c should be chosen such that the intersection between an axis Z_c, which is parallel to the z axis and which goes through c, and any other natural slice in I is a point in the interior of S.

EDGE DETECTOR

The edge detector is based on a one-dimensional second-order polynomial filter. As the shape modifications of anatomical structures Q_k ($k = 0$ for the section on which the seed contour B_0 is traced, $k \leqslant 1$ for the following sections)

are roughly smooth scale changes, the filters are applied to a set of limited-length scan lines on slice Q_k. The scan lines are derived from the H_{k-1} points $P_{h,k-1}$, $h = 1, H_{k-1}$ belonging to the contour B_{k-1}, extracted or traced on slice Q_{k-1}, as follows (see Fig.2):

(1) For each point $P_{h,k-1}$, calculate the unit vector orthogonal to B_{k-1} in the point considered.

(2) Along the direction of such a vector, resample the input image on a S_w-sample long scan line $L_{h,k-1}$ centered around $P_{h,k-1}$.

Using this strategy, the filter is applied only to a neighborhood of the contour extracted on the last-processed slice. This prevents detection of edges similar to the ones belonging to the anatomical structures to be segmented that are far from the region of interest and possibly belong to other structures.

The detector is based on a one-dimensional polynomial filter cascaded with a thresholding stage. The filter operates on N_p samples of a scan line that are located in the neighborhood of the sample x_i for which the filter output $o(x_i)$ is computed. It has the following structure:

$$o(x_i) = c_0 + \sum_{j=1}^{N_p} c'_j \cdot x_{i-d_j} + \sum_{l=1}^{N_p} \sum_{j=1}^{l} c''_{j,l} \cdot x_{i-d_j} \cdot x_{i-d_l} \quad (5)$$

where c_0, c'_j, $c''_{j,l}$ ($j, l = 1, \cdots, N_p, l \geqslant j$) are the filter coefficients and d_j ($j = 1, N_p$ are the offsets along the scan line, with respect to x_i ($i = 1, S_w$) of the N_p samples. The edge points extracted by the filter have the coordinates of those samples x_i for which the value of $o(x_i)$ is greater than a threshold T. The output of the detector $O(x_i)$ is therefore computed as follows:

$$O(x_i) = \begin{cases} 1 & \text{if } o(x_i) > T \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

In the tests performed, we used this edge-detection strategy. In an alternative implementation, thresholding can also be performed according to the following rule:

$$O(x_i) = \begin{cases} 1 & \text{if } (o(x_i) - Z) < 0 \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

which implements a threshold-crossing detector, with threshold Z.

Tracing contours manually is the most accurate method to produce segmentations of 3D structures. Contour-tracking procedures have been used with some success by constraining the segmentation process with some regularizing assumptions. This allows a contour, detected on one section, to be used to seed the segmentation of the same structure in a neighboring slice of the sequence. The seed contour and the reference contours are drawn (using special sectioning and visualization tools) on parallel sections, which can arbitrarily cut the 3D data set.

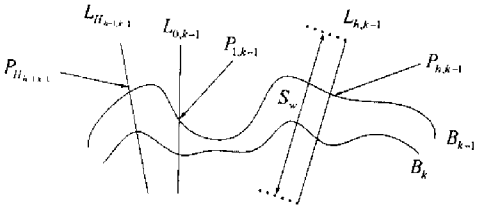

Fig.2 Definition of the scan lines for the contour-tracking procedure: for each point $P_{h,k-1}$ of the contour B_{k-1}, a scan line $L_{h,k-1}$ centered around $P_{h,k-1}$ is defined, with a length of S_w samples, along the direction of the unit vector orthogonal to B_{k-1} in $P_{h,k-1}$

If other "correct" contours, lying on sections which are not parallel to the references, are available, the contour points resulting from the intersection between such contours and the slices that still have to be segmented can be pre-set and used to further constrain the elastic contour model.

ALGORITHM EVALUATION

In assessing the goodness of a segmentation method, three factors (precision, accuracy and efficiency) need to be considered. (1) Precision refers to the repeatability of the method and can be measured by evaluating the variations in the result of segmentation because of subjective operator input. (2) Accuracy refers to the degree of agreement with truth. In practice, especially in medical applications, true segmentation is almost impossible to establish, as such various surrogates of truth are used. Usually, delineation by a knowledgeable operator is used as such alternative one. (3) Efficiency refers to the practical viability of the method (Table 1). This consists

of two parts, the computational time and the operator time. Computational time independent of operator involvement does not really matter so long as it is not prohibitive. Operator time required per study, or alternatively the effective number of slices that can be segmented per unit time, can be measured to express efficiency of the method.

Table 1 User's time (min) for both operators and methods for segmenting 3D model with different number of slices

Number of slices	35	42	46	52	58	61	67	73	81
The time of O_1 in 2D	6.3	6.9	7.1	7.5	8.3	8.5	8.9	9.1	10.2
The time of O_2 in 2D	7.1	7.3	7.9	8.2	8.6	9.3	9.9	10.9	11.3
The time of O_1 in 3D	3.3	3.6	3.8	3.8	3.9	3.7	3.8	3.4	3.6
The time of O_2 in 3D	3.5	3.6	4.0	3.7	3.8	3.6	3.5	3.9	41

EXPERIMENTAL RESULTS

We scanned and normalized the sequence of CT slices. For comparison, we extracted 2D features from the slice shown and clustered the feature image.

The features image extracted directly extracted from slice are more accurate than that extracted from 3D volume(Fig. 3). It is obvious that segmentation directly based on slice are better than segmentation contours, details, and continuity of issues based on 3D volume.

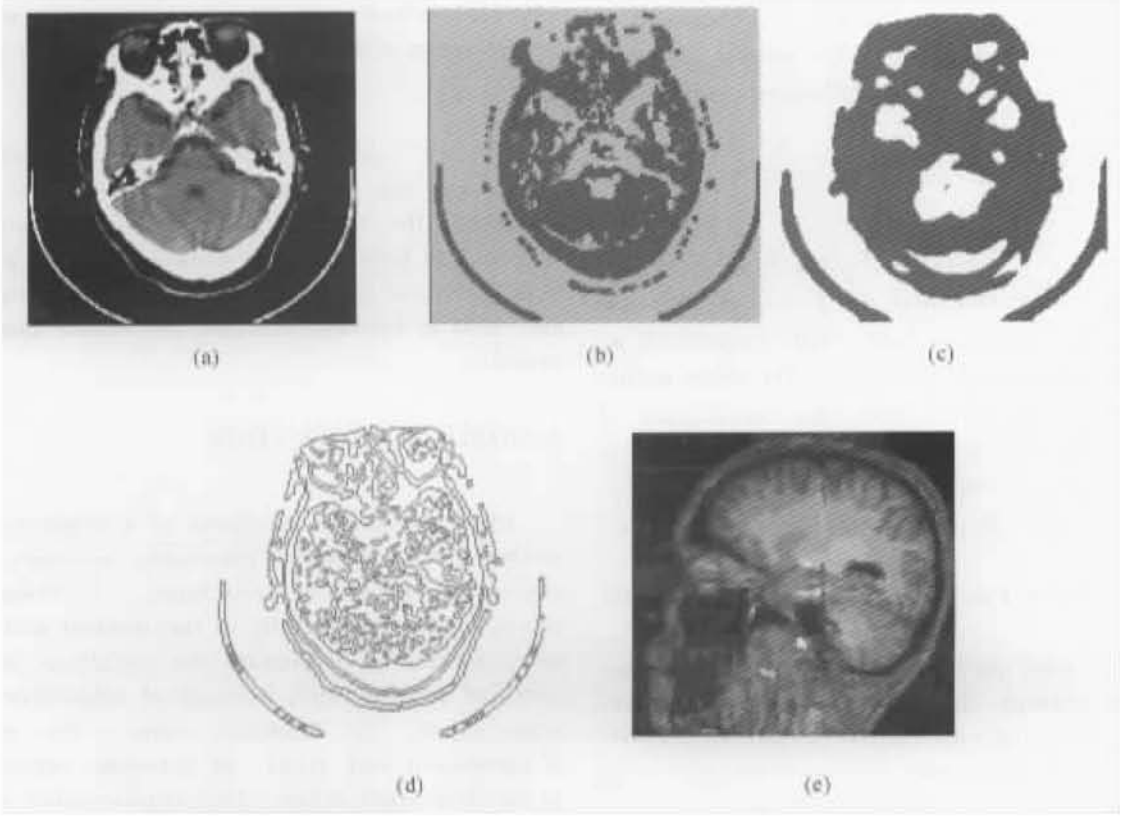

Fig.3 The experimental results of CT slices segmentation and brain volume segmentation.
(a) Original CT slice; (b) Segmented CT slice; (c) Opening and closing; (d) Edge detector; (e) Brain segmentation

The segmentation algorithm has been applied to 68 different 3D CT scans of 12 healthy subjects, 50 subjects with temporal epilepsy, and 6 subjects with extra-temporal epilepsy. Fig. 3

shows a typical result of the brain CT slices and brain volume segmentation algorithm. The quality of the segmentation depends on the level of noise and on the presence of artifacts in the original image. In fact, even with the preprocessing operation, some important artifacts such as bright dots, lines, or planes with spurious signal, may not be completely removed and may defeat the 3D region growing technique. The brain segmentation in which the segmentation results were not validated correspond to 3 CT images that present a plane with spurious signal or very bright dots. The connected 3D region obtained in these cases included a part of the skull around the artifact.

DISCUSSIONS

One possible initial disadvantage of the method is the somewhat unfamiliar object cross-section presented on arbitrarily oriented orthogonal slices. In applications involving the processing of a large number of similar data sets, some time spent initially in understanding this issue and in planning the selection of slabs and orthogonal slices, can significantly improve the efficiency of the method.

References

Bao, X.D. and Xiao, S.J., 1998. Three-dimensional segmentation of CT images using neural network. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, **20** (2).

Feng, T., 2002. An unified approach to missile guidancelaws: a 3D extension. *American Control Conference*, **2**: 1711 – 1716.

Flyun, P. J., 1990. CAD Based Computer Vision Modeling and Recognition Strategies. PhD thesis, Michigan State University.

Ge, Z.Y., Venkatesan, V. and Mitra, S., 2001. A statistical 3-D segmentation algorithm for classifying brain tissues in multiple sclerosis. *In*: Computer-Based Medical Systems. CBMS 2001. Proceedings of 14th IEEE Symposium, p. 455 – 460.

Luca Foresti, G. and Pieroni, G., 1998. Exploiting neural in range image understanding. *Pattern Recognition Letters*, **19** (2): 869 – 878.

Marcondes, R., Cesar, C. Jr. and Luciano da Fontoura, C., 1999. Computr-vision-based extraction of neural dendrograms. *Journal of Neurosience Methods*, **93** (5): 121 – 131.

Masato, Y. and Hasegawa, M., 1996. Extraction of Brain Tissues by Non-parametric Region Growing Method. *In*: MR CT Brain Image Analysis. 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Amsterdam.

Poli, R., Cagnoni, S. and Valli, G., 1992. Genetic and learning automata algorithms for adaptive digital filters. *Proc. ICASSP*-92, **4** (1): 41 – 44.

Shen, D., Herskovits, E. H. and Davatzikos, C., 2001. An adaptive-focus statistical shape model for segmentation and shape modeling of 3-D brain structures. *Medical Imaging, IEEE Transactions on*, **20** (4): 257 – 270.

Winterer, J. T., Schaefer, O. and Uhrmeister, P., 2002. Contrast enhanced MR angiography in the assessment of relevant stenoses in occlusive disease of the pekvic and lower limb arteries: diagnostic value of a two-step examination protocol in comparision. *European Journal of Radiology*, **41**(2): 153 – 160.

Wyatt, C. L., Ge, Y. and Vining, D. J., 2000. Automatic segmentation of the colon for virtual colonoscopy. *Computerized Medical Imaging and Graphics*, **24** (3): 1 – 9.