Prototype of rectal wall ultrasound image analysis system

Di Xiao[a], Wan Sing Ng[a], Udantha R. Abeyratne[b], Charles Bih-Shiou Tsang[c]
[a]School of Mechanical & Production Eng., [b]School of Electrical & Electronic Eng., Nanyang Technological Univ.; [c]Dept. of Colorectal Surgery, Singapore General Hospital, Singapore

ABSTRACT

This paper presents a software system prototype for rectal wall ultrasound image processing, image display and 3D reconstruction and visualization of the rectal wall structure, which is aimed to help surgeons cope with large quantities of rectal wall ultrasound images. On the core image processing algorithm part, a novel multigradient field active contour model proposed by authors is used to complete the multi-layer boundary detection of the rectal wall. A novel unifying active contour model, which combines region information, gradient information and contour's internal constraint, is developed for tumor boundary detection. The region statistical information is described accurately by Gaussian Mixture Model, whose parameter solution is computed by Expectation-Maximization algorithm. The whole system is set up on Java platform. Java JAI technology is used for 2D image display, Java3D technology is employed for 3D reconstruction and visualization. The system prototype is currently composed of three main modules: image processing, image display and 3D visualization.

Keywords: Active contour algorithm, rectal ultrasound image, 3D visualization, system software

1. INTRODUCTION

During the past decade, endoscopic ultrasonography (EUS) has been becoming one of the most common techniques for screening test and early staging of the colorectal tumors. One of the advantages of EUS is that it can get cross-sectional images of the rectum [1]. From these cross-sectional images, the anatomical structure of the organs and the situation of tumors are clearly displayed. But present commercial ultrasound machines just supply the 3D box display of the rectum wall images. A further image information acquisition is not supported from the image processing technology. Therefore, in order to build the structure information of the rectum and the position information of potential rectal tumor from a ultrasound sequence, surgeons need to view the ultrasound images slice by slice, and finally to form a full structure picture about that rectum in their mind. Thus, it is a time-consuming and tedious work for surgeons to perform object recognition and information extraction. To our knowledge, in image processing area, there are still no methods that have been described in the literature to help surgeons complete this task.

The goal of our project is to develop an automatic or semi-automatic analysis system to aid surgeons to cope with thees large quantities of rectal wall ultrasound images, involving rectal layer detection, rectal wall tumor detection, 3D reconstruction of rectal anatomic structure and corresponding quantitative analysis. This will be greatly helpful for reducing the surgeons' work intensity.

The muscular layer's boundary detection from a rectal wall ultrasound image is difficult because of the low spatial resolution of ultrasound images, the thin muscular layer structure and the absence of the layer segment penetrated by the tumor. Therefore, conventional approaches, such as low-level edge detection and edge linking methods etc., are hare to complete our task. Rectal tumor detection is another difficult problem, because of the non-homogeneous feature of the tumor region.

In the recent years, elastic deformable model (snakes) is one appreciated algorithm used by many researchers for boundary detection in image processing. The development of active contour models results from the works of Kass, Whitkin, and Terzopoulos (1987), and they offer a solution to a variety of tasks in image analysis and machine vision [2]. The active contour model is defined as an energy-minimizing spline, whose energy depends on its shape and location within the image. Local minimum of this energy corresponds to desired image properties such as boundary of object. A brief review about its development is given in the below.

Amini et al, (1988) pointed out some of the problems with this approach, such as instability and a tendency for points to bunch up a strong portion of an edge, and proposed an algorithm for the active contour model using dynamic programming [3]. After analyzing the snake method and dynamic programming, Willians gave a fast algorithm for

Medical Imaging 2002: Image Processing, Milan Sonka, J. Michael Fitzpatrick,
Editors, Proceedings of SPIE Vol. 4684 (2002) © 2002 SPIE · 1605-7422/02/$15.00

active contour, which was stable, flexible, allows hard constraints, and runs much faster than the dynamic programming method [4]. In Amit's work, an integrating gradient and region information are added into the deformable model to form a new framework [5]. Deformable template is a method which is less sensitive to the initial position. This kind of method usually sets up a set of template database, then uses template matching algorithm to get the optimal solution. Cootes and Taylor's research is a good example of this kind of method. The main partition of method emphasizes on building the statistical model of template. In the early stage of their work, an algorithm of set up statistical model such as electrical resistors and human hands is given [6]. At present, they have applied this algorithm on the building of appearance and recognition of human face. The "energy matching based on deformable templates" method given by Tawfik is also a combination of template method and deformable model algorithm [4].Gradient vector flow (GVF) is a novel method to solve the initial position problem presented by Dr. Xu recently[7]. A dual active contour model is proposed in [8] aimed to solve the initial sensitivity problem.

Rueckert [9] proposed an adaptive spline model, which describes the contour by a set of piece interpolating polynomial spline. Thus the object contour becomes continuous and smooth and the accuracy of model can increase by interpolating more control points. Menet also gave a B-snake model in [10][11], which is fitted to the discrete vertices using a least-square fitting technique. LeGoualher [12] gave a snake-spline model, originally developed in [13], whose internal energy is implicitly defined by the B-spline representation. Staib proposed a parametrically deformable model, which is based on the elliptic Fourier decomposition of the boundary [14]. The similar works are also done by Bonciu [15] and Szekely [16]. Pizer proposed a defomable shape loci [17], which is integrated the local shape properties from the medial primitives sought throughout the image space. Methods involving these continuous loci of the medial primitives are robust against the noise and the variation of the shape [18][19].

The first and primary uses of deformable models in medical image analysis was the application of deformable contour models to segment structures in 2D images [20][21][22].

In the following section of this paper, we will firstly introduce the algorithm we proposed for rectal wall muscular layer detection and tumor boundary detection. Then, a designed prototype of system software is given for implementing the image processing, boundary detection and 3D visualization of the rectal wall structure from the ultrasound image sequence.

2. METHODS

For rectal wall image processing task, we proposed a multi-gradient field active contour model for rectal wall muscular layer detection, and a unifying active contour model based on image's region and gradient information for tumor boundary detection. The details of above two models can refer to [23][24]. In the following subsection, we will give a brief introduction.

2.1 Multi-layer boundary detection of rectal wall by multigradient field active contour

The original idea aims at detecting the rectal wall layers one by one from inner to outer by an active contour. When one layer is detected, the algorithm should be able to drive the active contour expand again and search the next outer layer automatically. Traditional "snakes" algorithm cannot meet this requirement, because after finding a boundary, it has arrived at an energy minimum and can not deform again.

By analyzing the energy function and kinetic function of traditional snakes, we found that image energy is a gradient vector field after performing a gradient operation on the original image. The image force, which deforms the active contour, is a gradient vector field obtained from the gradient operation on the image energy component. It means that image force is a second-order derivative component derived from the original image.

In order to explain the multigradient field algorithm, two gradient vector fields are defined to clarify the purpose. (1). *First-order gradient vector field (FOGVF)* – Gradient vector field obtained from the gradient operation on original image. The corresponding image is called first-order gradient image (map). (2). *Second-order gradient vector field (SOGVF)* – Gradient vector field obtained from the gradient operation on the first-order gradient map. The corresponding image is called second-order gradient image (map).

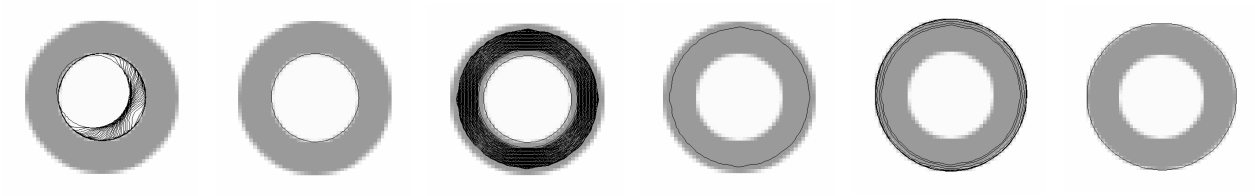
A novel active contour model incorporated with a multigradient vector field (multi-GVF) algorithm has been developed. This model, called multigradient field active contour model, can perform the task of multilayer boundary detection. The energy model of the contour is defined as:

$$E_{snake} = \int E_{\text{int}}(v(s)) + w_{image}E_{image}(v(s)) + w_{adpt}E_{\exp}(v(s))ds \tag{1}$$

where, w_{image} and w_{adpt} are weighting values. $v(s) = [x(s), y(s)]$ represents the deformable model and s is proportional to the arc length. E_{int} is internal energy of the active contour. E_{image} is image energy. Adaptive expanding energy $E_{\exp}$ is an external energy. The numerical solution of this energy function is as follows,

$$\begin{aligned} \mathrm{x}_t &= (\mathrm{C} + l\mathrm{I})^{-1}(l\mathrm{x}_{t-1} - w_{image}\mathrm{f}_{\text{image\_x}}(\mathrm{x}_{t-1}, \mathrm{y}_{t-1}) - w_{adpt}\mathrm{f}_{\text{adpt\_x}}(\mathrm{x}_{t-1}, \mathrm{y}_{t-1})) \\ \mathrm{y}_t &= (\mathrm{C} + l\mathrm{I})^{-1}(l\mathrm{y}_{t-1} - w_{image}\mathrm{f}_{\text{image\_y}}(\mathrm{x}_{t-1}, \mathrm{y}_{t-1}) - w_{adpt}\mathrm{f}_{\text{adpt\_y}}(\mathrm{x}_{t-1}, \mathrm{y}_{t-1})) \end{aligned} \tag{2}$$

here, $\mathrm{f}_{\text{image\_x}}(\mathrm{x}_{t-1}, \mathrm{y}_{t-1})$ and $\mathrm{f}_{\text{image\_y}}(\mathrm{x}_{t-1}, \mathrm{y}_{t-1})$ are image force derived from image energy, $\mathrm{f}_{adpt\_x}(i) = \partial E_{adpt}(x, y)/\partial x_i$ and $\mathrm{f}_{adpt\_y}(i) = \partial E_{adpt}(x, y)/\partial y_i$ are adaptive external forces. l is step size and I is unit matrix. x_t and y_t are x and y coordinates of the contour node at current time t. C is a pentadiagonal matrix that includes the contour's shape weighting coefficients.



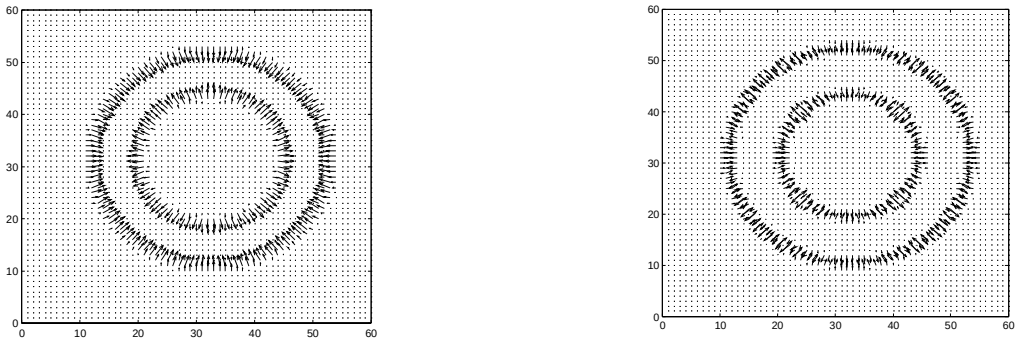
(a) Extract inner boundary under SOGVF

(b) Escape from the inner border

(c) The intermediate result

(d) Extract the outer boundary under SOGVF again.

Fig. 1: Demonstration of multigradient active contour algorithm



(a) Gradient vector field derived from the inverse map of the original image – first-order gradient vector field

(b) Gradient vector field derived from first-order gradient map–second-order gradient vector field

Fig. 2: FOGVF and SOGVF of image

Fig. 1 illustrates how to detect the inner and outer boundary of a ring. Here, we adopt an image, which is a black ring on a white background, as our sample. A detailed procedure of the algorithm's implementation is described as follows:

1. Compute the FOGVF and SOGVF of the original image, and the FOGVF of its inverse image.

2. Set up the initial active contour (a circle is adopted here) in the central region of the ring. The initial contour should be fully closed by the ring.
3. Deform the active contour under the constraints of the internal forces, image forces and expanding forces. Here, the image forces in Eq. (2) are the second-order gradient field, $\partial \mathrm{E}_{\mathrm{image}}(\mathrm{x}_{t-1}, \mathrm{y}_{t-1})/\partial \mathrm{x}_{t-1}$ and $\partial \mathrm{E}_{\mathrm{image}}(\mathrm{x}_{t-1}, \mathrm{y}_{t-1})/\partial \mathrm{y}_{t-1}$. In the homogenous region, the adaptive expending forces play an important role to deform the active contour. In the capture range of the boundary, the adaptive expanding forces are decreasing, and the image forces (SOGVF) act as the strong constraints on the deformation of the contour. When this active contour reach the equilibrium of its energy, the inner boundary of the ring is detected.
4. Use above result as a new initial contour, deform the active contour model again under the constraint of the internal forces, image forces and expanding forces. The image forces in Eq. (2) change to the first-order gradient field, $\mathrm{E}_{\mathrm{image}}(\mathrm{x}_{t-1}, \mathrm{y}_{t-1})$, which is derived from the inverse map of the original image. The expanding forces are chosen as a constant rather than an adaptive changeable one. Because the image is a black ring on a white background in this example, the orientation of FOGVF from inverse image is centrifugal on the inner boundary as what we wish. These two centrifugal forces (image force and expanding force) can help active contour escape from the current location as illustrated in Fig. 1 (b). The active contour passes through the homogeneous area of the ring and finally reach the equilibrium of its energy when it near to the outer boundary of the ring. Because the equilibrium location is not the minimum of the second-order image force, the current location is not really a place of the outer boundary, just near to it, as shown in Fig. 1 (c).
5. This step is a fine-tuning or refinement of the final border position. The SOGVF is used again like step 3, and the contour model continues to deform under the second-order gradient vectors, internal forces and adaptive expanding forces. Finally, when energy minimum is reached, the accurate location of the outer border is obtained (Fig. 1 (d)).

If there are more concentric layers, the algorithm can repeat steps 3 and 4 to implement the detection task, but the image force should be treated carefully. For example, if there is another layer outside, we need a centrifugal image force again as image force. In order to meet this requirement, at this time, FOGVF should be calculated from the original image. So there is a simple regulation for this kind of concentric layer structure's detection. If the detecting procedure is from inner to outer and the gray values of all layers change according to the form of "white' and "black" alternately from inner to outer, we should arrange the boundary detection as follows. Firstly we should adjust the FOGVF's orientation of the innermost boundary, then choose the FOGVF for different boundary alternately from the original image or its inverse map.

2.2 Tumor boundary detection by unifying region- and boundary-based active contour

2.2.1 A unifying active contour

The tumor region and its boundary detection are what the surgeons more care about. The boundary detection of the tumor by traditional active contour algorithm encounters the problem of non-homogeneous feature of the tumor region. Therefore, the deformation of the active contour guided only by image gradient feature becomes unstable. To solve this problem, we proposed a novel unifying active contour model, which combine the region feature and gradient feature to guide the deformation of the contour. Its energy function is as follows

$$E(\partial C) = w_{regn}\left[-\iint_{(x,y)\in R(\mathbf{I}_O)} \log\big(p_O(I(x,y))\big)dxdy - \iint_{(x,y)\in R(\mathbf{I}_B)} \log\big(p_B(I(x,y))\big)dxdy \right] + \tag{3}$$

$$w_{grad}\left[\iint |\nabla I(x,y)|dxdy \right] + \iint ((\partial(s)|v_s(s)|^2 + \beta(s)|v_{ss}(s)|^2)/2)ds$$

where, w_{grad} and w_{reg} are weighting parameters. $\mathbf{I}_O$ and $\mathbf{I}_B$ represent the image intensity on object region and background region of a image, and $p_O(I(x,y))$ and $p_O(I(x,y))$ are the probability distribution functions of pixel gray

values in these two regions, respectively. The whole energy function consists of three components: image region energy, image boundary (gradient) energy, and internal energy.

The region-based energy is defined to maximize the *a posterior* probability of the region information and perform the segmentation of the image. For the special case such as boundary finding of a single object, one initial contour seed with the corresponding object's and background's intensity probability distribution is needed. From above formulation, we have known that when a pixel is classified correctly, such as belonging to the object area, the probability of $p_O(I(x,y))$ will be greater than that of $p_B(I(x,y))$, otherwise, the probability of $p_B(I(x,y))$ greater than that of $p_O(I(x,y))$.

This module that decides the boundary finding of the object is based on the boundary information. Here we choose the gradient magnitude from the image as the boundary information of the object like traditional approach. The segmentation of image or object detection is to find the boundary that has the maximum energy on the gradient magnitude field of a image.

In order to keep the shape feature itself, contour is also constrained by internal energy, which is borrowed from the traditional snakes that the tension and rigidity of the contour is constrained by its first- and second- derivatives.

In order to deform the active contour, we should derive its the numerical solution. Above energy function can be minimized using a gradient descent method. Through Euler-Lagrange algorithm, we can obtain the kinetic function of the active contour as

$$\frac{\partial}{\partial t}\partial(C) = w_{grad}\nabla\left|\nabla I(x,y)\bullet\nabla I(x,y)\right| - w_{regn}\log\left(\frac{p_O(I(x,y)}{p_B(I(x,y)}\right)\vec{\mathrm{n}}_O(\partial C) + \alpha(s)v_{ss}(s) + \beta(s)v_{ssss}(s) \tag{4}$$

Where, $\vec{\mathrm{n}}_O(\partial C)$ is the normal direction of the contour nodes. The motion of a contour is guided by three forces: internal force, region force, and boundary force (or gradient force).

2.2.2 Statistical model of the region feature

In this subsection, we will discuss how to set up an accurate statistical model for a specific image region by GMM algorithm and how to get the optimal parameters solution for this GMM.

For the purpose of our project, we set up a GMM comprising of three weighted Gaussian components to describe the statistical feature of the image intensity, expressed by

$$p(\mathbf{I}\mid\Theta) = \alpha_1 p_1(\mathbf{I}\mid\mu_1,\Sigma_1) + \alpha_2 p_2(\mathbf{I}\mid\mu_2,\Sigma_2) + \alpha_3 p_3(\mathbf{I}\mid\mu_3,\Sigma_3) \tag{5}$$

$$p_i(\mathbf{I}\mid\Theta_i) = p(\mathbf{I}\mid\mu_i,\Sigma_i) = (2\pi)^{-d/2}\left|\Sigma_i\right|^{-1/2}\exp\left[-\frac{(\mathbf{I}-\mu_i)\Sigma_i^{-1}(\mathbf{I}-\mu_i)}{2}\right] \tag{6}$$

The parameter solution of this Gaussian model is the estimation of the following nine parameters $\Theta = \{\alpha_1, \alpha_2, \alpha_3, \mu_1, \mu_2, \mu_3, \sigma_1, \sigma_2, \sigma_3,\}$. A common method to estimate these parameters is by maximizing the likelihood function of the above,

$$l(\Theta\mid\mathbf{I}) = \sum_{j=1}^{N}\ln\left(\alpha_1 p_1(\mathrm{I_j}\mid\mu_1,\Sigma_1) + \alpha_2 p_2(\mathrm{I_j}\mid\mu_2,\Sigma_2) + \alpha_3 p_3(\mathrm{I_j}\mid\mu_3,\Sigma_3)\right) \tag{7}$$

The EM algorithm is an iterative procedure to find the maximum-likelihood estimate of the parameters from an observed data set when these data are considered as incomplete data. In general, the maximization of the likelihood function $l(\Theta\mid\mathbf{I})$ (where, I is observed data, and Θ is the parameter set) is very complicated and may have no analytic solution. But by introduce the data set some intermediate variables, also called "latent data", which are never observable, it has shown that the estimation problem of maximum-likelihood becomes easily analytic solvable.

EM algorithm can be described by following two steps. The first step is E (Expectation) step and the second step is M (Maximization) Step. E step is to find the expected value of the complete-date likelihood function. The second step is aimed to maximize above expectation by iterative procedure and get the solution of the parameters Θ.

In the following, we give the steps that use EM algorithm to get the optimal parameter solution of a GMM,

1. Set the initial values for the parameter $\Theta=[\alpha_i, \mu_i, \sigma_i]$ by K-means algorithm, where i =1, 2, 3.
2. Calculate the *a posteriori* probability distributions $p(i \mid \mathrm{I_j})$ by using obtained Θ, where i =1, 2, 3, and j =1, …, N.
3. Now, the a posteriori probability distributions have known, the parameter $\Theta=[\alpha_i, \mu_i, \sigma_i]$ can be calculated directly as follows,

   3a. Calculate the mean of the model,

$$\hat{\mu}_i = \sum_{j=1}^{N} p(i \mid \mathrm{I_j})\mathrm{I_j} \Big/ \sum_{j=1}^{N} p(i \mid \mathrm{I_j}) \tag{8}$$

   3b. Calculate the variance of the model,

$$\hat{\sigma}_i = \sum_{j=1}^{N} p(i \mid \mathrm{I_j})(\mathrm{I_j} - \mu_i)^T(\mathrm{I_j} - \mu_i) \Big/ d\sum_{j=1}^{N} p(i \mid \mathrm{I_j}) \tag{9}$$

   3c. Calculate the weighted value,

$$\hat{\alpha}_i = \frac{1}{N}\sum_{j=1}^{N} p(i \mid \mathrm{I_j}), \quad i = 1,2,3 \tag{10}$$

4. A new set of the estimate parameter $\hat{\Theta}=[\hat{\alpha}_i, \hat{\mu}_i, \hat{\sigma}_i]$ is obtained. If the sequence of Θ becomes steady, such as $\left\|\Theta - \hat{\Theta}\right\|$ is less than a threshold, the EM algorithm is ended, otherwise, we return to the step 2 for further calculation.

3. PROTOTYPE OF APPLICATION SOFTWARE

3.1 system overview

The whole software system is developed on the Java platform for its implantable feature in the different operation systems. The more important point is the Java's beautiful look and feel style and strong support to 2D display and 3D visualization. The abundant classes for image processing, 2D image display and easier 3D scenario programming is one of the important reasons we adopted this development environment. In the following, we will introduce this rectal wall image analysis system image processing point of view to introduce each functional module.

3.2 Software functional modules

As illustrated in Fig. 3, at present, the whole software system consists of three main functional parts: image processing, 2D image display and 3D visualization. Image processing functional part provides various functional modules to perform the tasks such as rectal wall image processing and boundary detection of the muscular layer. The design of the 2D image display functional part is aimed to help users to browse the image slices from x, y and z three directions and accurately know the position information of the current slice in the image sequence. The third functional part is what the users mostly hope. This part provides the function of visualization of a 3D rectal wall muscular layer structure. The details of each functional part are explained in the following parts.

3.2.1 Image processing functional part

The image processing part consists of three main functional modules presently. They are image I/O module, image preprocessing module and rectal muscular boundary detection module, and boundary detection module of ROI (region

of interest) as tumor. Each module is further composed of several sub-functional modules implemented by Java classes, which encapsulate the necessary variables and functions to implement the special functions.

Image I/O module is used to manage the loading and saving of the image slices. Image loading is performed by a *FileSchooser* class to import the image sequence. Multiple images can be selected in and imported simultaneously. Image export class is used to save the possible ROI (Region of Interest) images to the subdirectory of the current image sequence directory. Because the sampled image from the ultrasound machine is very large and the interesting rectal wall region is much smaller, we design a ROI function that lets the user be able to choose his ROI (512×512 or 256×256 size) from one slice and the system will cut this ROI from each slice of the sequence and save them.

Image preprocessing module is aimed to perform some necessary image processing tasks before the procedure of boundary detection. Image preprocessing functions are encapsulated in an image preprocessing class. The provided functions include computing the first-order gradient field, second-order gradient field, inverse map of the original image, as well as Gaussian blur and smooth blur filtering.

Boundary detection module is one of the kernel parts in the software including several important classes. *VolumeMatrix* is a class that is used to store the image sequence data in a 3D matrix using 8 bits per voxel, which is convenient to create the needed image from the different sectional direction such as x, y, or z. Some necessary functions are also encapsulated in the class, such as parameters setting and getting for the dimension of the matrix. *Snakenode* class is used to define the structure of the node on the active contour for convenience of node's operation like insertion, deletion and motion. *Snake* class is a very large class with very strong function that provides a series of functions for performing the deformation of active contour and the boundary detection. The global searching algorithm, multigradent field active contour algorithm, and unifying active contour algorithm for boundary finding and tracking are all encapsulated in this class.

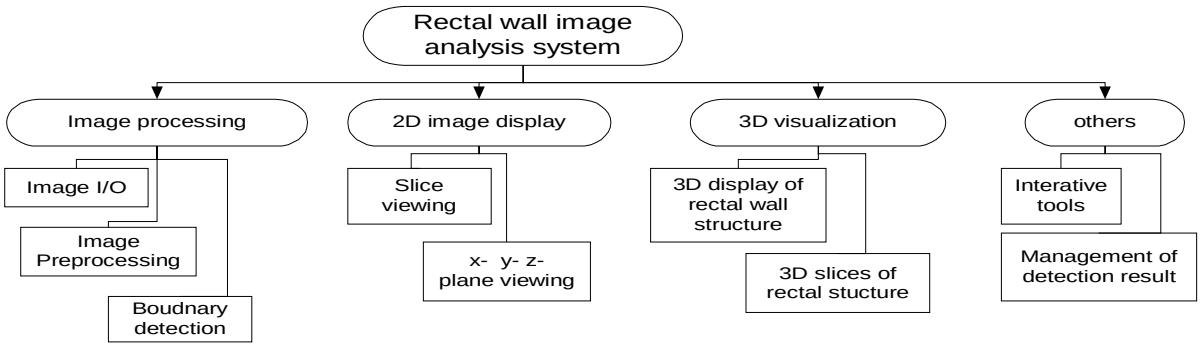


Fig. 3: Diagram of the functional modules of the software

3.2.2 2D image display functional part

2D image display functional part consists of two functional modules, involving slice view, x-, y- and z-directional slice view. The former module is similar to the latter, but provides more abundant drawing functions to support the image display, contour drawing and manual drawing by users interactively. This functional module is also designed to combine with a slider component to control the slice view of image sequence. The latter module is designed for display x- y- and z-directional slices on one panel for convenient image view. On each slice, there are two perpendicular lines to represent the position of the other two slices, which help users understand the accurate position of the slice he is viewing.

3.2.3 3D visualization functional part

3D visualization functional part is aimed to help users to view the anatomical structure of rectal wall from 3D space. Two important functional modules are arranged in this part. They are 3D object display module and 3D slice views module. 3D object display module is support by several Java classes, such as *RectalSurface*, *Panel3dWorld*, *Rectallayer3D* etc., developed by us. The function of this module is to perform the generation of 3D locale, generation of 3D rectal muscular layer surface from slices' contour, and surface rendering. 3D object as surface of rectal muscular layer can be shown in different ways: surface, voxels, and wire frame, for convenient visualization. Triangular patch

algorithm is used here for creation of the rendering surface of the object. Displayed objects can be rotated and viewed at any angles by mouse's operation. Java 3D technology provides a lot of functional classes for programming of 3D scenoria design, 3D lighting, color setting, interactive operation etc.. 3D slice views module provides the function of x- y- z- directional slice display at 3D scene. Three slices at x- y- z-direction can be inserted in the displayed 3D rectal layer surface for convenience of the comparable view between created rendering surface with its corresponding original images. This function benefits from the Java 3D texture rendering technology.

3.2.4 Others

In this part, some trivial functional modules such as toolbar module for interactive operation, file management and storage of the detection results, ROI measurement etc..

4. SOFTWARE SURFACE DESIGN AND EXPERIMENTAL RESULTS

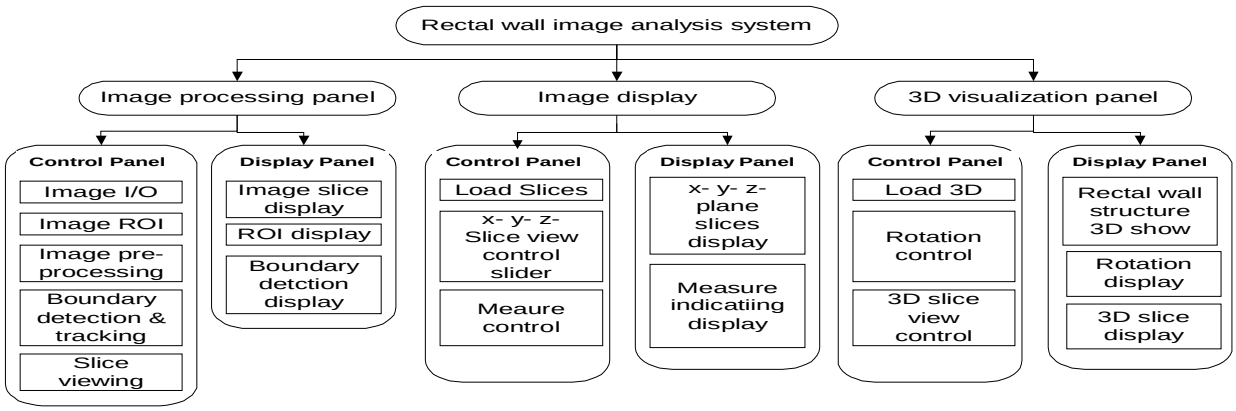


Fig. 4: Diagram of software surface

System software surface consists of three main panels according to their different functions as shown in Fig. 4. They are image processing panel, image display panel and 3D visualization panel. The entire technological process for rectal wall image processing can be described simply as load image sequence, automatic or semi-automatic rectal wall structure extraction, slice display and manual detection of interesting region, 3D visualization display of rectal wall and 3D slice display. According above idea, we designed three display panels to show the current concerned object and technological process. Three tabbed panels (control panels) are designed to control the switch among these display panels.

4.1 System surface design

4.1.1 Image processing panel

Image processing panel is aimed to load the rectal wall image sequence and guide users to complete the muscular boundary detection from the image sequence. The entire procedure of image processing is divided into three functional steps, including image loading, rectal wall muscular boundary detection and boundary detection of ROI (region of interest) like tumor or ulcer tunnel, as shown is Fig. 5 (a).

Image sequence loading is performed by a *File Chooser* dialog. Usually the image size obtained from ultrasound is bigger than that of the display panel. Therefore, a scroll panel is arranged for image display. An image view slider is put below the image display panel to help view the image sequence slice by slice. A ROI cropping function is arranged to let user choose a rectangular region for image processing. The system will automatically crop the rectangular regions from each slice and save them as independent files to a new subdirectory. The image displayed in the panel will change to the ROI image.

Rectal muscular boundary detection is arranged on the second step of this control panel. Users can choose to perform rectal wall muscular boundary detection automatically or semi-automatically on one slice. After users confirm the results on the current slice, the system will perform the boundary detection through out the whole sequence. A series of

buttons on this control area can control the muscular layer boundary detection on one slice step by step. Three tool buttons on the tool bar can help the modification of the detected boundary (contour) interactively. You can insert, remove or move the node that you have chosen on the contour.

Boundary detection of ROI step is to let the user choose the region of his interest and detect the boundary of region automatically.

4.1.2 Image display panel

Image display panel is aimed to display the x- y- and z-directional slices simultaneously on one panel, which is helpful for image slice view. Manual measure function can let the user use rulers to measure the region he is interested in. The functions include image slice loading, slice view and manual measurement, as shown in Fig. 5 (b).

The raw data for creating x- y- and z-directional slices are extracted from the *VolumeMatrix* class, which save each pixel of the image in byte format. Java 2D technology makes it easier to convert these raw data into an image format. Using the x- y- and z-directional control slider, users can control each slice display on three directions. On each slice, there are two lines to indicate the positions of other two slices, which give accurate position information to users. Manual measure function can let users to measure the distance and area of one region. Three tool buttons are design to implement these functions. Rectangle ruler button is used to create a rectangular region to supply the height and width information. Straightline ruler button help to measure the distance of any two points in the slice. Polygon ruler button help to measure the circumference and area of the selected polygon region. The resulting information is displayed on the status bar.

4.1.3 3D visualization panel

At present, 3D visualization panel just provides the function of 3D display, as show in Fig. 5 (c). The boundary information detected in the image processing module will be organized here to form a 3D object through Java 3D technology.

Each muscular layer's surface of the rectal wall can be displayed in a 3D box. By selecting the corresponding radio buttons, we can display the rectal wall layers by point, wire frame or surface style in a 3D scene.3D object can be rotated by mouse control. Hence, users can view the object from any angle. X – y- and z-directional slices also can be displayed in the 3D box with the rectal surfaces simultaneously. 3D slices display aims at providing more strong function for help users understand the position and structure information of the reconstructed 3D surface through comparing with the 3D slices. Each slice can move along its moving axis, controlled by its corresponding slider.

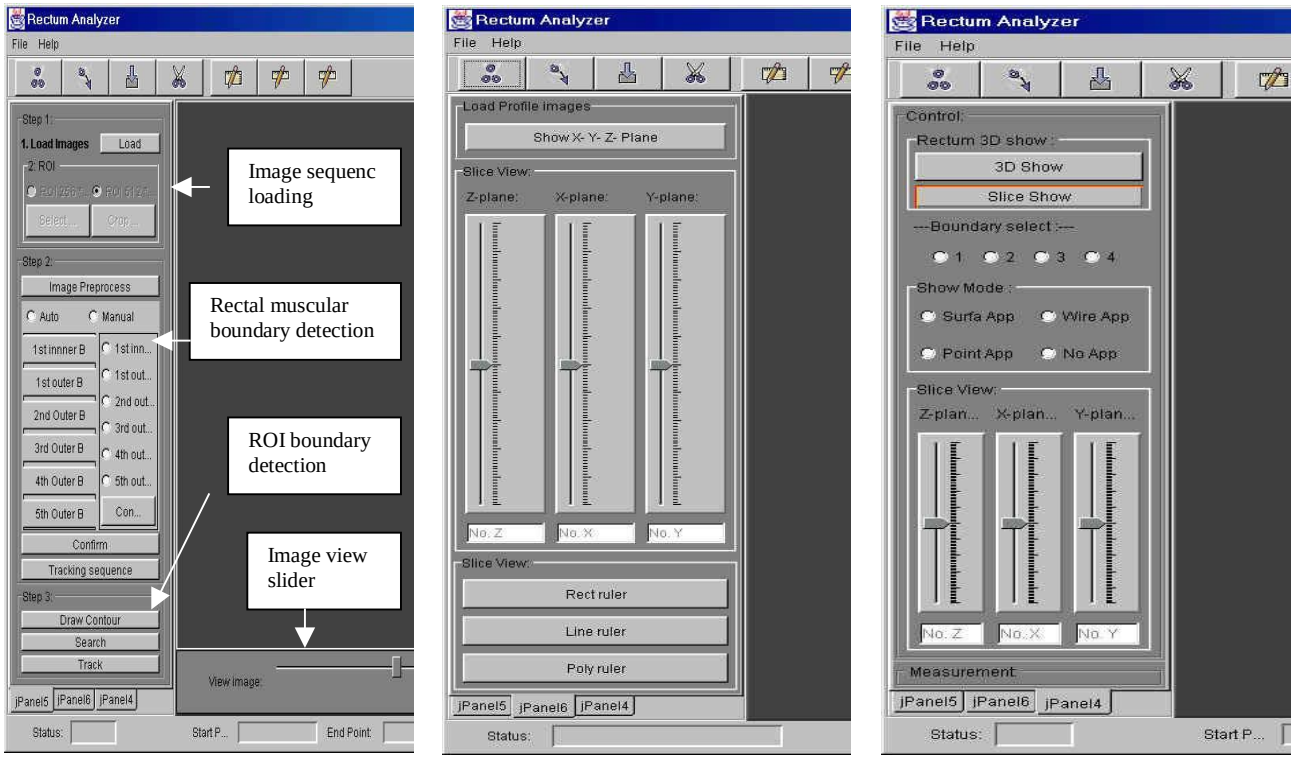
4.2 Experimental results

As we have mentioned above, users can be guided through out the whole procedure of the rectal wall layer detection and 3D reconstruction by our designed three functional panels. At present, we have complete the surface designed and rectal muscular boundary detection and 3D reconstruction and visualization. The tumor boundary detection algorithm has finished testing by Matlab, the java programming task is working in progress.

Fig. 6 (a) and (b) illustrate the detected innermost boundary and outer boundary of first inner muscular layer by multigradient field active contour algorithm. Fig. 7 (a) is a demonstration of x- y- and z-directional slice show. You can find the horizontal line on the z slice indicates the position of the y-directional slice below it, and the vertical line on z slice give the position information of the x-directional slice beside it. The rectangular frame on the z slice is a ruler that can measure the width and height of a rectangular region.

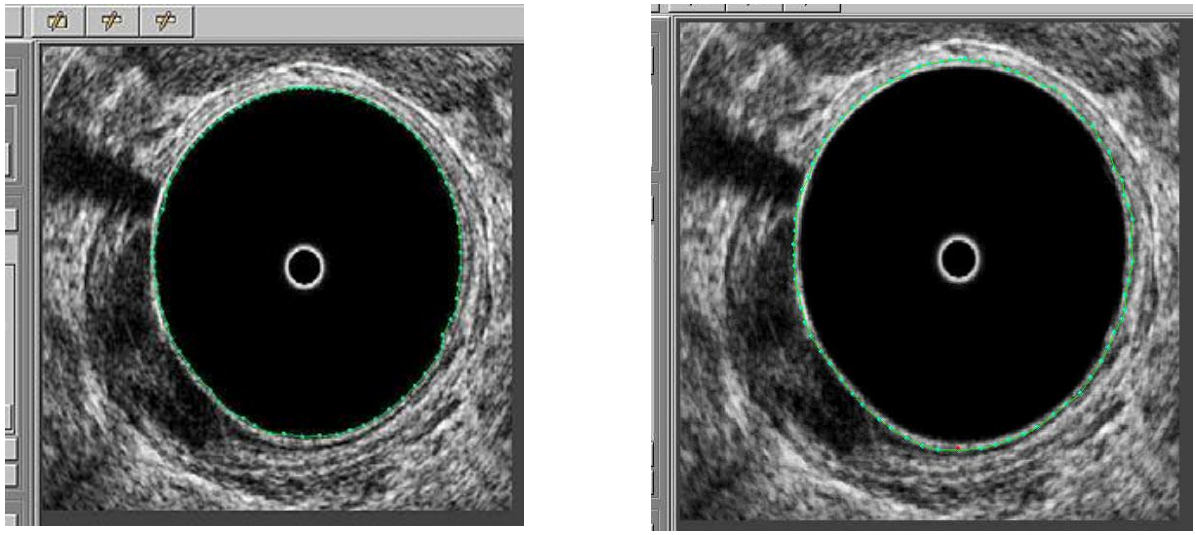
Fig. 7 (b) is a demonstration of 3D structure of two rectal muscular layer surface. In order to help view clearly, the inner layer is displayed by surface rendering mode, and the outer one the wire frame rendering mode. A z-directional 3D slice is also displayed with the 3D rectal surfaces at the same time. By mouse control, we can rotate the whole 3D box and view it at any angle.

Fig. 8 shows some tumor boundary detection results. These experiments just were completed on Matlab development environment. The work of transferring these codes to Java language is going in progress.



(a) Image processing panel (b) Image display panel (c) 3D visualization panel

Fig. 5: Three main panels of the software surface



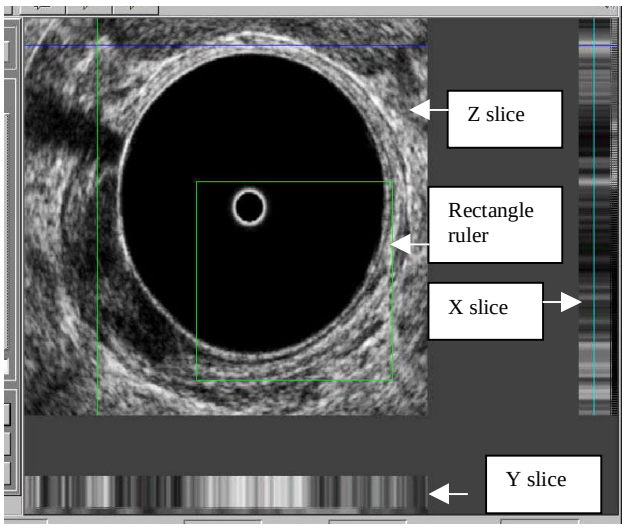
(a) Innermost boundary (b) Outer boundary of the inner first layer

Fig. 6: Boundary detection of the muscular layer

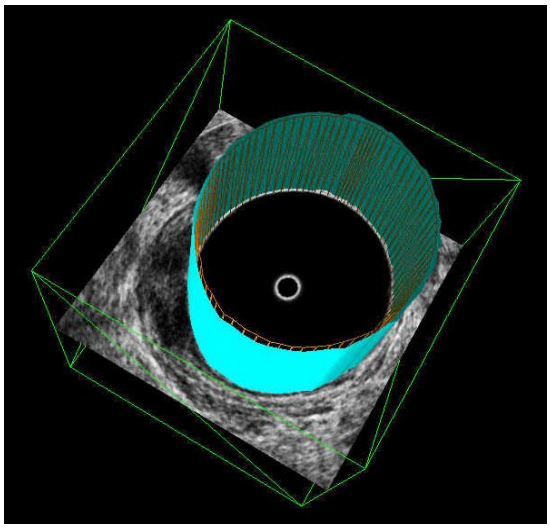
5. CONCLUSIONS

For rectal wall ultrasound image processing and structure detection of its muscular layer, we have developed two novel active contour algorithms for performing the tasks. Experimental testing of these two algorithms has been completed on

Matlab. A software prototype based on Java platform has been developed. Some basic functional modules such as user surface, interactive function, boundary detection algorithm for muscular layesr, and 3D visualization have been completed. More functional modules, like document management, tumor boundary detection, and more measurement functions are in programming.



(a) x- y- and z-directional slice show



(b) 3D visualization

Fig. 7: Slice show panel and 3D visualization panel

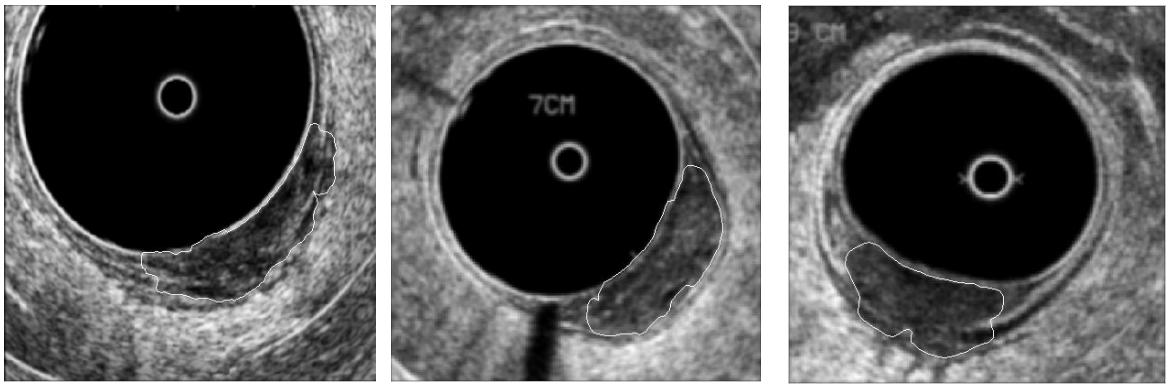


Fig. 8: Some experimental results of tumor boundary detection

REFERENCES

1. R. H. Hawes, "New staging techniques: Endoscopic ultrasound, Cancer," [Online Abstract]. Available: http://endosonography.com/program/eus-def.html, [1999 July 9].
2. M. Kass, & A. Witkin, & D. Terzopoulos, "Snake Active Contour Model," *Int. J. Comput. Vision*, **1**, pp.321-331, 1987.
3. A. A . Amini, & S. Tehran, & T. E. Weymouth, "Using dynamic programming for minimising the energy of active contours in the presence of hard constraints," *Proceedings of the Second International Conference on Computer Vision*, pp. 95-99, 1988.
4. D. J. Williams, & Mubarak Shah, "A Fast Algorithm for Active Contours," *Third International Conference on Computer Vision*, pp. 592 –595, 1990.

5. Y. S. Tawfik, & A. M. Darwish, & S. I. Shaheen, "Energy matching based on deformable templates," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, **6**, pp. 3481-3484, 1999.
6. T. F. Cootes, & C. J. Taylor, "Active shape Models-Their Training and Application," *Computer Vision and Image Understanding*, **61**, pp. 38-59, 1995.
7. C. Xu, & J. L. Prince, "Snakes, Shapes, and Gradient Vector Flow," *IEEE Trans. on Image Processing*, **7**, pp. 359-369, 1998.
8. S. R. Gunn, & M. S. Nixon, "A robust snake implementation; a dual active contour," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **19**, pp. 63-68, 1997.
9. D. Rueckert, & P. Burger, "Contour fitting using an adaptive spline model," *British Machine Vision Conference*, pp. 207-216, 1995.
10. S. Menet, & P. Saint-Marc, & G. Medioni, "Active contour mdoels: overview, implementation and application," *IEEE International Conference on Systems, Man and Cybernetics*, pp. 194-199, 1990.
11. S. Menet, & P. Saint-Marc, & G. Medioni, "B-snakes: Implementation and application to stereo," *DARPA Image Understanding Workshop*, pp. 720-726, 1990.
12. G. LeGoualher, & C. Barillot, & Y. J. Bizais, & J. M. Scarabin, "Three-dimensional segmentation of cortical sulci using active models," *SPIE Medical Imaging*, pp. 254-263, 1996.
13. F. Leitner, & I. Marque, & S. Lavall'ee, & P. Cinquin, "Dynamic segmentation: finding the edge with snake-splines," *Proceedings of International Conference on Curves and Surfaces*, pp. 1-4, 1990.
14. L. H. Staib, & J. S. Duncan, "Boundary Finding with Parametrically Deformable Models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, **14**, pp. 1061-1075, 1992.
15. C. Bonciu, & C. Lger, & J. Thiel, "A Fourier-Shannon approach to closed contour modeling," *Bioimaging*, **6**, pp. 111-125, 1998.
16. G. Szekely, & A. Kelemen, & C. Brechbuhler, & G. Gerig, "Segmentation of 2-D and 3-D objects from MRI volume data using constrained elastic deformations of flexible Fourier contour and surface models," *Medical Image Analysis*, **1**, pp. 29-34, 1996.
17. G. D. Stetten, & S. M. Pizer, "Automated Identification and Measurement of Objects via Populations of Medial Primitives, with Application to Real Time 3D Echocardiography," *Information Processing in Medical Imaging*, pp. 84-97, 1999.
18. G. D. Stetten, & S. M. Pizer, "Medial-Node Models to Identify and Measure Objects in Real-Time 3-D Echocardiography," *IEEE Trans. on Medical Imaging*, **18**, pp. 1025-1035, 1999.
19. S. M. Pizer, & D. S. Fritsch, P. A. Yushkevich, "Segmentation, registration, and measurement of shape variation via image object shape," *IEEE Transactions on Medical Imaging*, **18.** 851-865, 1999.
20. D. T. Morris, & C. Donnison, "Identifying the neuroretinal rim boundary using dynamic contours," *Image and Vision Computing*, **17**, pp. 169-174, 1999.
21. R. N. Czerwinki, & D. L. Jones, "Detection of Lines and Boundaries in Speckle Images-Application to Medical Ultrasound," *IEEE Trans. on Medical Imaging*, **18**, pp. 126-136, 1995.
22. S. Lobregt, & M. A. Viergever, "A Discrete Dynamic Contour Model," *IEEE Trans. on Medical Imaging*, **14**, pp. 12-23, 1995.
23. D. Xiao, W. S. Ng, U. R. Abeyratne, C. K. Kwoh, & C. B. Tsang, "Rectal wall structure delineation and broken layer recognition by multigradient filed active contour," The 7th Australian and New Zealand Intelligent Information Systems Conference, pp.123-127, 2001.
24. D. Xiao, W. S. Ng, U. R. Abeyratne, & C. B. Tsang, "Rectal tumor boundary detection by unifying active contour model," SPIE Medical Imaging 2002, 2002 (Accepted).