Automated Lung Nodule Segmentation Using Dynamic Programming and EM Based Classification

Ning Xu[a], Narendra Ahuja[a] and Ravi Bansal[b]

[a]ECE Department and Beckman Institute, University of Illinois at Urbana-Champaign, IL 61801
[b]Siemens Corporate Research, Inc. Princeton, NJ 08540

ABSTRACT

In this paper we present a robust and automated algorithm to segment lung nodules in three dimensional (3D) Computed Tomography (CT) volume dataset. The nodule is segmented out in slice–per–slice basis, that is, we first process each CT slice separately to extract two dimensional (2D) contours of the nodule which can then be stacked together to get the whole 3D surface. The extracted 2D contours are optimal as we utilize dynamic programming based optimization algorithm. To extract each 2D contour, we utilize a shape based constraint. Given a physician specified point on the nodule, we blow a circle which gives us rough initialization of the nodule from where our dynamic programming based algorithm estimates the optimal contour. As a nodule can be calcified, we pre–process a small region–of–interest (ROI), around the physician selected point on the nodule boundary, using the Expectation Maximization (EM) based algorithm to classify and remove calcification. Our proposed approach can be consistently and robustly used to segment not only the solitary nodules but also the nodules attached to lung walls and vessels.

Keywords: Lung nodule, Segmentation, Dynamic Programming, Expectation Maximization, Calcification Pattern

1. INTRODUCTION

Mortality due to lung cancer is a leading cause for the cancer related deaths in the country. One of the main cause for such a high rate of mortality is the fact that it is very difficult to detect malignant lung nodules. Usually by the time nodules are detected, it is too late. The nodules are either too large or too advanced to be effectively cured. Thus, there is a need for lung screening with the motivation for early detection of the malignant lung nodule at a stage where it can be effectively treated. Conventional chest X–rays (CXR)[1, 2] have been utilized for long time. However, CXR are of limited use as using CXR only large lung nodules can be detected. With the advances in the X–ray Computed Tomography (CT) technology, there is a potential for screening of nodules which can be malignant. Using thin section multi–slice helical CT (hCT) scans, it is now possible to detect nodules which are as small as 3 mm in diameter.[3, 4] Usage of high resolution CT (HRCT) image dataset allows for quantitative measurements, such as, size, shape and density, for each nodule to be made.

However, each helical CT scan of a patient leads to a volume consisting of 500 to 600 slices with 512×512 voxels in each slice. Thus, the advantages of having high resolution CT over CXR can be fast lost without the help of efficient image analysis and interpretation methods. Computer–assisted nodule detection[4–7] has already transformed the way lung cancer screening is done by providing better ways for visualization, detection and characterization of lung nodules.

Physical characteristics of the nodules,[8, 9] such as rate of growth, pattern of calcification, type of margins are very important in the investigation of the lung nodules. Every lung nodules grows in volume over time. However, the malignant nodules grow at an exponential rate, which is usually expressed as tumor's doubling time. Malignant nodules have a doubling time of between 25 to 450 days where as the benign nodules are stable and have a doubling time of more than 500 days.[8] In addition to the rate of growth of the nodules, the pattern

E-mail: {ningxu,ahuja}@vision.ai.uiuc.edu, ravi.bansal@scr.siemens.com

of the calcification is an important indicator whether the nodule is benign or malignant. Nodules which are centrally or diffuse calcified are usually benign.

Before the nodules can be characterized, it is necessary to detect them in the volume of 3D CT image dataset that is being acquired. Manual lung nodule detection, which was possible while using CXR, is no longer possible. It is necessary to have automated tools that can assist a physician in quickly detecting the nodules. A number of automated lung nodule systems have already been proposed in the literature.[4, 6, 10–15] While the automated detection of the lung nodules is very important task, segmenting the nodules once they have been detected remains to be equally challenging task. The difficulty of the task comes from the fact that some of the nodules maybe sitting on the chest wall or on the lung vessels. Accurate and consistent segmentation of the lung nodule over time acquired CT volume datasets is necessary to study the rate of growth of the nodules and hence to predict whether the nodule is malignant or benign.

In this paper we present an automated method of segmenting the lung nodules from CT images on a per slice method. Once the nodules have been segmented in each slice, a 3D surface of the nodule can then be reconstructed using surface reconstruction methods.[16] Our approach allows for human interaction to correct for any errors in the automated technique. In addition, our method allows, naturally, to extract a pattern on the calcification of each nodule which can further be used to classify the nodules.

2. BACKGROUND

Since accurate and robust segmentation of lung nodule is very important in accurately measuring the volume, various of methods to segment nodules have been proposed in the literature.

Fuzzy clustering algorithm[14] is used to extract both the lung and pulmonary blood vessel region and then use feature based diagnostic rules to determine the candidate nodule locations. Zhao et al.[17] propose a method to process the CT volume data in a slice–by–slice manner. They also propose a set of multiple criteria, such as density, gradient strength and a shape constraint, to separate the nodule from the surroundings. In their approach, a whole region of interest need to be selected prior to the segmentation. They extend their approach to segment nodules directly in 3D volume.[18] Fan et al.[6] propose an adaptive histogram threshold followed by connected component analysis based approach to detect and segment isolatory nodules. A series of 3D morphological operators[5, 19, 20] are proposed to segment the nodules from the lung vessels. Armato et al.[15] utilize a *rolling ball* algorithm to define the segmented lung region after the lungs have been segmented from the thorax region using gray–level thresholding. After that Armato et al. utilize a 10–point connectivity scheme to identify contiguous 3D structures.

While all the approaches in the literature have their merits, there is no one single approach which can robustly and consistently segment all types of nodules, that is, nodules on the lung walls, nodules attached to the vessels or the isolated nodules. In this paper we present a shape based approach which can segment lung nodules consistently and robustly. Our approach uses dynamic programming[21, 22] to find the optimal lung nodule boundary. However, due to the calcification of the lung nodules, many times the initialization of the shape based approach fails. Thus, to overcome this problem we first pre–process the CT volume dataset, locally around the nodule selected by the physician, using *Expectation Maximization* (EM)[23–25] algorithm to detect and remove the calcification from the nodules. This approach can also feasibly give information about the pattern of calcification of the nodules. Our approach is completely automated and requires only an initial point from the physician on the surface of the nodule. However, since any completely automated algorithm may fail in a complicated case, in our approach we have provided efficient user interaction for corrections of the segmentation results, if desired.

3. OUR APPROACH

3.1. A shape model based framework for segmenting object boundary

In our framework, the object to be segmented has a defined shape, such as a circle or an ellipse. The task of segmentation is to estimate the parameters $\beta = [O, s]^T$ of the shape in an image and estimate the optimal boundary based on these parameters. Here O is the position of the shape and s is the scale of the shape. Since

the real world objects are far more complicated than those simple shapes, we need to allow certain degree of shape morphology. Also, in real world images, object may be occluded, or some part of the object boundary may be indistinguishable from the background. All these make the object boundary to be discontinuous. In our framework, we are trying to solve the segmentation problem of objects with discontinuous boundaries. An object boundary, B, is represented as $B = (\bigcup_i B_{ci}) \bigcup (\bigcup_j B_{dj})$, where B_{ci}, for $i = 1, \ldots, M$, represents parts of the visible object boundary and B_{dj}, for $j = 1, \ldots, N$, represents the parts of the object boundary which are occluded. Note that we model the object boundary to be a union of $N + M$ contours where some, N, of the contours are visible and the other M are occluded, forming discontinuities in the object boundary. We initialize β and then determine the discontinuous parts of the object contour, find out the optimal segments of the continuous parts and estimate β based on our results. This procedure is carried out iteratively until β converges. Then the estimated segments of the continuous parts of the object boundary, i.e. B_{ci}'s, are linked by straight lines to give an optimal estimate of the object boundary. The whole procedure is described as follows:

1. Initialize β;

2. Determine discontinuities in the object boundary;

3. Find out the optimal realization for continuous parts:

$$\hat{B}_{ci} = \arg\max_{B_{ci}} p(B_{ci}|I, \beta)$$

4. Link all those continuous parts, $B_{ci} \forall i$, using straight lines to get the object boundary along the discontinuous parts, $B_{dj} \forall j$, of the object boundary;

$$\hat{B}_{di} = f(B_{ci})$$

5. Estimate new parameters of the shape

$$\hat{\beta} = \arg\min_{\beta} g(\beta, B),$$

where $g(\beta, B)$ is a distance measurement between the resulting object boundary B and the shape β;

6. If β converges then stop; otherwise go to step 2;

In our nodule segmentation problem, we model a nodule as a circle shaped object with some deformation and iteratively apply dynamic programming algorithm to find out the optimal contour of the nodule. After each iteration, we fit a circle onto the result and estimate the center and radius of the circle. The parameters β of the circular shape are center O and radius r. The probability $p(B_{ci}|\beta, I)$ is set as

$$p(B_{ci}|\beta, I) = \frac{1}{C_f} \exp(-f_{\beta}(B_{ci})),$$

where

$$f_{\beta}(B_{ci}) = w_{shape} \cdot f_{shape}(B_{ci}) + w_{smooth} \cdot f_{smooth}(B_{ci}),$$

and

$$f_{smooth}(B_{ci}) = \sum_{B_{ci}} (w_1 |B_{ci}^k - B_{ci}^{k-1}|^2 + w_2 |B_{ci}^{k+1} - 2B_{ci}^k - B_{ci}^{k-1}|^2)$$

and $f_{shape}(B_{ci}) = \frac{1}{N_c} \sum_k f_{shape}(B_{ci}^k)$, where N_c is the number of pixels on B_{ci}. f_{shape} consists of four parts as

$$f_{shape}(B_{ci}^k) = w_z f_z(B_{ci}^k) + w_g f_g(B_{ci}^k) + w_d f_d(B_{ci}^k) + w_v f_v(B_{ci}^k),$$

where f_z is a Laplacian zero-crossing measurement at B_{ci}^k

$$f_z(B_{ci}^k) = \begin{cases} 0, & \text{if} & \nabla^2 G \otimes I(B_{ci}^k) = 0 \\ 1, & \text{if} & else \end{cases},$$

f_g is the gradient magnitude of pixel B_{ci}^k

$$f_g(B_{ci}^k) = 1 - \frac{\|\nabla G \otimes I(B_{ci}^k)\|}{\max_i \|\nabla G \otimes I(B_{ci}^k)\|},$$

f_d is the gradient direction of pixel B_{ci}^k

$$f_d(B_{ci}^k) = \arccos(\frac{< \nabla I(B_{ci}^k), B_{ci}^k - O >}{\|\nabla I(B_{ci}^k)\| \cdot \|B_{ci}^k - O\|}),$$

f_v is a heuristic gray value criterion based on the shape and the intensity of pixel B_{ci}

$$f_v(B_{ci}^k) = \|I(B_{ci}^k - k(O - B_{ci}^k)) - I_{in}\| + \|I(B_{ci}^k + k(O - B_{ci}^k)) - I_{out}\|,$$

where I_{in} is the intensity value inside the object and I_{out} is the intensity value outside of the object.

3.2. Computing optimal object boundary using dynamic programming

Given initial parameters or estimated parameters from previous iteration, i.e. β, the following steps estimate the optimal nodule boundary:

1. Detect discontinuous parts of the object boundary,

2. estimate $B_{ci} \forall i$ using dynamic programming,

3. link all segments of continuous parts by straight lines to obtain the optimal object boundary, B. Then, $B_{dj} \forall j$ are the straight lines required to link $B_{ci} \forall i$ to form a closed nodule boundary.

3.2.1. Detecting discontinuity directions

From the center of the given circle, defined by β, we draw N rays which divide the circle into N equally spaced directions. The number N is set according to the length of the radius R of the circle. An area of interest is set to be a ring shaped area whose inner circle radius is $R - r$ and outer circle radius is $R + r$. Both inner circle and outer circle have the same center as the original circle. For each direction i, we check each pixel p from $R - r$ to $R + r$ and evaluate

$$f_i(p) = w_z \times f_z(p) + w_g \times f_g(p) + w_d \times f_d(p).$$

We set a threshold $\mathcal{T}$ and if $\min_p f_i(p) > \mathcal{T}$, there is a discontinuity on the object boundary in this direction.

This first step will detect object boundary discontinuities directions around the circle defined by β. If the object boundary is detected to be discontinuous, the dynamic programming procedures in the next step are then well defined. If no discontinuities in the object boundary are detected, we can either detect the weakest radial direction or detect a direction with a very high gradient pixel that can be a starting pixel and break the circle at this direction. Again, the dynamic programming procedure in the next step is well defined.

3.2.2. Using dynamic programming to compute continuous parts segments of object boundary

For each segment of those continuous parts of the object boundary, we use time-delayed discrete dynamic programming to compute an optimal boundary segment B_{ci} $\forall i$. Suppose there are n directions within this segment, $v_1, v_2, ..., v_n$, and in each direction, there are m candidate pixels. For each direction $k > 2$, keep an $m \times m$ matrix for each entry within this direction.

$$f_\beta(v_1, v_2, ..., v_n) = f_{\beta 1}(v_1, v_2, v_3) + f_{\beta 2}(v_2, v_3, v_4) + ... + f_{\beta(n-2)}(v_{n-2}, v_{n-1}, v_n),$$

where $f_{\beta(k-1)}(v_{k-1}, v_k, v_{k+1}) = f_{shape}(v_k) + f_{smooth}(v_{k-1}, v_k, v_{k+1})$ for each direction $k > 2$, an m^2 matrix $m_k(v_k, v_{k-1})$ is calculated and the best value of v_{k-2} is kept together with the calculated value

$$m_k(v_k, v_{k-1}) = \min_{v_{k-2}}(m_{k-1}(v_{k-1}, v_{k-2}) + f_{\beta(k-2)}(v_{k-2}, v_{k-1}, v_k)).$$

Finally, the optimal path can be derived by back tracking from the last direction to the first direction. Once we have estimated each segment, B_{ci} $\forall i$, we link them using straight lines to get the final optimal object boundary. Again, note that since B_{ci} $\forall i$ are linked by straight lines to get the optimal closed boundary of the nodule, B_{dj} $\forall j$ are these straight lines.

3.2.3. User interaction

If the result is not satisfactory, user can interactively select a pixel p to be on the following resulting contours. No matter whether this pixel p is inside a continuous part or a discontinuous part, we just set $f_{shape}(p) = 0$, and set the shape function of all the other pixels at the same direction to be a very large value. Then the resultant optimal boundary will pass through this pixel p selected by the user.

3.3. Initialization and circle fitting

3.3.1. Initialization by blowing a circle

In our approach, initial center and radius, i.e. an initial estimate of β, is required before we can start. At the very beginning, we only need the user to select a point near the nodule boundary. This initializing point can be detected manually by an expert or by an automated algorithm. Then we get an estimate of the initial contour by blowing a circle from a nearby points with a highest intensity, which means the point should be a soft tissue point. We first use Canny's edge detection method to generate an edge map of the original image data and proceed with our circle blowing algorithm as follows:

1. Select a pixel P in a region with highest intensity, set $O^0 = P$;

2. Set $radius = 0$;

3. Set $radius = radius + 1$, check whether there is any point on the circle is an edge point in the edge map. If none, repeat this step;

4. Suppose a set of points $P_i, i = 1, 2, \ldots, m$, on the circle are edge points, and the shortest curve which consists of all these points is shorter than half a circle, then set

$$O^{n+1} = O^n + \frac{(O^n - (P_1 + P_m)/2)}{(\|O^n - (p_1 + p_m)/2\|)};$$

5. Go to step 3 until the shortest curve mentioned in step 4 is longer than half a circle;

3.3.2. Circle fitting

After we link all the continuous segments, i.e. B_{ci} $\forall i$, computed using dynamic programming to obtain a closed contour, we fit a circle to this contour by trying to find a center and radius which will minimize some kind of distance measurement. First, express all the pixels $p_i, i = 1, 2, ..., N$ on the contour to be $[x_i, y_i]^T$. To improve numerical stability, we need to subtract the mean of the pixels and then scale them by the range of the resulting values. The equation of a circle can be expressed as

$$a_1 x^2 + a_1 y^2 + a_2 x + a_3 y + a_4 = 0.$$

If given $[a_1, a_2, a_3, a_4]^T$, then

$$O = [-a_2/(2a_1), -a_3/(2a_1)]^T,$$

and

$$r = \sqrt{a_4/a_1 - \|O\|^2},$$

where O is the center of the circle and r is the radius.

In our approach, we use approximated Euclidean distance to measure the distance. If we try to minimize the Euclidean distance

$$\min \frac{1}{N} \sum_{i=1}^{N} \frac{f(R)^2}{\|\nabla f(R_i)\|^2},$$

the desired solution is the minimum eigenvalue of the system $L\vec{a} = \lambda S\vec{a}$, where

$$L = \frac{1}{N} \sum_{i=1}^{N} B(R_i) \cdot B(R_i)^T,$$

$$S = \frac{1}{N} \sum_{i=1}^{N} D(R_i) \cdot D(R_i)^T,$$

and

$$B(R_i) = [x_i^2 + y_i^2, x_i, y_i, 1]^T,$$

$$D(R_i) = \begin{pmatrix} 2x_i & 2y_i \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

3.4. Pre-processing using EM based classification

In our approach described above, a circle is blown from the initial physician selected point on the nodule boundary. The radius of the initial circle is slowly increased as proposed above. Such an approach leads to a nodule segmentation approach which is quite independent of the initial selected point on the boundary. However, a nodule can be calcified in a variety of locations, which will cause problems when blowing a circle or looking for optimal contours. To overcome this problem, we propose to first pre-process the CT volume, in the region of interest around the physician selected point using *Expectation Maximization* (EM)[24, 25] based algorithm. In the volume of interest, the voxel values can be due to three tissue types, air, soft tissue or calcified. Using EM algorithm, we classify each voxel to be one of these three classes. The voxels which are labeled as calcified are then removed from the volume and the removed voxels are interpolated from its neighborhood.

The EM algorithm has been successfully used to estimate parameters which maximizes the likelihood of the observed data, in the presence of missing data. The EM based maximum likelihood classification algorithm consists of two steps, the *E–Step* and the *M-Step*, as the name suggests. In the *E–Step*, the algorithm estimates the expected value of the missing information. In the *M–step*, the algorithm utilizes the expected values of the missing information from the *E–step* to estimate the optimal parameters. The two steps are computed iteratively till convergence.

3.4.1. Mathematical formulation

Consider random variables $\mathbf{x}$ and $\mathbf{y}$ to be defined over the sample spaces, $\mathcal{X}$ and $\mathcal{Y}$ respectively, and there is a many–to–one mapping from $\mathcal{Y}$ to $\mathcal{X}$. $\mathcal{Y}(\mathbf{y})$ denotes the subset in $\mathcal{Y}$ from which the observed data $\mathbf{x}$ can be obtained. The observed data $\mathbf{x}$, called *incomplete data*, is a realization of $\mathcal{X}$. The corresponding $\mathbf{y}$, called *complete data* in $\mathcal{Y}$ is not observed. Let $\mathcal{Y}(\mathbf{x})$ denote the subset in $\mathcal{Y}$ from which the observed data $\mathbf{x}$ can be obtained. If $p_{\mathbf{x}}(x|\theta)$ is the probability density function of the observed data over the sample space $\mathcal{X}$, and $p_{\mathbf{y}}(y|\theta)$ is the corresponding probability density function of the complete data over the sample space $\mathcal{Y}$, the two density functions are related as follows:

$$p_{\mathbf{x}}(x|\theta) = \int_{\mathcal{Y}(\mathbf{x})} p_{\mathbf{y}}(y|\theta) \, dy$$

Notice that for a given incomplete data specification $p_{\mathbf{x}}(x|\theta)$, there are many possible complete data statistics $p_{\mathbf{y}}(y|\theta)$ that will generate $p_{\mathbf{x}}(x|\theta)$. The EM algorithm aims at estimating the value of θ which maximizes $p_{\mathbf{x}}(x|\theta)$ given the observed data, but in doing so it uses the associated family $p_{\mathbf{y}}(y|\theta)$. Then, the two steps in the EM algorithm to compute the parameters θ are derived to be:

E Step: Compute

$$Q(\theta|\theta^k) = E[\ln p(y|\theta)|x, \theta^k]$$

where the expected value is evaluated with respect to the density function $p(y|x, \theta^k)$.

M step: Estimate

$$\theta^{k+1} = \arg\max_\theta Q(\theta|\theta^k)$$

The E–step requires evaluation of the expected value of $\ln p(y|\theta)$, the reason being to *integrate out* the missing data, and thus, to write the objective function only in terms of observed data and the parameters to be estimated. Taking the expectation with respect to the density function $p(y|x, \theta^k)$ is of core importance to the iterative nature of the algorithm. Note that as the expectation is taken with respect to the density function with parameters estimated from the previous iteration, θ^k, this helps give an iterative nature to the algorithm.

3.4.2. Classification using EM

In our model, we assume that the pixel intensities from the three tissue classes are Gaussian distributed. Thus, the problem of classification is then a maximum likelihood problem of estimating the parameters of these three Gaussian distribution. However, a nodule may or may not be calcified. Thus, in addition to classifying the voxels to a class, we also need to find out how many classes there are. In our case there can be either two or three classes only. To estimate both the number of classes and the parameters of the component mixture densities, we utilize fully Bayesian mixture modeling.[26] The basic formulation for observations x_i, using a mixture model can be written as

$$x_i \equiv \sum_{j=1}^{k} w_j \;\; f(.|\theta_j)$$

where the weight, w_j, are called the *component weights* such that $\sum_{j=1}^{k} w_j = 1$, θ_j are the component mixture density parameters and k is the number of components which are also unknown.

In our implementation of the EM algorithm to find the number of component densities k and the mixture density parameters, we always assumed that the number of components are three and then estimate the parameters for three mixtures from the given image data. If the mean of the distribution for the class *calcified* is in between the means for the other two classes, then we assume that there are only two classes in the mixture model. Otherwise there are three classes.

Once the pixels have been labeled to belong to one of the three classes, *soft tissue*, *air* or *calcified*, the pixels belonging to the *calcified* class are removed from the volume of interest and the corresponding pixels are replaced with bilinear interpolation of the neighboring pixels.
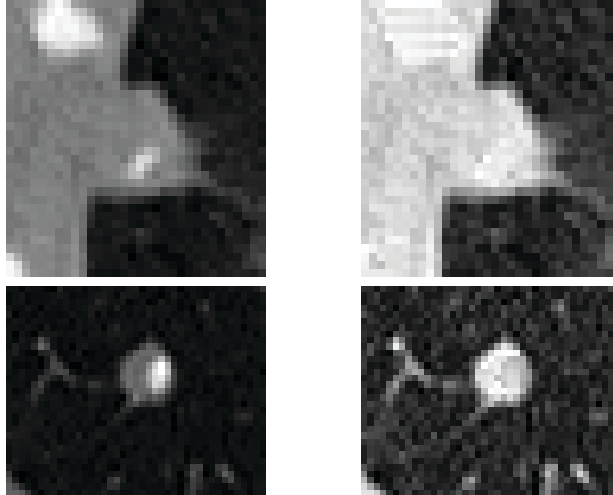
Figure 1. Images in the left column show the ROI before pre–processing using the EM based algorithm. Images in the right column show the results obtained after running the EM algorithm to classify and removed the detected calcified nodules.

4. RESULTS

In this section we present the results of our proposed lung nodule segmentation framework. Our algorithm starts with user selected point near the boundary of the nodule, in each slice in CT. Thus, our algorithm assumes that the nodule detection has already been done, either manually by an expert or by an automated algorithm. Once a user selects a point near the nodule boundary, we grow a circle to give us a rough initialization of the nodule boundary. This initial rough estimate of the nodule boundary forms the initialization of our algorithm. Note that this initialization makes our algorithm robust to the variability in user selection of the point. We tested our algorithm with various initial selected point near a nodule and our algorithm estimated the same initial circle. However, calcification of a nodule can effect the circle growing step. Thus, to overcome this problem, we first pre–process a region–of–interest (ROI) around the selected point to remove the calcification. Fig.1 shows the effects of removing the calcification in a nodule. The images in the left column of Fig.1 shows the ROI around the user selected point near two example nodules. The images in the right column of Fig.1 shows the results obtained after removing the calcification. Note that once the calcification has been removed, the pixel intensity within the nodule is more uniform. This estimated calcification of the nodules can then, in addition to the estimated volume and surface characteristics, be used to classify the nodule as benign or malignant.

Fig.2 shows the results obtained using our algorithm for a variety of nodules. These are the results obtained after running our algorithm completely automatically, with no corrections. In Fig.2, the initial expert selected point near the nodule boundary is shown as a single point. The initial grown circle is shown in black cross marks and the final nodule estimated boundary is shown as a white curve. In the images in the top and the third row show the segmentation results obtained for segmenting the nodules which are attached to lung wall. The images in third row show that even if a nodule appears as a small bump in the lung wall, our algorithm does a very reasonable job in segmenting it from the wall. Images in the second row of Fig.2 show the results obtained for segmenting the nodule that is attached to a vessel. Note that in this case the vessel is actually passing through the nodule. Still our algorithm is able to automatically segment the nodule from the vessel. The fourth row of the Fig.2 shows that we can segment the isolated nodule also with our proposed segmentation framework. These results show that visually our algorithm is able to segment the lung nodules accurately. However, in some of the obtained results, it feels that corrections might be needed.

We feel that a completely segmentation algorithm, including our proposed algorithm, might fail in some of the cases. This motivated us to provide a mechanism to an expert to correct an estimated nodule boundary. In
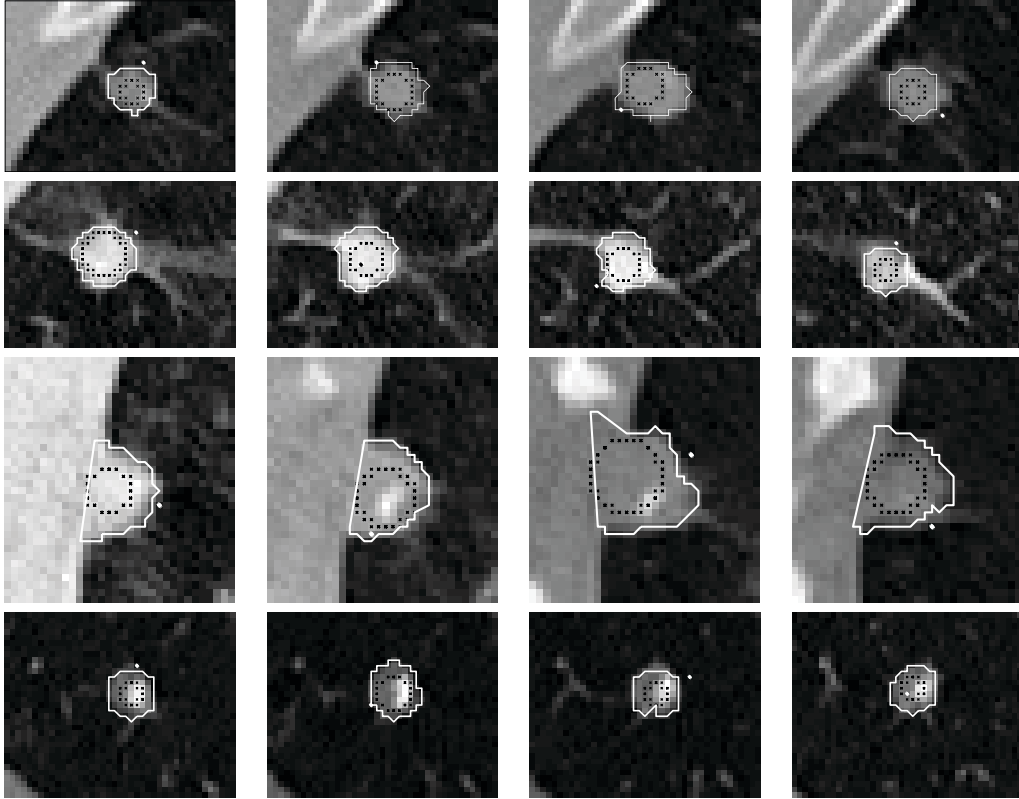
Figure 2. These results are obtained after an expert selects a point, shown as a single point near the nodule boundary. The initial blown circle is shown in black cross marks and the final estimated boundary of the nodule is shown as a white curve. The results shown are the estimated nodule boundary obtained after running our algorithm completely automatically, with no corrections. Note that while our algorithm did a very good job in segmenting the nodules, corrections might be needed in some of the results.

our nodule segmentation framework, corrections to the estimated boundary has been incorporated as a single-click mechanism. An expert selects a point through which the estimated nodule boundary must pass and our algorithm takes that as a hard constraint and automatically updates the estimated nodule boundary. Results obtained after expert corrections are shown in Fig.3. Images in Fig.3 show the results obtained after corrections to the results in Fig.2. Note that only one or two corrections, for only two images, were required. The final estimated boundaries are shown as white curve. Note that after these corrections, the lung nodule has been perfectly segmented in all the cases.

5. DISCUSSION

In this paper we presented an automated lung nodule segmentation algorithm where the segmentation of the nodule is carried out on a per slice basis from the initial physician selected point. We implemented our algorithm in Matlab, and for each slice, the automated algorithm took less than 5 seconds to compute the optimal contour. Once the 2D contours of a nodule have been segmented out of the CT volume dataset, the surface fitting algorithm[16] can be used to fit a surface. All the surface features have been preserved which can then be used to visually characterize a nodule. The preprocessing step using EM algorithm classifies those pixels within the region of interest as one of the three classes: soft tissue, air or calcified. This information can then, in addition to the volume and surface features of the segmented nodule, be utilized for further calcification of the nodule. Our proposed algorithm can be used to segment not only isolated nodules but also the nodules on the lung walls and the vessels without any modifications. As we utilize dynamic programming to estimate nodule boundary, based
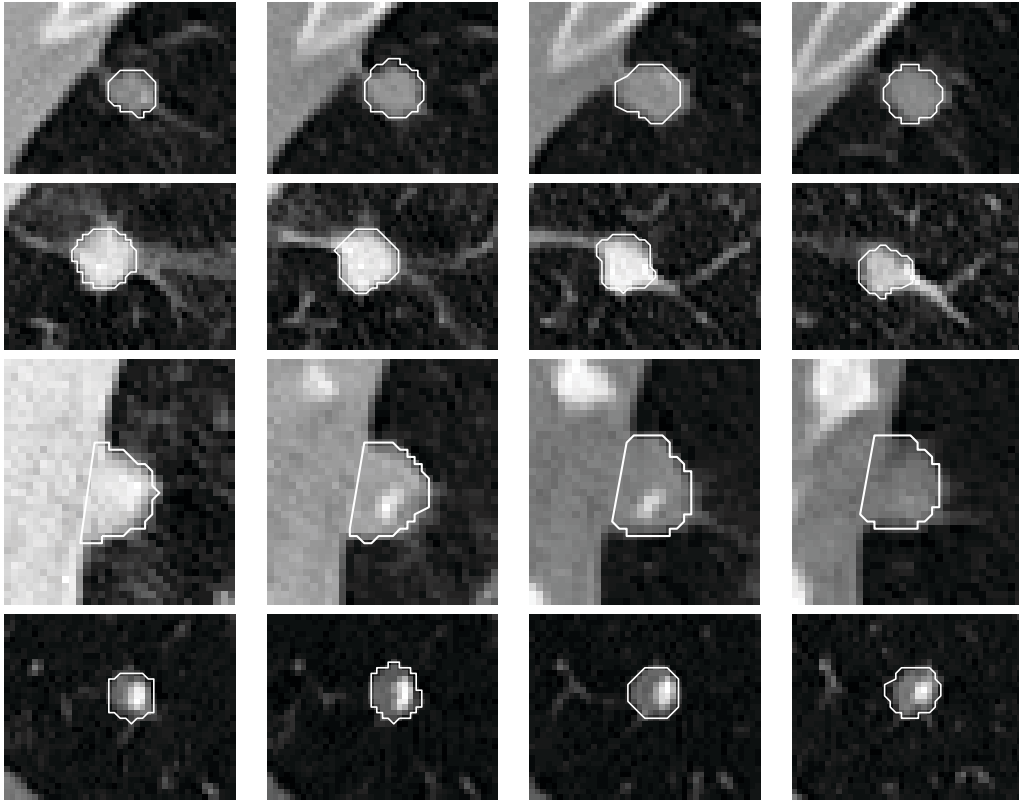
Figure 3. Final estimated lung nodule boundaries after corrections to the results obtained automatically, shown in Fig.2.

on external image information, such as gradient and the internal energy of the estimated contour, the estimated nodule boundary is optimal. For our future work, we will like to do further validations of our approach and to extend this method to directly segment nodules in 3D, i.e. without having to first segment 2D contours in each slice.

REFERENCES

1. M. Carreira, D. Cabello, *et al.*, "Computer-aided lung nodule detection in chest radiography," *Lecture Notes in Computer Science, Image Analysis Applications and Computer Graphics* **1024**, pp. 331–338, 1995.
2. B. van Ginneken, *Computer–Aided Diagnosis in Chest Radiography.* PhD thesis, University of Utrecht, 1970.
3. M. Brown, D. Aberle, *et al.*, "Model-based segmentation architecture for lung nodule detection in CT," *Radiology* **217**(P), 2000.
4. M. Brown, D. Aberle, *et al.*, "Computer-aided method for lung micronodule detection in CT," *Radiology* **217**(P), 2000.
5. A. P. Reeves and W. J. Kostis, "Computer–aided diagnosis of small pulmonary nodules," *Seminars in Ultrasound, CT, and MRI* **21**(2), pp. 116–128, 2000.
6. L. Fan *et al.*, "Automatic detection of cellular necrosis in epithelial cell cultures," *SPIE Medical Imaging* , Feb 2001.
7. P. F. Judy and F. L. Jacobson, "Evaluation of segmentation using lung nodule phantom CT images," *SPIE Medical Imaging* , Feb 2001.
8. G. A. Lillington, "Management of solitary pulmonary nodules," *Postgraduate Medicine* **101**(3), 1997.

9. S. G. Armato, M. L. Giger, and H. MacMahon, "Analysis of a three-dimensional lung nodule detection method for thoracic CT scans," *SPIE Medical Imaging* **3979**, pp. 103–109, 2000.

10. M. Penedo, A. Cabello, *et al.*, "Computed-aided diagnosis: A neural network based approach to lung nodule detection," *IEEE Transactions on Medical Image* **17**(6), pp. 872–880, 1998.

11. Y. Lee, T. Hara, *et al.*, "Nodule detection on chest helical CT scans by using a genetic algorithm," *Proceedings of the 1997 IASTED International Conference on Intelligent Information* , 1997.

12. G. Cox, F. Hoare, and G. Jager, "Experiments in lung cancer nodule detection using texture analysis and neural network classifiers," *citeseer.nj.nec.com/cox92experiments.html* , 1992.

13. G. Gonzalez *et al.*, "Application of computer-performed holographic recognition to lung nodule detection and evaluation in thoracic CT scans," *European Congress of Radiology (ECR)* , 2000.

14. K. Kanazawa *et al.*, "Computer–aided diagnosis for pulmonary nodules based on helical CT images," *Comput. Med. Imaging Graph.* **22**, pp. 157–167, 1998.

15. S. G. Armato, M. L. Giger, *et al.*, "Three-dimensional approach to lung nodule detection in helical CT," *SPIE Medical Imaging* **3661**, pp. 553–559, 1999.

16. B. Geiger, *Three–Dimensional Modelling of Human Organs and its Application to Diagnosis and Surgical Planning.* PhD thesis, INRIA, France, 1993. Number 2105.

17. Z. B, Y. DF, *et al.*, "2-D multi-criteria automatic segmentation of pulmonary nodules of helical CT images," *Medical Physics* **26**(6), pp. 889–895, 1999.

18. Z. B, R. A, *et al.*, "3-D multi-criteria automatic segmentation of pulmonary nodules of helical CT images," *Optical Engineering* , 1999.

19. W. Kostis *et al.*, "Three–dimensional segmentation of solitary pulmonary nodules from helical CT scans," *Proc. of Comp. Asst. Rad. and Surg. (CARS)* , pp. 203–207, 1999.

20. D. Yankelevitz *et al.*, "Small pulmonary nodules: Volumetrically determined growth rates based on CT evaluation," *Radiology* **217**, pp. 251–256, 2000.

21. A. Amini *et al.*, "Using dynamic programming for solving variational problems in vision," *IEEE Trans. on PAMI* **12**, September 1990.

22. D. Geiger, A. Gupta, L. A. Costa, and J. Vlontzos, "Dynamic programming for detecting, tracking, and matching deformable contours," *IEEE Trans. on PAMI* **17**, March 1995.

23. T. K. Moon, "The expectation–maximization algorithm," in *IEEE Signal Processing Magazine*, pp. 47–60, IEEE, November 1996.

24. A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via EM algorithm," *J. Royal Statistical Soc., Ser. B* **39**, pp. 1–38, 1977.

25. R. A. Redner and H. F. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Review* **26**(2), pp. 195–239, 1984.

26. S. Richardson and P. Green, "On Bayesian analysis of mixtures with an unknown number of components," *J. R. Statist. Soc. B* **59**(4), pp. 731–792, 1997.