Boundary Detection in 3D Ultrasound Reconstruction using Nearest Neighbor Map

P. C. Pedersen[a], V. Mitra[a] and J. Dey[b]
[a]Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, 100 Institute Rd., Worcester, MA 01609, USA; [b]Department of Radiology, University of Massachusetts Medical Center, 55 Lake Avenue North, Worcester, MA 01655, USA

ABSTRACT

Ultrasound imaging is a noninvasive technique well-suited for detecting abnormalities like cysts, lesions and blood clots. In order to use 3D ultrasound to visualize the size and shape of such abnormalities, effective boundary detection methods are needed. A robust boundary detection technique using a nearest neighbor map (NNM) and applicable to multi-object cases has been developed. The algorithm contains three modules: pre-processor, main processor and boundary constructor. The pre-processor detects the object(s) and obtains geometrical as well as statistical information for each object, whereas the main processor uses that information to perform the final processing of the image. These first two modules perform image normalization, thresholding, filtering using median, wavelet, Wiener and morphological operation, estimation and boundary detection of object(s) using NNM, and calculation of object size and their location. The boundary constructor module implements an active contour model that uses information from previous modules to obtain seed-point(s). The algorithm has been found to offer high boundary detection accuracy of 96.4% for single scan plane (SSP) and 97.9 % for multiple scan plane (MSP) images. The algorithm was compared with Stick's algorithm and Gibbs Joint Probability Function based algorithm and was found to offer shorter execution time with higher accuracy than either of them. SSP numerically modeled ultrasound images, SSP real ultrasound images, MSP phantom images and MSP numerically modeled ultrasound images were processed. The algorithm provides an area estimate of the target object(s), which along with position information of the ultrasound transducer, can be used for the calculation of the object volume(s) and for 3D visualization of the object(s).

Keywords: Boundary detection, visualization, nearest neighbor, segmentation and rendering, image filtering, active contour model, 3D ultrasound imaging, object detection.

1. INTRODUCTION

Ultrasound is a widely used imaging modality, which is finding new applications in areas such as emergency medicine and surgery. Along with these applications has come the need for 3D imaging and automated image analysis. Three-D imaging is used in obstetrics[1], neurosurgery[2,3], cardiac imaging[4], and prostate therapy[5]. Automated image analysis is carried out for ovarian ultrasound images, automated morphology analysis of vessels with intravascular ultrasound, and image sequence for dynamic studies of the left ventricles.

Boundary detection followed by image segmentation are necessary steps for the above-mention applications, in particular if an automated, real-time interactive 3D reconstruction is desired. The challenge to boundary detection is the often low quality of the ultrasound images, along with corruption due to speckle noise, shadowing, boundary drop-out and non-uniform brightness. Segmentation of fetal anatomical structures has been carried out using a low order parameterization of the contour shapes[6]. To visualize the complex 3D shape and motion of the left ventricle, a 3D extension of the 2D deformable model ("snakes") was developed[7]. Boundary detection has also commonly been applied to the prostate[8], and a semiautomatic prostate boundary segmentation algorithm using virtual operators has been developed[9].

The specific application of our boundary detection technique is in emergency medicine, specifically localization of abdominal free fluid using 3D imaging, which requires clear delineation of boundaries of the regions of interest. This application imposes several requirements. First, a given image may contain one or more 'objects' (= fluid-filled

regions); therefore, the boundary detection technique needs to be able to locate such objects automatically. Second, given that the boundary detection technique will operate on a sequence of adjacent scan planes, it needs to utilize scan plane-to-scan plane information to separate valid objects from artefacts. Third, *a priori* information regarding permissible contour shapes needs to be incorporated into the boundary detection technique.

This paper presents an automated boundary detection technique using nearest neighbor map (NNM) and active contour model. Cysts, free fluid regions and high scattering lesions were considered as the target objects, but the approach is generic enough to be applicable to any other type of abnormalities. The method can operate in a *Multiple Scan Plane* (MSP) mode or in a *Single Scan Plane* (SSP) mode. In the MSP mode, 'objects' (such as cysts or lesions) in individual scan planes are tracked from scan plane to scan plane, and are accepted or rejected as valid objects based on the variation in area and centroid location across scan planes. The primary goal of this automated technique is to produce interactive, automated 3D realization of objects.

The paper presents first a systems description of the *Nearest Neighbor Map* (NNM) based boundary detection, with explanation of the image and data flow for the MSP and the SSP cases. This is followed by a review of the individual processing steps: normalization, filtering including morphological closing and opening operation, nearest neighbor map processing, and boundary construction by a snake-based active contour model. Next, boundary detection results are presented for both the SSP and the MSP situations, with a quantitative evaluation of the boundary detection accuracy by comparison to the Sticks algorithm and the Gibb's Joint Probability Function algorithm.

2. COMMON BOUNDARY DETECTION METHODS

Edge detection algorithms typically look for image locations where the pixel intensity changes sharply, which is equivalent to the regions of large derivative values[10]. Thus, in principle, edge detection can be performed with a gradient operation. However, ultrasound images are often corrupted with speckle type noise, which will cause a simple derivative-based edge detection algorithm to fail.

2.1 Sticks Algorithm

The Sticks algorithm[10,11] is mainly designed to detect boundaries that appear as bright lines between speckle-filled regions of similar intensity. This algorithm uses an operator, called 'Sticks' operator, that operates parallel to the edges of the target object. The Sticks operators are usually line segments ("sticks") of differing length that are used to generate an outline of the region of interest. Each pixel of the image is projected onto a family of sticks that differs in orientation, but is centered at and passing through the pixel under consideration. The greatest total projection of any stick is plotted as the pixel intensity of the enhanced image at that point.

2.1. Gibb's Joint Probability Function based Algorithm

A different boundary detection algorithm is developed around the *Gibb's Joint Probability Function* (GJPF)[12], here referred to as the GJPF algorithm. As the first step, the ultrasound image is transformed by using a special function obtained from Gibb's joint probability function, which assigns a probability value to each pixel in the image, based on image pixel intensity, coordinates and texture characteristics, for the purpose of noise reduction. Based on the assigned probability values, the likelihood that a given pixel belongs to an object (such as a cyst) is then determined. In the second step, the resultant image is decomposed into its low resolution components by means of the discrete wavelet transform. The obtained low resolution image is then hard thresholded, median filtered and processed by a morphological opening and closing filter.

2.3 Active Contour based Algorithm

Snakes, also known as active contours, adapt to the edges in an image. The concept of active contour model was first proposed by Kass et. al.[13], and active contours are widely used for image segmentation, analysis of dynamic image data, and 3-D rendering of images[14,15,16]. An active contour model or 'snake' is an energy minimizing spline, whose energy depends upon its shape and location in the image. The local minima of this energy correspond to the desired image properties; hence it is a technique of matching a deformable model to an image by means of energy minimization. Snake uses *a priori* information to attain a desired appropriate solution. The energy function used in snakes is a

weighted combination of internal and external forces, where the internal force is generated from the shape of snake and the external force arise from the image.

3. SYSTEMS APPROACH TO NNM BOUNDARY DETECTION

3.1. Systems Description

The boundary detection system is modularized, as shown in Figure 1. The input image may be in the form of a *Single Scan Plane* (SSP) or in the form of a sequence of scan planes, referred to as *Multiple Scan Planes* (MSP). If only a SSP is processed, *the pre-processor* is not utilized, and the information flow is via *the main processor* and *the boundary constructor*.

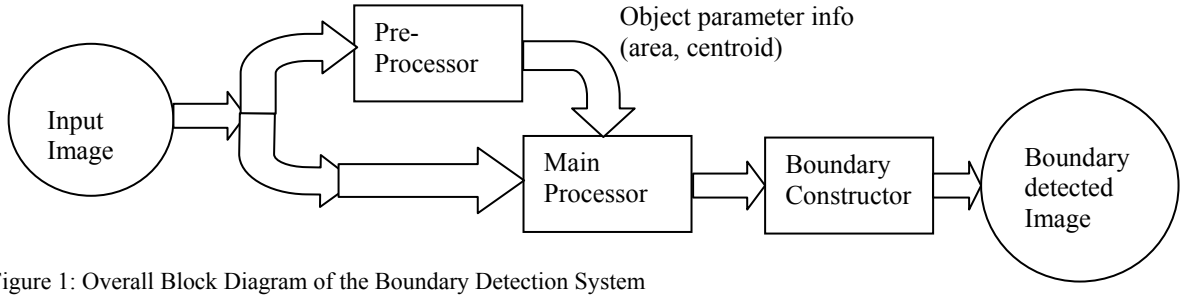

Figure 1: Overall Block Diagram of the Boundary Detection System

The pre-processor and the main processor modules contain exactly the same blocks, as shown in Figure 2, yet they perform very different functions. The pre-processor analyzes each scan plane separately for the existence of one or more 'objects', where an 'object' is a region in the image plane whose boundaries we wish to determine. Depending on the user specifications, an 'object' may refer to a bright region (*e.g.* a lesion) or a dark region (*e.g.* a cyst). For each object in a given scan plane, parameters such as the area and the centroid of the object are extracted. Tracking the change of area and the centroid location across many adjacent scan planes allows us to determine the likelihood of an object being an artefact or a true object. An artefact will manifest itself by rapid change in area and/or in rapid shift centroid location from scan plane to scan plane, and by only being present in a few consecutive scan planes. This type of processing is also beneficial for eliminating shadows arising from highly attenuating objects, where the effect of shadowing will be most pronounced when the transducer is centered over the object. Thus, the MSP case does not necessarily guarantee a more accurate boundary detection (except for shadowing), but it makes it likely that only true objects and not artefacts are included in the output.

The output of the pre-processor is a set of parameters, which informs the main processor module as to which objects for the MSP case are valid or true objects. This allows the main processor module to eliminate artefacts from the actual boundary detection processing. The blocks of the pre-processor and the main processor modules are shown in Figure 2; these blocks perform normalization and thresholding of images, noise filtering using median, wavelet, Wiener and morphological operation, object detection using nearest neighbor map (NNM), and finally estimation of object area and location across all scan planes.

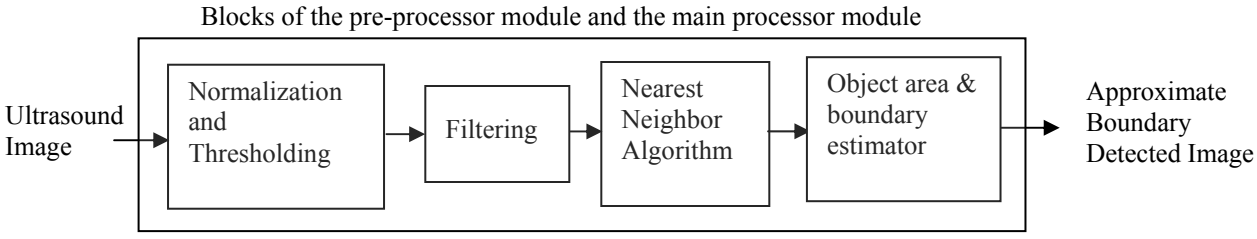

Figure 2: Blocks in the pre-processor and the main processor modules

Specifically, the pre-processor tracks the object within each scan plane by measuring the area and centroid location for each object. It can be assumed that under normal circumstances, the *centroid of the object* will not shift by more than ±10 pixels in x-coordinate or y-coordinate from scan plane *n* to scan plane *n+1*, and at the same time the *rate of change of area* from scan plane *n* to scan plane *n+1* is not greater than 60%. If these two assumptions are satisfied, then the object in scan plane *n* and in *n+1* is identified as the same object. After the object has been identified as the same object, then it is tracked to determine how consistently the same object is present over a sequence of scan planes. Specifying that a given object must be present in 80% of consecutive scan planes to be a valid object gives a reasonably good result.

3.2. Normalization and thresholding

The functional aspects of the individual blocks of the pre-processor and the main processor modules will now be described, and the process is illustrated by means of the appearance at each stage of a numerically generated ultrasound image, containing 3 lesions and 4 cysts. The original image is shown in Figure 3(a). In order to better illustrate the noise reducing effects of the different processing steps, the original simulated image is corrupted with 'salt and pepper' type noise, yielding the noise corrupted image, shown in Figure 3(b).

As seen in Figure 2, the first step is normalization and thresholding. The normalization ensures that the image conforms to the desired format, such as an 8 bit format, and the image may optionally be histogram equalized. The histogram equalization program first finds the max and min values of the pixel intensities in a single scan image and then normalizes the image by scaling it to the range 0-255. For example, if an image contains the max and min values of 160 and 34, respectively, then a pixel having intensity 65 will be normalized to (65 – 34) x [255/(160-34)].

A threshold operation is performed by calculating a threshold, defined in (1) and based on the mean, μ, and standard deviation, σ, of the given image.

$$t_h = \frac{\sigma}{4\mu}. \qquad (1)$$

If the goal is to detect fluid filled regions such as cysts, pixel values above the threshold are set to zero (= black), while pixel values below the threshold are increased by $(255 - t_h)$. The result is an inverted image with enhanced contrast. If the goal is to detect lesions or other regions that are brighter than the average, then pixel values *below* the threshold are set to zero (= black). The result of the threshold operation on the fluid-filled regions is shown in Figure 3(c).

3.3. Filtering Steps

The normalization and thresholding is followed several filtering steps. The first step is an optional discrete wavelet based filtering, based on the Daubechies mother wavelet, that performs 2 level Discrete Wavelet Transform (DWT) of the image and hence extracts and eliminates the noise that gets separated in the form of detail coefficients. The effect of this filtering can be seen by comparing the image in Figure 3(c) with the image in Figure 3(d). We have found that the presence or absence of the discrete wavelet filter does not make a big difference for the SSP and MSP simulated images; however, the wavelet filter does improve the boundary detection for noisy ultrasound images. Thus, the wavelet filter is retained to improve robustness, since it will increase the accuracy of the boundary detection for noisy images.

The median filter is a common filtering operation for edge detection, as it is effective in noise removal, yet preserves edge information. That this indeed is the case can be observed by comparing the image in Figure 3(d) with the image in Figure 3(e). For finer processing, the image is next filtered by Wiener adaptive filtering. Wiener filtering calculates the local mean and variance around a user specified neighborhood of each pixel, which is used to estimate the additive noise power of that neighborhood. A comparison between Figures 3(e) and (f) shows the additional noise reduction through Wiener filtering.

A morphological closing and opening operations needs to be performed on the image after Wiener filtering. Morphology is an image processing technique based on shapes, where each output pixel value of the image is obtained by comparing the corresponding pixel of the original image with its neighborhood pixels. Morphological operation masks out object of predefined shape and size. Usually, if we are considering objects with a representative dimension of

say 10-30 mm, then we can use morphological operation by specifying the masking function to eliminate any object that is smaller than 10 mm and greater than 30 mm.

Selecting a proper neighborhood size results in a morphological operation that is sensitive to specific shapes in the input image. Here, *morphological structuring elements* of circular shape and with a radius of 3 pixels were used. Morphological closing of the image results in dilating the image and then eroding the resultant image (image obtained from morphological closing) using a specific structuring element. The morphologically closed image is then opened by using the same structuring element. During this process of morphological opening, details with radius smaller than 3 pixels are removed.

3.4. Nearest Neighbor Map

After the filtering steps have been executed, the image is processed by a Nearest Neighbor Map (NNM) algorithm that utilizes a user defined neighborhood region relative to the given pixel, to calculate the probability of a pixel belonging to the target object. As can be seen by examining Figures 3(e) and (f), the NNM algorithm can smooth the edge contour of the objects in the image. Different parameter settings and selection of the shape of the shape of the NNM determines the degree of smoothing.

For a *rectangular* neighborhood, the output pixel values from the Nearest Neighbor Map (NNM) algorithm are calculated by using the following expression

$$a(i,j) = 1, \text{ when } \frac{1}{N}\sum_{r=i-n}^{i+n}\sum_{k=j-n}^{j-n} a(r,k) > t_h$$
$$a(i,j) = 0, \text{ when } \frac{1}{N}\sum_{r=i-n}^{i+n}\sum_{k=j-n}^{j-n} a(r,k) < t_h \qquad (2)$$

where t_h = threshold value, n = neighborhood distance (in number of pixels), N = total number of pixel inside the map, $a(r,k)$ = input pixel values and $a(i,j)$ = the resultant output pixel value. For rectangular maps, $n = 3 - 4$ with threshold = 0.4 – 0.45 gave good results, while for disc shaped maps $n = 3 - 5$ with threshold = 0.35 – 0.40 gave the best results.

Alternatively, the threshold for determining whether the output value of the pixel is 1 or 0 may be calculated dynamically, that is, based only on the pixels that are inside the nearest neighbor map. The threshold value in (3),

$$t_h = \frac{2(\mu - \sigma)}{\max}, \qquad (3)$$

has empirically been found to give good results, where μ = mean pixel intensity, σ = standard deviation of pixel intensity and max = maximum pixel intensity.

After processing by the NNM, the final boundary construction will operate on the objects that have been identified as valid objects. This is done by using a snake-based Active Contour Model (ACM), where three parameters must be specified: (i) the elasticity parameter, (ii) the rigidity parameter, and (iii) the viscosity parameter. Basic information about the AMC algorithm was given in Sect. 2.3. The model uses the object boundary information to initiate region growing. When the reconstructed object boundary points are sufficiently close to the boundary values, the ACM is stopped, and the reconstructed boundary points are used to construct the boundary of the object. Note that ACM can use only one seed point per object to initiate the region growing algorithm.

4. PERFORMANCE OF BOUNDARY DETECTION ALGORITHMS

The boundary detection system has been evaluated based on 4 different types of ultrasound images: SSP numerically generated ultrasound images, SSP actual ultrasound images, MSP ultrasound images obtained from an ultrasound phantom, and MSP numerically generated ultrasound images. The numerically generated images were produced by using Fields II[17]. In order to carry out a more objective evaluation of the NNM boundary detection algorithm, it was compared to a Sticks-based boundary detection algorithm and a Gibbs Joint Probability Function (GJPF) based algorithm, as described earlier in the paper. For this comparison, only real single scan planes (SSP) and real multiple

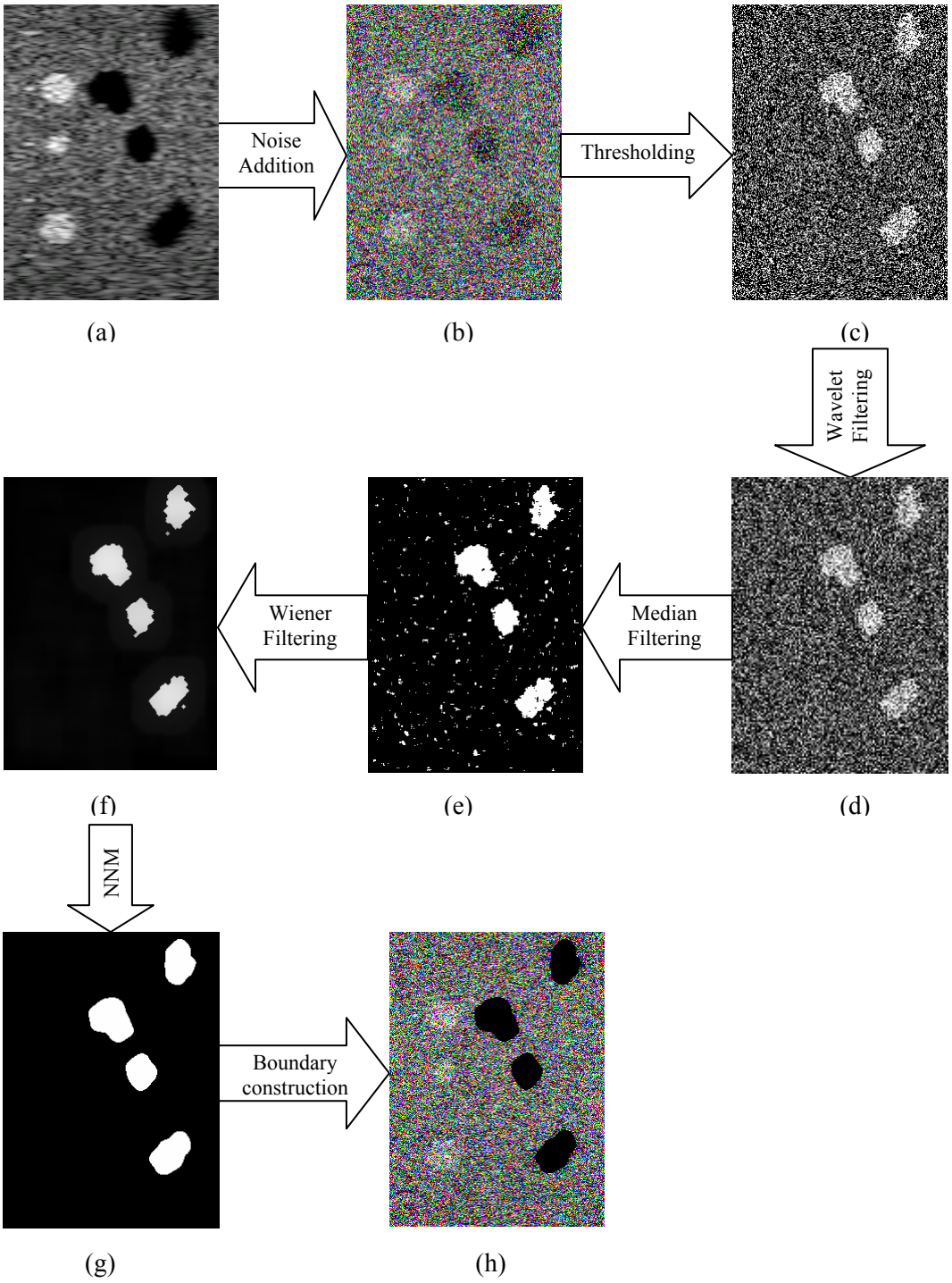

Figure 3. (a) Simulated ultrasound image, (b) noise added image, (c) thresholded image, (d) wavelet filtered image, (d) median filtered image, (f) wiener filtered image, (g) NNM processed image, (h) coarse boundary constructed image.

scan planes (MSP) ultrasound images were used, where the MSP images were separated into their individual frames for the processing by the Sticks algorithm and the GJPF algorithm.

Figure 4 shows the boundary detection results obtained from the NNM algorithm as well as from the Sticks algorithm and the GJPF algorithm, operating on an actual cyst image. Two types of noise are added to the images to better assess the performance of the proposed architecture in a noisy environment: Gaussian noise ($\mu = 0$ and $\sigma^2 = 0.05$) and Salt and Pepper (SNP) noise (noise density = 0.6). Figure 4(a) shows the original image, while the noise corrupted image is not shown. Figure 4(b), 4(c) and 4(d) show the boundary detected image using the NNM system, the boundary detected image using Sticks algorithm, and the boundary detected image using GJPF algorithm, respectively.

The next example demonstrates the ability of the NNM algorithm to ignore the effect of shadowing, when operating in MSP mode. The image sequence was obtained from an ultrasound phantom. As discussed earlier, removal of shadowing can only be achieved when the shadowing effect is present only in a subset of the sequence of image planes. Figure 5(a) and Figure 5(c) show two images from the sequence of images in the MSP sequence. The image in Figure 5(a) has only a negligible amount of shadow, while the image in Figure 5(c) has a significant amount of shadow. Yet, the NNM boundary detection algorithm reconstructs both boundaries perfectly, based on the centroid and area information across all the scan planes in the sequence.

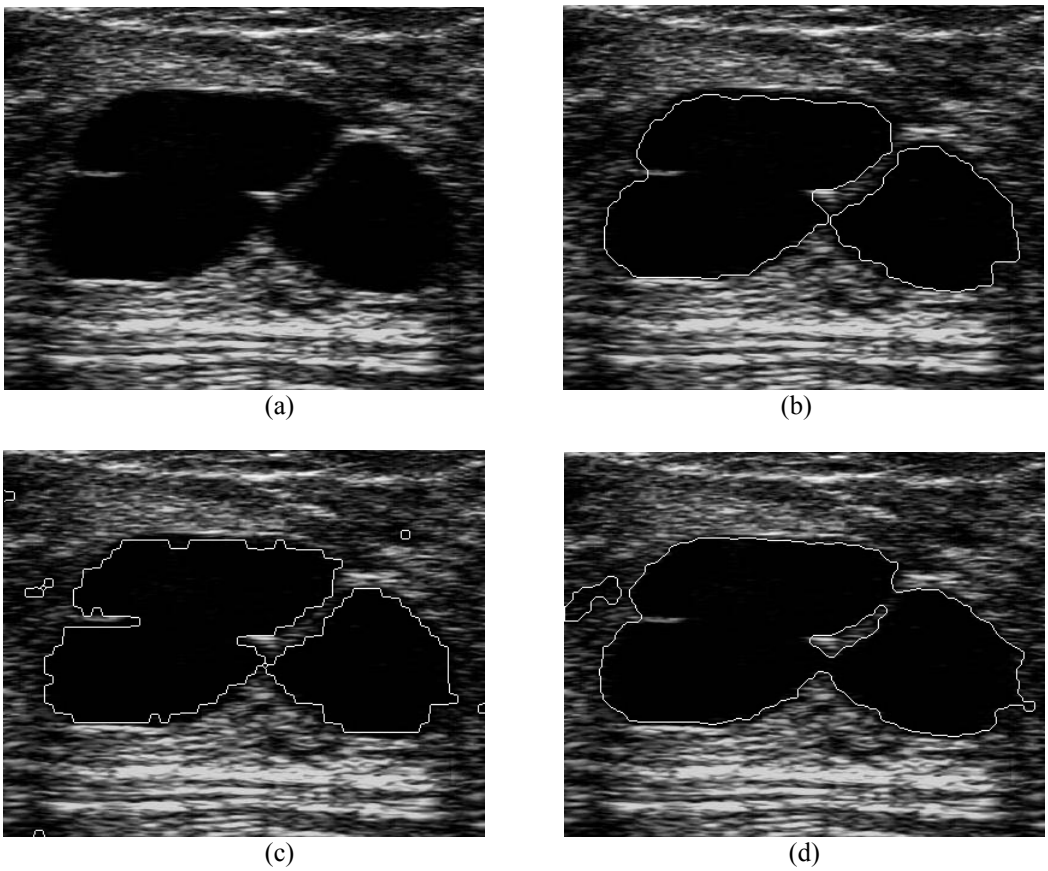

Figure 4. (a) Original real ultrasound cyst image, (b) boundary detected image using the NNM algorithm, (c) boundary detected image using Sticks algorithm, (d) boundary detected image using GJPF algorithm.

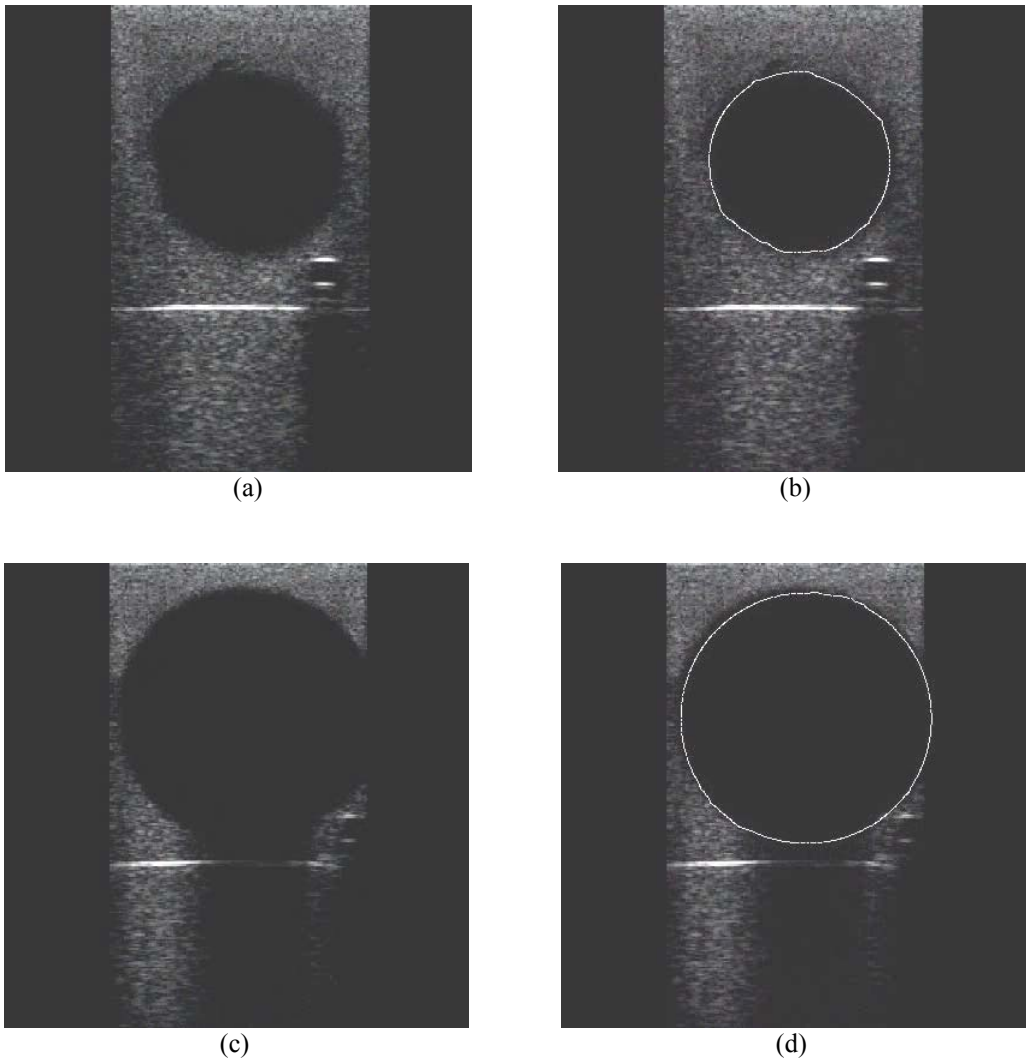

Figure 5. (a) SSP image from a MSP ultrasound image of an ultrasound phantom with no shadow, (b) boundary detected image using the NNM algorithm, (c) SSP image from a MSP ultrasound image of an ultrasound phantom with shadow, and (d) boundary detected image using the NNM algorithm.

The processing time has been evaluated for SSP cases, using the NNM algorithm, the Sticks-based algorithm and the GJPF algorithm. The results for a numerically generated image with noise added and for a real image (containing a larger number of scan lines) are presented in Table 1. As can be seen, the NNM algorithm performs 3 to 4 times faster than the comparison algorithms, evaluated in this paper.

Table 1. Execution time (in seconds) for the 3 boundary detection algorithms

	Nearest Neighbor based Algorithm	Sticks based Algorithm	GJPF algorithm
Numerical generated image + 'salt and pepper' noise	6	24	18
Real image	11	33	17

Of even greater importance than the execution time is the *boundary detection accuracy* (BDA). In order to correctly evaluate the accuracy, only numerically generated ultrasound images (with or without noise added) or images of ultrasound phantoms constructed in our lab, are used for this evaluation. As with the execution time evaluation, the accuracy of the NNM algorithm is contrasted with that of the Sticks-based algorithm and that of the GJPF algorithm.

Accuracy results are given in Table 2, for four different SSP cases and for one MSP case. The performance of the NNM algorithm is significantly better than those of the two competing algorithms, especially for the noisy images. The expression used to calculate the boundary detection accuracy for a given object is based upon the mean square error criterion. For a rectangular object of dimensions *M x N* pixels, the expression is presented in (4).

$$Accuracy = \left[1 - \frac{\sum_{r=1}^{N} \sum_{c=1}^{M} \left[b(r,c) - b_d(r,c) \right]^2}{\sum_{r=1}^{N} \sum_{c=1}^{M} b(r,c)} \right] \times 100\% \qquad (4)$$

where $b(r, c)$ gives the true area of the actual object and $b_d(r, c)$ gives the area of the boundary detected object. The accuracy calculation for objects with arbitrary contours is more complex and does not lend itself to a simple algorithmic expression. The obtained result from the 10 different simulated images is shown below in Table 2.

Table 2. Average boundary detection accuracy for the three different algorithms. The number in parenthesis indicates over how many scan planes the average was generated.

	Nearest Neighbor	**Sticks Algorithm**	**GJPF Algorithm**
Single scan Numerically generated image (12)	96.4	95.7	86.1
Single scan Num. gen. image (12) + Salt and Pepper noise	93.3	90.7	13.9
Single scan Num. gen. image (12) + Gaussian noise	92.5	83.3	78.0
Multi-scan Num. gen. image (18)	96.40	72.3	34.2
Multi-scan Phantom image (84)	97.9	65.2	43.3

5. DISCUSSION

The boundary detection method, based on the Nearest Neighbor Map, is a unique way to predict the probability of a pixel belonging to the target object, based upon the information obtained from the neighborhood pixels. Two types of map were considered – rectangular and disk shaped maps. The normalization, thresholding, filtering and the NNM

processing of the image, yielded a resultant coarse boundary, which greatly improved the precision of the boundary construction using active contour model, relative to using no processing before the active contour model.
Information from a sequence of consecutive scan planes was also incorporated. Analyzing a sequence of scan planes and hence tracking the locus of the object centroid not only helps to address shadowing in a unique way, but also helps tracking the changes in object contour, when the object is moving (*e.g.*, the heart). The area estimation of the object from each scan plane, along with the transducer position information (not discussed here), can be used to generate voxels necessary to create 3D volume rendering. The proposed system is not only robust against noise, but also addresses multi-object detection and makes possible a unique approach for 3D ultrasound image generation.

6. CONCLUSIONS

The NNM boundary detection algorithm performed much better than the two algorithms, chosen for comparison, both in terms of processing speed and boundary detection accuracy. It should be noted that the processing speed can be reduced significantly by the application of embedded system with a dedicated processor. The Sticks based algorithm performed well in certain situation, but suffered severely from low processing speed and jagged boundary constructions. MSP images are found to be beneficial for addressing shadowing phenomenon and also for eliminating unwanted artifacts. This research has shown that the information from adjacent scan planes can be successfully utilized to obtain a reliable object boundary. Object centroid and object area, coupled with transducer displacement information, can be used for 3D reconstruction of the object, thus introducing a novel approach for 3D ultrasound imaging.

ACKNOWLEDGEMENT

The support by the *Telemedicine and Advanced Technology Research Center* (TATRC) grant, DAMD17-03-2-0006, is gratefully acknowledged.

REFERENCES

1. I.E. Timor-Tritsch, and L.D. Platt, “Three-dimensional ultrasound experience in obstetrics,” *Current Opinion in Obstetrics & Gynecology*, **14**, pp. 569 - 575, December 2002.

2. G. Unsgaard, S. Ommedal, T. Muller et al, ”Neuronavigation by interoperative three-dimensional ultrasound: Initial experience during brain tumor resection,” *Neurosurgery*, **50**, pp. 804 – 812, 2002.

3. F. Lindseth, J. Bang and T. Langø, “A robust and automatic method for evaluating accuracy in 3-D ultrasound-based navigation,” *Ultrasound in Medicine and Biology*, **29**, pp. 1439 – 1452, 2003.

4. J.A. Panza, “Real-time three-dimensional echocardiography: an overview,” Int’l J. Cardiovasc. Imaging, **17**, pp. 227 – 235, 2001.

5. A. Fenster and D. Downey, “Three Dimensional Ultrasound Imaging of the Prostate,” *Proceedings SPIE - Medical Physics 3659*, pp. 2 - 11, 1999.

6. S. Jardim and M. Figueiredo, “Segmentation of fetal ultrasound images,” *Ultrasound in Med. & Biol.*, **31**, pp. 243 – 250, 2005.

7. V. Zagrodski, V. Walimbe, C.R. Castro-Pareja et al, “Registration-assisted segmentation of real-time 3D echocardiographic data using deformable models,” *IEEE Trans. Medical Imaging*, **24**, pp. 1089 – 1099, 2005.

8. S. D. Pathak, V. Chalana, D. R. Haynor and Y. Kim, “Edge-Guided Boundary Delineation in Prostate Ultrasound Images”, *IEEE Trans. Medical Imaging*, **19**, pp. 1211-1219, 2000.

9. H. Ladak, Y. Wang, D. Downey, and A Fenster, “Testing and Optimization of a Semi-Automatic Prostate Boundary Segmentation Algorithm using Virtual Operators,” *Medical Physics*, **30**, pp. 1637 - 1647, 2003.

10. R. N. Czerwinski, D. L. Jones and W. D. O'Brian, "An Approach to Boundary Detection in Ultrasound Imaging," *IEEE Ultrasonics Symposium*, pp 951 – 954, 1993.

11. R. N. Czerwinski, D. L. Jones and W. D. O'Brian, "Detection of Lines and Boundaries in Speckle Images – Application to Medical Ultrasound", *IEEE Trans. Medical Imaging*, **18**, pp 126 – 136, 1999.

12. A. Eslami, M. Jahed and M. Naroienejad, "Fully automated cyst segmentation in ultrasound images of kidney", *Conference on Biomedical Engineering, BIOMED 2005, IASTED*.

13. M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision,* **1**, pp. 321-331, 1987.

14. L. Cohen, "On active contour models and balloons," *Computer Vision, Graphics, and Image Processing: Image Understanding*, **53**, pp. 211-218, 1991.

15. M. Jacob, T. Blu and M. Unser, "Efficient energies and algorithms for parametric snakes," *IEEE Transactions on Image Processing*, **13**, pp. 1231-1244, 2004.

16. T.M. Lehmann, J. Bredno and K. Spitzer, "On the design of active contours for medical image segmentation - A scheme for classification and construction," *Methods of Information in Medicine*, **42**, pp. 89-98, 2003.

17. J. A. Jensen, "Users' guide for the Field II program", Technical University of Denmark, 2800 Lyngby, Denmark. http://www.es.oersted.dtu.dk/staff/jaj/field/users_guide.html