BOUNDARY ESTIMATION IN ULTRASOUND IMAGES

W J Lin[1], S M Pizer[1,2,3], and V E Johnson[4]

Departments of [1]Computer Science,
[2]Radiology, and [3]Radiation Oncology
University of North Carolina
Chapel Hill, NC 27599-3175

[4]Institute of Decision Science and Statistics
Duke University
Durham, NC 27706

Abstract

Surface definition, a process of defining three dimensional surface from volume data, is essential in three dimensional volume data rendering. The traditional method applies a three dimensional gradient operator to the volume data to estimate the strength and orientation of surface present. Applying this method to ultrasound volume data does not produce satisfactory results due to noisy nature of the images and the sensitivity of certain signals to the direction of insonation. We propose a Bayesian approach to the surface definition problem of ultrasound images, and study this approach in two dimensions. We formulate the problem as the estimation of posterior means and standard deviations of Gibbs distributions for boundary believability and normal direction. A set of filters of directional derivatives of Gaussians are used to measure the edge strength and orientation at multiple scales. The likelihood function is based on the measurement at the smallest scale. The prior distribution reflects shape properties at multiple scales. It uses a pyramid algorithm for contour analysis where the lengths of contours are computed and contour gaps are closed at multiple scales. The outcome of the pyramid algorithm is the length and weight global attributes for each pixel. These attribute values are incorporated into the Gibbs prior using a data augmentation scheme. The design and implementation of such an approach are the subject of this paper.

Keywords

Volume rendering; surface estimation; Bayesian; likelihood function; prior; posterior; Markov random field; Gibbs distribution; data augmentation.

1. INTRODUCTION

Three dimensional arrays of digital data are being generated in many areas of medical imaging in ever increasing number. Multiple 2D slices of computed tomography (CT), magnetic resonance (MR), and single-photon emission computed tomography (SPECT) create volume data. A research project currently conducted in the Duke/UNC Engineering Research Center on Emerging Cardiovascular Technologies includes building a new generation transducer that can capture a three dimensional volume of ultrasound data in real time (von Ramm et al, 1988), (Shattuck et al, 1984). These volume data represent complex anatomy or functional process under study. Effective visualization of these volume data helps physicians in diagnostic interpretation or treatment planning.

Volume rendering, a method of direct rendering of volume data, has been successfully applied in visualizing volume data of CT, MRI, and PET images. The operational principle of volume rendering is to render the volume data directly instead of fitting geometric primitives and then rendering the primitives. This direct rendering is done by compositing images from the results of two separate and parallel processes. The first process performs surface classification to obtain a partial opacity for every voxel. The second process performs surface shading at every voxel of volume data with a locally computed surface normal. Non-binary classification increases the likelihood that small or poorly defined features are perserved (Levoy 1988).

Successful application of volume rendering depends heavily on the estimation of the local surface normal and surface classification. Usually one tries to take advantage of knowledge about the relationship of voxel values with surfaces and their normals. For example, in CT image studies

one can make the assumption that CT numbers represent the percentages of material contained in voxels; hence the surface normal and surface classification can be obtained from the local gradient of an approximate percentage measure. Unfortunately this simple technique can not be satisfactorily applied to ultrasound volume data since ultrasound images suffer from serious speckle phenomena due to the coherent radiation source. Common speckle phenomena include random speckles spots from within soft tissues and broken contours on organ boundaries. Applying the above simple classification technique serves to pick up boundaries of random speckle spots as surfaces and to miss boundaries at contour gaps. The locally computed surface normals tend to be incorrect.

In this paper we study the surface classification and normal estimation problem from a Bayesian perspective in two dimensions. We formulate the problem as the estimation of posterior means and standard deviations of Gibbs distributions for boundary believability and normal direction. We show that the Gibbs distribution can be extended to model global structures by using a data augmentation scheme. We apply this method on ultrasound images in two dimensions and show the results. The remainder of this paper is organized as follows. The next section presents the Bayesian approach. The following two sections describe our filter design for producing edge-related measurements of ultrasound images to which Gibbs-compatible likelihood functions pertain. Then the design of Gibbs priors reflecting shape-related knowledge about object boundaries is presented, followed by a summary of the complete algorithm. We then present results and close with a discussion.

2. METHODS

2.1 Approach

Given an observed image we can apply measurements on the voxel values to determine whether a given voxel is on a boundary and, if it is, the associated normal direction. A representation of the target boundaries and normal directions on a grid can then be determined from the measurements. However, this representation will not always be the true representation of the boundaries and normal directions of underlying targets due to the image noise. Instead, there is a certain strength of conviction or believability associated with this representation.

The believability is obviously dependent on the measurements and is generally different in different parts of an image. We usually have strong conviction as to the representation of a part of an image when the measurements on that part show relatively high values and weak conviction when the measurement values are low. For example, the output of an edge detector affects our conviction about the presence of an edge — the higher the output value the stronger the conviction. This conviction is also affected by the measurements in the neighboring voxels. For example, the believability concerning the existence of a boundary at two neighboring voxels increases when their normal direction measurements are aligned and conform to the assumed boundary orientation but decreases when the normal directions are not aligned or contradict to the assumed boundary orientation.

Putting this dependency of believability on measurements and consistency into a mathematical form produces a Bayesian formulation for the conviction. In other words, the believability is really the posterior odds of a representation given measurements. It depends on a likelihood function, which is a distribution of measurements conditioned on a boundary representation, and a prior, which models the consistency in a representation, that is, the geometry of the situation.

Under the Bayesian framework we can define a random field with the same dimensions as the input image. Each site of the random field has two components: a *boundary value*, whose value is either zero or one, representing whether the corresponding voxel is on a boundary or not, and a *normal* which is the normal vector associated with the boundary. A representation of object boundary and normal direction is thus a sample drawn from the random field. In regard to this representation there is an associated posterior odds between zero and one, and the value is monotonic with the believability of this representation.

We compute the posterior means and standard deviations of boundary value and normal directions for each voxel. This produces a summary representation of the ensemble of all possible representations. This summary representation provides information useful for volume-rendering. For example, the mean of the normal direction of a voxel can be used as the surface normal for shading. The standard deviation of normal direction can be used to make the mean normal direction fuzzy before it is used as the surface normal. As a result a shiny surface might show a high confidence about the local surface normals while a dull surface show a low confidence. The mean of the boundary value can be used to modulate the opacity of the voxel and the standard deviation of the boundary value

can be used to modulate the color. Here the color of a surface would show the relative confidence about our judgement of the presence of a local surface. Ideally this creates an image portraying the target shapes with additional visual information of confidence about the shapes.

Later we will show that under certain assumptions this random field has the Markovian property. By the equivalence theorem of Markov random fields and Gibbs distributions due to Geman and Geman (1984), the posterior distributions of this random field is a Gibbs distribution. The Gibbs sampler can then be used to draw samples from the random field repeatedly. We thus can compute the ensemble posterior mean and standard deviation.

2.2 Filter Design for Boundary Measurement

We take as our likelihood function the distributions of edge measurements conditioned on the presence or absence of boundaries and, in the case of boundary presence, the true normal direction. The edge measurements, including edge strength and edge orientation, are outputs of applying a edge fileter to the observed image. They should have well separated edge stength distributions between the cases of boundary presence and boundary absence, with the edge orientation distribution peaking at the true boundary normal direction.

The design of the edge measurement filter is application dependent. In this paper we shall concentrate on ultrasound images in two dimensions. We shall discuss briefly the properties of ultrasound image noise and then introduce a set of filters for measuring boundary and associated normal directions based on those properties. We shall then present our measured distribution of the filter outputs and approximations to the filter output distributions.

Ultrasound imaging is an unusual imaging modality in that it uses a coherent radiation source. A sound beam is produced at a transducer and directed into targets. Echoes are generated when the incident beam hits reflectors along the path of sonic transmission. Basically there are two types of reflection, the *specular reflection* and the *non-specular reflection*. The specular reflection is responsible for major organ outlines seen in diagnostic ultrasound examinations. These reflections usually appear to be very bright in the image and they form mostly continuous boundaries. We shall call this kind of boundary the *specular boundary*. Specular boundaries tend to be broken where the boundary orientation is parallel to the beam direction and the reflection is directed away from the transducer, a phenomenon called *echo drop-outs*. Specular boundaries also appear to be broken where the random scatterers around the boundary happen to produce a destructive component echo which is out of phase with the specular reflection.

Non-specular reflections originate from random inhomogeneous continuum. They appear in the image as a texture of random bright spots, called *speckles*. These echoes are mostly due to density and elasticity variations in soft tissues (Waag et al, 1989), (Nassiri and Hill, 1986). The intensity distribution of these echoes usually approximates Rician distributions with different mean intensities for different tissue types. The interface of different tissues also forms a boundary and sometimes can be visually detected in the image. We shall call this kind of boundary the *texture boundary*. We are interested in both specular boundaries and texture boundaries.

We wish to design a set of filters which provide accurate edge response and orientation for both specular and texture boundaries. We have selected directional derivatives of anisotropic Gaussians as the filter kernels (Korn, 1988), (Canny, 1986). The rationale for this is as follows. Suppose we know the local boundary orientation. We can compute the average intensities on the two sides of the hypothesized boundary. The absolute value of the difference of the two averages then is a good measure of edge strength. We would like to get as many samples as possible on either side to get reliable statistical averages. But at the same time we would like to avoid the problem of mixing intensities from other nearby image structures. The directional derivatives of anisotropic Gaussians with the elongated side oriented along boundaries satisfy these constraints. In our implementation we use such filters of fixed widths but increasing lengths at multiple scales. The purpose of more elongated filters, which will be detailed in the section describing the prior specificaton, is to aid closing broken boundaries of different gap sizes. The shape of such a filter is shown in Figure 1.

Notice that these filters work for specular boundaries as well as texture boundaries. The edge measurements from the smallest size filter is the basis of the likelihood function. The size of the smallest filter is chosen such that 2σ's is roughly the speckle size. That gives roughly two speckle spots on either side and should be large enough for good statistical averages. For specular boundaries smaller filters can be used to get better localization of edges. In current implementation we use the same filter sizes for both kinds of boundaries. (Strictly speaking, the filter size should change with the orientations of edges since the speckles are themselves anisotropic and always lie perpendicularly
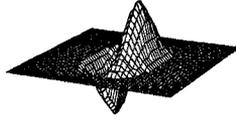
Figure 1: The shape of our filters for measuring edge strengths and orientations. The filters are directional derivatives of Gaussians. Multiple such filters of fixed widths and increasing lengths are applied at every pixel.

to the beam direction.)

We apply the filters in several orientations and averaging the responses from all orientations. In the current implementation we use four orientations at 0, 45, 90, and 135 degrees. The response from each filter is taken to form a vector in the polar coordinate system with its corresponding angle. These vectors are transformed to a new vector by doubling their angles, averaging by vector summation, and then halving the angle of resulting vector. This angle doubling and halving before and after the vector summation averaging is necessary because we do not distinguish inner normals from outer normals for shading's purpose.

2.3 Empirical Approximation of the Likelihood Function

The likelihood function that we need to estimate is $p(f_b(i,j), f_n(i,j)|B(i,j), N(i,j))$, i.e. the filter response at voxel (i,j) given the true boundary situation at voxel (i,j). f_b and f_n are the magnitude and phase of the smallest scale filter response, $B(i,j)$ is either 1 or 0 representing boundary presence or not, respectively, and $N(i,j)$ is the normal direction in the case of boundary presence. Both f_n and $N(i,j)$ are modulated by 180 degrees, since inner normals are not distinguished from outer normals for shading's purpose, and quantized to 16 discrete orientations for computational efficiency. Here We shall assume that $f_b(i,j)$ is independent of $N(i,j)$ to simplify the mathematical treatment, although We recognize its limitation on purely specular reflections. It then follows that we need only estimate $p(f_b(i,j)|B(i,j))$ and $p(f_n(i,j)|f_b(i,j), N(i,j))$ since

$$p(f_b(i,j), f_n(i,j)|B(i,j), N(i,j)) = p(f_n(i,j)|f_b(i,j), B(i,j), N(i,j))p(f_b(i,j)|B(i,j), N(i,j))$$

and $p(f_b(i,j)|B(i,j), N(i,j)) = p(f_b(i,j)|B(i,j))$ due to the assumption.

Three test images were used as the training set for determining these conditional distributions. These three images are a slice through a cone phantom simulating a diffuse target, a slice through a liver, and a slice of a baby doll hanging in a water tank, as shown in Figure 2. Contours in these images have been manually drawn as reference ground truth.

The histograms of f_b on the contours are computed and plotted in Figure 3. Regions in the images containing no boundaries were randomly selected to gather samples for computing the histograms of f_b in the absence of boundaries. These regions are marked in Figure 2 and the histograms are plotted in Figure 3. We have empirically fitted these histograms by normal distributions after a cube root transform. Figure 4 shows the transformed histograms and the fitting normal distributions. In these plots f_b is linearly scaled to between zero and one by normalizing by the maximal f_b in each image.

$P(f_n(i,j)|N(i,j), f_b(i,j))$ is estimated by the error in $f_n(i,j)$ with respect to $N(i,j)$, i.e., $p(h(f_n(i,j), N(i,j))|f_b(i,j))$, where $h(\cdot, \cdot)$ is a distance measure of normal angles. N(i,j) is computed from the drawn boundaries by fitting a straight line through five neighboring boundary voxels with least-square error. Figure 5 shows the histograms of $h(f_n(i,j), N(i,j))$. These histograms all peak at zero, indicating that our filters give approximately unbiased normal directions. These histograms also suggest normal distribution approximations.

Figure 6 shows scatter-plots of $h(f_n(i,j), N(i,j))$ versus $f_b(i,j)$. It appears that there is an exponential dependency relationship. We have approximated this dependency with an exponential function using a least-square-fit. Such an exponential function is shown in Figure 6 for the liver data set. The circular phantom does not show such a strong correlation between $h(f_n(i,j), N(i,j))$ and $f_b(i,j)$. This could be due to the facts that this image consists of speckles only and that there are relatively fewer sample points for this image. In summary, we have approximated $p(h(f_n(i,j), N(i,j))|f_b(i,j))$
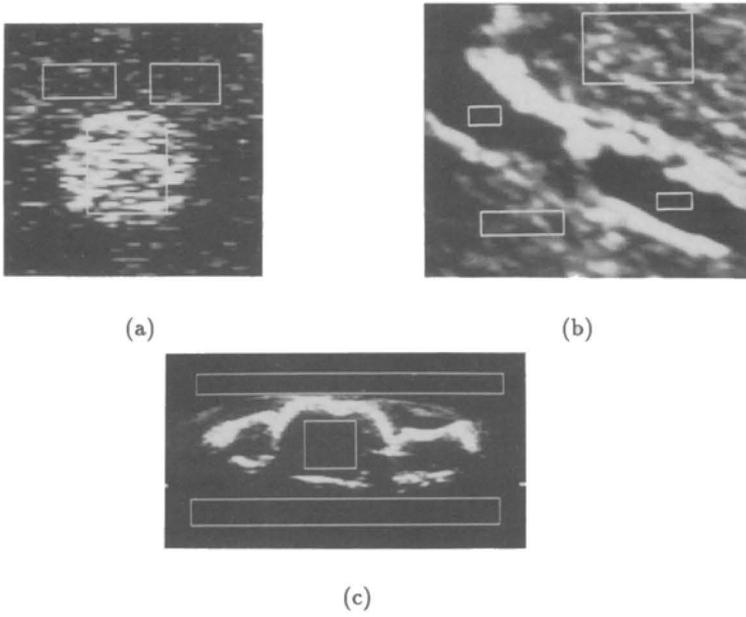
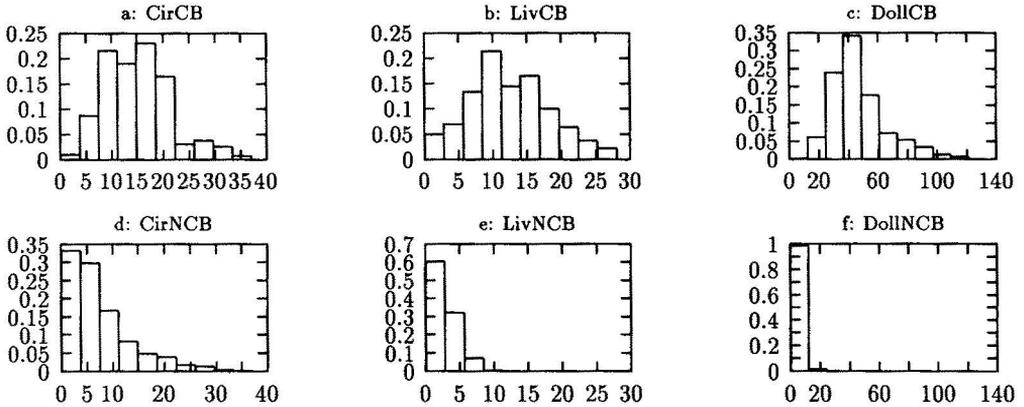**Figure 2:** Test images: (a) Cone phantom, (b) Liver, and (c) Baby doll.

Figure 3: (a), (b), and (c): Histograms of f_b measured on boundaries of cone phantom, liver, and baby doll. (d), (e), and (f): Histograms of f_b measured on no boundary regions of cone phantom, liver, and baby doll.
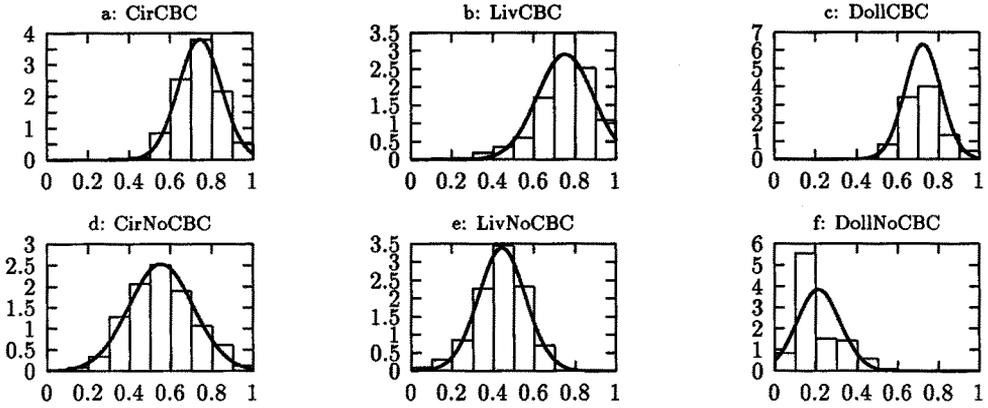
Figure 4: Histograms of f_b after cube root transform and normalization and the fitting normal distributions. Graph order is the same as in Figure 3.
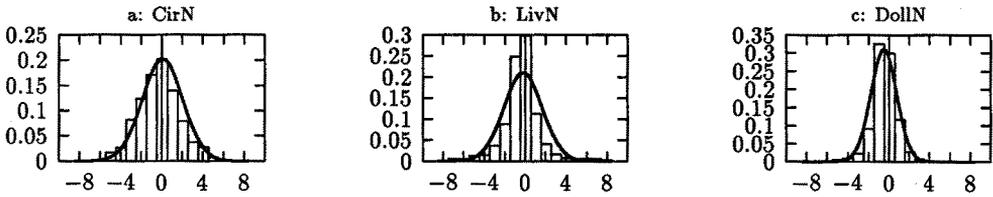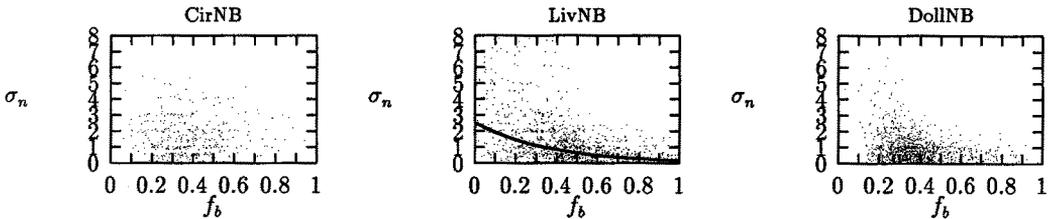

Figure 5: Histograms of f_n.

Figure 6: Scatter-plots of σ_n vs. f_b shows an exponential dependency. A least-square-fit gives $\sigma_n = 1.8 * \exp(-f_b/0.41)$, where f_b is the normalized edge strength measurement, for the liver data set.

with zero mean normal distributions with the variances depending exponentially on $f_b(i,j)$.

Finally, we have

$$
p(f_b(i,j), f_n(i,j)|B(i,j), N(i,j)) =
$$
$$
\begin{cases}
\frac{1}{2\pi\sigma_{b1}\sigma_n}e^{-V_{c1}} & \text{where} \quad V_{c1} = (f_b - \mu_{b1})^2/2\sigma_{b1}^2 \\
& \qquad\qquad\quad +h(f_n(i,j), N(i,j))^2/2\sigma_n^2 \quad \text{if } B(i,j) = 1 \\
\frac{1}{(2\pi)^{1/2}\sigma_{b2}}e^{-V_{c1}} & \text{where} \quad V_{c1} = (f_b - \mu_{b2})^2/2\sigma_{b2}^2 \qquad\qquad\qquad \text{if } B(i,j) = 0
\end{cases}
\tag{1}
$$

where $\mu_{b1}, \mu_{b2}, \sigma_{b1}$, and σ_{b2} are all estimated from the test data set. $\sigma_n = \sigma_n(f_b(i,j))$ is computed from the approximating exponential function.

2.4 Prior Specification

The prior defines the distribution of object shapes. Due to the enormous possibilities of object shapes projecting onto the random field it is difficult to directly specify the joint distribution of the large number of random components in the random field. However, since local object shapes do not have absolute correlation with distant parts of global shapes, it seems reasonable to assume a Markovian property for the random field. In other words, we assume that the values of a given site depend only on values of nearby sites.

Random fields having Markovian property are called Markov Random Fields (MRFs). Geman and Geman (1984) have shown that for a MRF an equivalent Gibbs distribution can be defined on the field. Let $S = \{s_1, s_2, \cdots, s_N\}$ be the set of grid points on a MRF. A *neighborhood system*, $\mathcal{G} = \{\mathcal{G}_s, s \in S\}$ for S, is any collection of subsets of S for which 1) $s \notin \mathcal{G}_s$ and 2) $s \in \mathcal{G}_r \Leftrightarrow r \in \mathcal{G}_s$, where $\mathcal{G}_s$ is the set of neighbors of s. A subset $C \subseteq S$ is a *clique* if every pair of distinct sites in C are neighbors. A *Gibbs distribution* relative to $\{S, \mathcal{G}\}$ is a probability measure π on Ω, the state space of the random field, with the following representation:

$$
\pi(\omega) = \frac{1}{Z}e^{-U(\omega)}
\tag{2}
$$

where ω is an outcome of the random field, Z is a constant and U, called the *energy function*, is of the form

$$
U(\omega) = \sum_{C \in \mathcal{C}} V_C(\omega)
\tag{3}
$$

where $\mathcal{C}$ is the set of cliques for $\mathcal{G}$. Each V_C is a function on Ω with the property that $V_C(\omega)$ depends only on those components x_s of ω for which $s \in C$. Such a family $\{V_C, C \in \mathcal{C}\}$ is called a *potential*. Z is the normalizing constant:

$$
Z \triangleq \int_\omega e^{-U(\omega)}
\tag{4}
$$

and is called a *partition function*.

We used a nearest-pair neighborhood system in modeling the local shape probability. There are four kinds of cliques in this neighborhood system: $(s_{i,j}, s_{i+1,j}), (s_{i,j}, s_{i,j+1}), (s_{i,j}, s_{i+1,j+1})$, and $(s_{i,j}, s_{i+1,j-1})$. For each kind of pair there are three combinations for the boundary values on the two grid points: both on boundaries, exactly one on a boundary, or neither on a boundary. For each case there are constraints on the normal directions for the points on boundaries: if both points are on boundaries the normal direction should change smoothly, while if only one point is on a boundary its normal direction should be oriented such that the assumed boundary direction does not point to the other point. The latter condition effectively penalizes blindly ending boundaries. Specifically, we have the following potentials for horizontal cliques:

$$
\begin{aligned}
V_{c2} = {} & B(i,j) * B(i+1,j) * h(N(i,j), N(i+1,j))^2/\sigma_{h1}^2 \\
& + (1 - B(i,j)) * B(i+1,j) * h(N(i+1,j), 0)^2/\sigma_{h2}^2 \\
& + B(i,j) * (1 - B(i+1,j)) * h(N(i,j), 0)^2/\sigma_{h2}^2
\end{aligned}
\tag{5}
$$

where $B(i,j)$ and $N(i,j)$ are the boundary value and normal direction at location (i,j), and $h(\cdot, \cdot)$ is the distance measure of two angles modulated by 180 degrees. σ_{h1} and σ_{h2} are scale parameters, which control how strongly the object shapes conform to the corresponding shape constraints. Potential functions for the other three cliques are similarly defined.
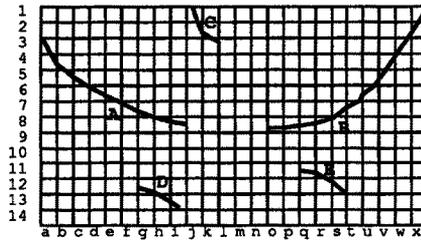
Figure 7: An example of current boundary estimation and the global attribute values. Please see text for the values of the global attributes.

This set of cliques can effectively capture the smooth property of surface normals and the unlikeliness of blindly ending boundaries. Nevertheless, it is insufficient for distinguishing true object surfaces from ones induced from erroneous filter response of random speckles. Furthermore, the broken surface speckle phenomenon may cause the filter to miss boundaries due to destructive echoes received at transducers. In either case we need to look beyond local neighboring properties to correctly determine the presence of true boundaries.

These problems can be overcome by analyzing object shapes in multiple scales to find support from increasing neighborhoods. Local boundary points suggested by the smallest scale filter are not to be regarded as true boundaries unless enough support is found from extended neighborhoods. Local non-boundary points are not to be regarded as non-boundaries if the global shape structures in some extended neighborhood strongly suggest that they are on broken boundaries.

The prior is designed with a component to capture global shape structures in multiple scales. It uses a parallel algorithm for the multiscale analysis and infers two global shape attributes, *length* and *weight*, for all points. The length attribute is the length of the boundary fragment of a boundary point. Boundary points induced from random speckles should have rather short length attribute values while those from true object boundaries should have longer length attribute values.

The weight attribute is designed to address the destructive echo problem on continuous surfaces. The weight of a point suggests the chances that point is actually on a boundary, and its value is between zero and one. The weight values are one for current boundary points and zero for non-boundary points not on boundary gaps. The weights of those points on boundary gaps are determined in two steps. First, a stochastic decision of whether to close the gap is made based on the configurations of the two matching ends and the filter measurements inbetween the matching ends. Then, if the decision is to close the gap, all points on the trajectory of the gap receive the same weight which is inversely proportional to the gap size. However, if the decision is not to close the gap, all points on the gap receive weight zero. The choice of a stochastic decision to close gaps, instead of a deterministic one, provides the capability to capture different closing alternatives when there are multiple possibilities.

Figure 7 gives an example of the length and weight attribute values. Curves A and B are part of a continuous curve with a gap while curves C, D, and E are random boundaries from speckles. The length attribute values are about 11 for all pixels covered by A, 13 for those pixels covered by B, 3 for those pixels covered by C, D, and E, and 0 for all other pixels. All pixels covered by any curve have a weight value 1. All other pixels have weight value 0, except those on the gap between curves A and B. For those pixels on the gap trajectory $((8,j)...(8,n))$, they will have the same weight value between 0 and 1. Our current implementation uses $1/(log(s/2) + 1.7)$ to compute this value where s is the gap size.

There are many choices for the gap-closing decision function and trajectory-computing algorithm. We propose a method to address the gap-closing and trajectory-computing problems in one unified approach. The idea is based on the active contour model where an energy-minimizing contour is sought given the image and internal constraints (Kass et al, 1988). For each potential gap, a spline is born to connect the two matching ends. The optimal positions of the spline is governed by the *image energy*, i.e., the filter measurements at the scale corresponding to the gap size and the internal bending and stretching energies of the spline itself. The filter measurements constrain the spline such that the spline is pulled towards pixels where the edge strength measurements are strong and the normal direction measurements conform to the local spline normal directions and is

pushed away from pixels where the edge strength measurements are strong but the normal direction measurements contradict to the local spline normal directions.

This algorithm not only finds the gap trajectory, but also provides the energy at stable state as the basis of gap-closing decision. Whenever the algorithm detects a potential gap a spline is generated and the optimal positions of the spline is computed. The total energy of the spline at the optimal positions is taken as a score, s. Then a random value, r, between zero and one is uniformly drawn and compared to e^{-s}. If $r < e^{-s}$, the gap is closed; otherwise the gap is not closed. When there are multiple possible matching ends for an end, each will have a chance to be closed with the chances decrease with increasing scores. This seems reasonable since the score inversely reflects the conviction of a boundary present at a gap where the likelihood function shows relative low confidence. Notice that this choice of the total energy is in favor of closing small gaps over large gaps when there are multiple choices.

The algorithm computes length and weight attribute values by building a pyramid of cells of 6 by 6 overlapping receptive fields (Shneier, 1981), (Meer et al, 1990). At the base of the pyramid is a "thinned" version of the current boundary estimations of the Gibbs random field. A thinning procedure is applied to reduce the boundary thickness to one pixel in order to correctly compute length and weight values in the pyramid operation. At each level in the pyramid, boundary fragments in child cells are joined together in the parent cell and their lengths are summed. Potential gaps between a pair of boundary fragments within the same receptive field are closed according to the active contour algorithm. A gap is closed at a level corresponding to its gap size. This algorithm then makes a downsweep of the pyramid, tracing down the exact boundary locations while propagating the length and weight attributes at the same time. Please see (Lin, 1991) for details of the thinning, gap-closing and pyramid algorithms.

With the values of the length and weight attributes, $L(i,j)$ and $W(i,j)$, we define the following potential function:

$$V_{c6} = B(i,j) * \left(\frac{A_{L1}}{L(i,j) + C_L} + \frac{A_{W1}}{W(i,j) + C_W} \right) + (1 - B(i,j)) * \left(\frac{L(i,j)}{A_{L2}} + \frac{W(i,j)}{A_{W2}} \right) \qquad (6)$$

where $A_{L1}, A_{W1}, A_{L2}, A_{W2}$ are scaling factors, and C_W, C_L are positive constants introduced to prevent infinite potentials when the attribute values are zeros. Although this potential function looks like a single element clique, it is actually derived from extended neighborhoods. The L and W in this potential are global shape attributes computed from the extended neighborhood, and serve as latent data in our data augmentation scheme to be described in detail in the following section. Without computing the L and W explicitly and viewing them as latent data, it would have been difficult to model the long range correlation in the Gibbs prior. By incorporating global attributes in a data augmentation scheme we have extended the use of Gibbs distribution to modeling global structures.

The parameters in the prior are determined empirically. There are two parameters in the local shape potential functions, i.e., σ_{h1} and σ_{h2}, and four parameters in the global attribute potential functions, i.e., A_{L1}, A_{L2}, A_{W1}, and A_{W2}. Their values are determined by assuming fixed energies for some typical configurations and are further fine-tuned by evaluating the results.

2.5 Architecture of the Solution Algorithm

Let X be the random field $X = \{x_{ij} = (B(i,j), N(i,j)), \forall i,j\}$. Let Y be the output of the smallest filter from the input image, i.e. $Y = \{y_{ij} = (f_b(i,j), f_n(i,j)), \forall i,j\}$. We shall assume the conditional distribution $p(y_{ij}|X)$ of observing filter output y_{ij} at voxel (i,j) given X depends only on x_{ij} and that the values of y_{ij} are independent of each other. We have made these assumptions to simplify the model. It follows from our prior specification and likelihood function that the data augmented posterior $p(X|Y, L, W)$ is a Gibbs distribution with the energy function:

$$U(\omega) = \sum_{C_i \in \mathcal{C}} \sum_{i=1,6} V_{c_i}(\omega) \qquad (7)$$

where V_{c1} comes from the likelihood function, $V_{c2}, V_{c3}, V_{c4}, V_{c5}$ come from the local shape constraints, and V_{c6} comes from the global shape attributes. The L and W act as latent data. They are not observed but have a distribution under the observed data Y. This distribution, $P(L,W|Y)$, is called a *predictive distribution*.

What we are really interested in is $p(X|Y)$. We use a data augmentation scheme, which is similar to an EM algorithm (Dempster et al, 1977), to iteratively calculate $p(X|Y)$ from the augmented posterior $p(X|Y, L, W)$ and the predictive distribution $p(L, W|Y)$ by alternatively sampling

these two distributions (Tanner and Wang, 1987), (Gelfand and Smith, 1990). The idea is as follows. First, we note the following relationship between the posterior density and the predictive density:

$$p(X|Y) = \int_{\Omega_{L,W}} p(X|Y,L,W)p(L,W|Y)dL,W \tag{8}$$

$$p(L,W|Y) = \int_{\Omega_X} p(L,W|X,Y)p(X|Y)dX \tag{9}$$

where $\Omega_{L,W}$ is the sample space of the latent data, L and W, and Ω_X is the sample space of X. Substituting Eq. (9) into Eq. (8) and interchanging the order of integration, we obtain the following relationship for $p(X|Y) = g(X)$:

$$g(X) = \int_{\Omega_X} K(X,X')g(X')dX', \quad \text{where}$$

$$K(X,X') = \int_{\Omega_{L,W}} p(X|Y,L,W)p(L,W|X',Y)dL,W. \tag{10}$$

This is an integral transformation and can be solved by the iterative method: start with any initial approximation $g_0(X)$ to $p(X|Y)$, and successively calculate

$$g_{i+1}(X) = (Tg_i)(X) \tag{11}$$

using Eq. (10).

The integral transform in Eq. (11) is usually difficult to calculated analytically. Monte Carlo method can be used here to perform the integration. Putting all the above together we obtain the following iterative scheme: given the current approximation g_i to $p(X|Y)$,

(a) Generate a sample $(L,W)^{(1)}, \cdots, (L,W)^{(m)}$ from the current approximation to the predictive density $p(L,W|Y)$.

(b) Update the current approximation to $p(X|Y)$ to be the mixture of conditional densities of X given the augmented data patterns generated in (a), i.e.,

$$g_{i+1}(X) = m^{-1} \sum_{j=1}^{m} p(X|(L,W)^{(j)}, Y). \tag{12}$$

In step (a), the sample (L,W) values can be generated in two steps:

(a1) Generate X from $g_i(X)$.

(a2) Generate (L,W) from $p(L,W|X',Y)$, where X' is the value obtained in (a1).

When m is large, the two steps (a) and (b) in the above iterative algorithm provide a close approximation to one iteration of Eq. (11). However, even when m is as small as 1, the iteration is still "in the right direction" in the sense that the average of $p(X|Y,L,W)$ over the augmented data patterns generated across iterations will converge to the true $p(X|Y)$.

In a typical Markov Random Field only the interactions among the neighboring sites are explicitly formulated. Remote sites have influence among each other only through a chain of intermediate sites. The interaction among remote sites can not be readily apprehended from a chain of local interactions. Thus it is difficult to model a desired global interaction through the specification of local interactions.

The data augmentation scheme provides a mechanism to employ Markov Random Field with global information. The global interaction among remote sites can be specified through an intermediate representation which encodes the global information from the observed data. This intermediate representation serves as the latent data and facilitates the analysis of the estimated quantities.

The operation of the complete algorithm is shown in Figure 8. The current boundary values and normal directions to be estimated are shown in the shaded area. The initial boundary values are obtained by applying a threshold on $f_b(i,j)$, and the initial values of normal directions are $f_n(i,j)$. The algorithm then makes repeated updates on the whole field. Each iteration includes successive updates to each voxel. A sample from $p(L,W|Y)$ is drawn by performing the pyramid operation on the the current configuration. The Gibbs sampler is used to produce a new sample x_{ij} for each voxel. It involves first computing the augmented marginal posterior $p(x_{ij}|X_{S/ij}, Y, L, W)$, where $X_{S/ij}$

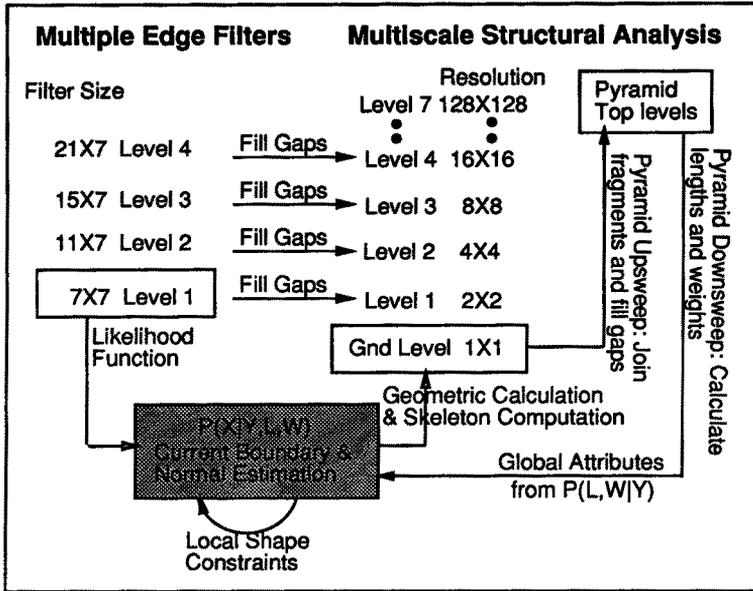**Multiple Edge Filters**   **Multiscale Structural Analysis**

Figure 8: Operation of the Solution Algorithm.

denotes the neighborhood of x_{ij}. Each $p(x_{ij}|X_{S/ij}, Y, L, W)$ is computed from Eqs. (2), (4), and (7) with only those cliques including voxel (i, j). A sample is then randomly drawn accordingly and replaces the old x_{ij}.

In the current implementation we compute the pyramid only up to the sixth level. At this level each cell covers a 32 by 32 pixel region. This size is considered large enough to compute reliable global attributes to reveal false boundaries. The gap-closing operation is only performed up to the fourth level. At this level, the maximal gap size that can be closed is roughly 45 pixels. The values of $B(i,j), N(i,j)$ and squared values of them are accumulated during each iteration. They serve to compute the means and variances of the boundary and normal estimations.

3. RESULTS

Figure 9 shows the magnitude output from the smallest scale filter, contrasted with those from a Sobel filter, which is a typical filter used in volume rendering. These pictures show our filter gives more accurate boundary measurements on the boundaries of the diffuse phantom and smooths out broken boundaries of the baby doll image.

Figure 10 shows the three test images and their corresponding mean boundary values displayed as images. The mean boundary values are obtained by averaging over 150 samples of binary boundary values. These samples are generated by the Gibbs sampler with the data augmentation scheme described in the previous section. The original images are displayed on the left and the mean boundary values on the right.

The boundaries in these result images seem to be quite thick. This is due to two factors. First, the filter is designed to produce correct boundary measurements on texture boundaries. It has a large support and produces significant magnitudes near boundaries. Second, these significant magnitudes near boundaries have higher magnitude than those measured on random speckles. The likelihood function thus tends to suggest those pixels as on true boundaries. These thick boundaries can be thinned by a thinning algorithm.

The result of the baby doll image (Figure 10a) shows enhancement of some weak boundaries which can not be clearly seen in the original images. Unfortunately, these boundaries are mostly acoustic artifacts. They can be removed by selecting a higher mean magnitude values in the likelihood
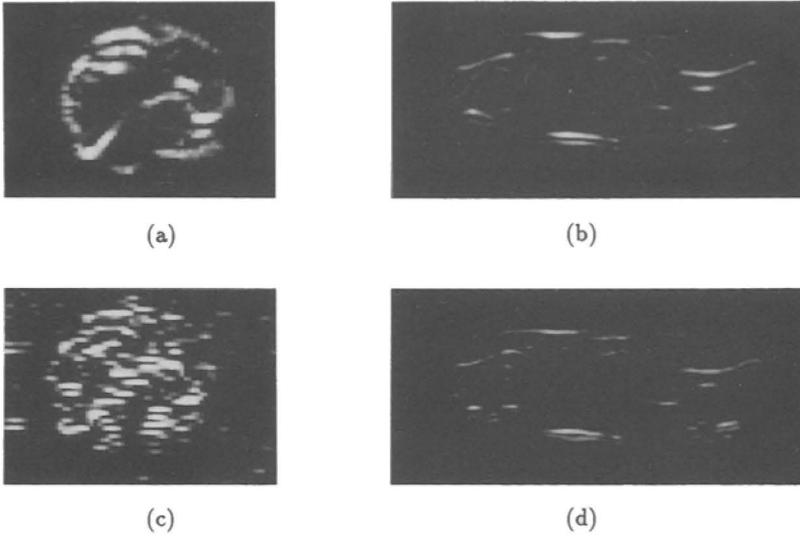
(a)

(b)

(c)

(d)

Figure 9: Our filter outputs: (a) Cone phantom, (b) Baby doll. Sobel filter outputs: (c) Cone phantom, (d) Baby doll.

image	μ_{b1}	μ_{b2}	σ_{b1}	σ_{b2}	σ_{h1}	σ_{h2}	A_{L1}	A_{L2}	A_{W1}	A_{W2}	C_L	C_W
doll	0.70	0.18	0.137	0.118	2	3	3	4	0.4	0.3	0.001	0.001
cone	0.73	0.51	0.137	0.118	2	3	3	4	0.4	0.3	0.001	0.001
liver	0.70	0.40	0.137	0.118	2	3	3	4	0.4	0.3	0.001	0.001

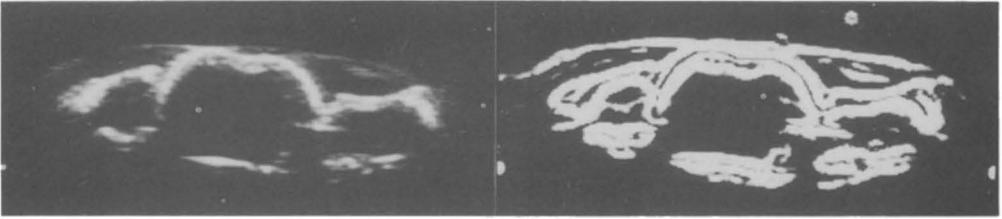**Table 1:** The parameter values used to generate the results in Figure 10.

function. Figure 10b shows that the algorithm detects gaps (pointed to by the white arrows) and tries to close them. The algorithm also detects the gap in the liver image (Figure 10c), but the stable state energy of the snake associated with that gap is so high that it is closed with a very low probability. Thus only a few samples out of the 150 have this gap closed and the averaging result does not show this clearly. This suggests that the parameters in the gap-closing algorithm are not optimal and further fine-tuning is necessary. The values of the parameters used in producing these results are shown in Table 1.
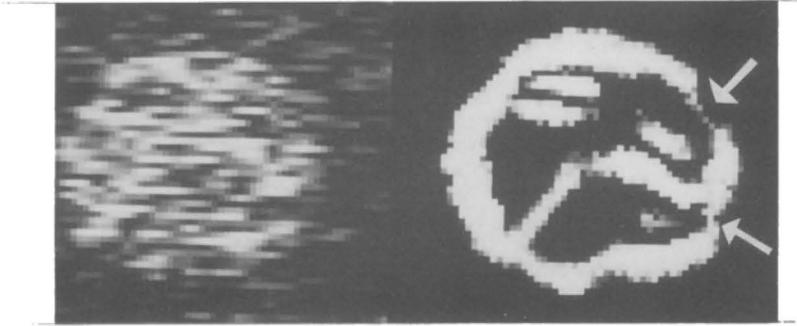
4. DISCUSSION

Due to the large size of our filter, the localization of boundaries is not optimal. Particularly, the filter can not produce correct boundary strength on corners. As a results, small features or corners tend to be missed. In the future reasearch we wish to use a multiscale likelihood function in which filter outputs of different scales are weighted by types of boundary and boundary curvatures. This will allow us to use a smaller filter or filters specially designed to measure corners and should help produce more accurate measurements.

The local shape constraints seem to be the least important factors, compared with the global shape attributes and the likelihood function, in determining the final results. The results from the algorithm without the local constraints do not appear different. The results are sensitive to the estimated mean magnitudes of the filter measurement. An over-estimated μ_{b1} results in low boundary probabilities for weak boundaries and an under-estimated μ_{b2} results in high boundary probabilities for strong speckles.
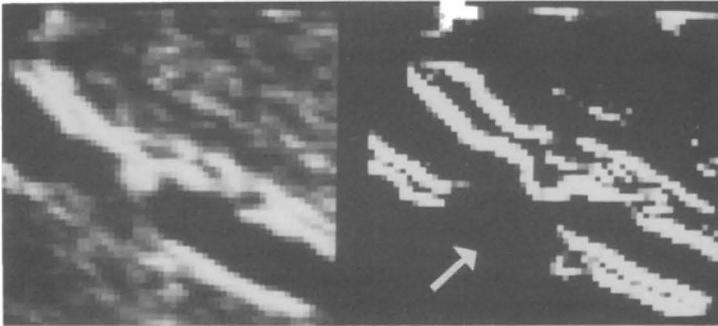
A related problem is that the standard deviations of the filter measurements, σ_{b1} and σ_{b2}, might be fairly large and the means, μ_{b1} and μ_{b2}, might not be well separated. The likelihood function can not resolve ambiguous features and the results tend to have more errors even though the estimations of these parameters are correct. This happens when there are low contrast targets

(a)

(b)

(c)

Figure 10: Test images and the mean boundary values displayed as images: (a) Baby doll, (b) Cone phantom, (c) Liver. The mean boundary values are obtained by averaging over 150 samples.

in the scene.

Current implementation takes about 25 seconds to produce the initial filter measurements at four scales and 4 seconds to produce one sample for a 128 by 128 images on a DEC3100 workstation with unoptimized code. Approximately 50 percent of the running time is spent on sampling, which can be optimized since most pixels are not on boundaries and have very small boundary strength measurement. About 15 percent of the running time is spent on gap-closing procedures and 35 percent on the pyramid algorithm and the rest of the algorithm.

5. CONCLUSIONS

We have described a Bayesian framework for estimating surfaces in noisy images. This framework generates samples of ensemble scenes for an observed image in the form of a random field of binary boundary value and normal directions. It computes posterior means and standard deviations of the ensemble at each pixel and produces a reference image of surface representation which can be further mapped to opacity for volume rendering.

We have applied this approach to two dimensional ultrasound images and have shown its usefulness in producing meaningful boundary representations. This boundary representation not only can be used in visualization but also can be the basis of image segmentation. We believe segmenting via this boundary representation will produce better results than segmenting via the local edge measurements only. We have implemented an active contour model using Gibbs distributions and are ready to test this idea using this model.

The Gibbs prior has also been extended to incorporate important properties of global shapes via multiscale analysis. By incorporating the global attributes in a data augmentation scheme we have extended the use of Gibbs distribution on modeling global structures. Future research includes quantitative evaluation of the algorithm and results and extending this model to three dimensions.

References

Canny J (1986). A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence. PAMI-8(6):679–698.

Dempster AP, Laird NM and Rubin DB (1977). Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society. Ser. B(39):1–38.

Geman S and Geman D (1984). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. IEEE transactions on Pattern Analysis and Machine Intelligence. PAMI-6(6):721–741.

Gelfand AE and Smith AFM (1990). Sampling-based approaches to calculating marginal densities. Journal of the American Statistical Association. 85(410):398–409.

Hykes D, Hedrick WR and Starchman DE (1985). Ultrasound Physics and Instrumentation. Churchhill Livingstone, New York, N.Y., 1985.

Kass M, Witkin A and Terzopoulos D (1988). Snake: Active Contour Models. International Journal of Computer Vision. 1:321–331.

Korn AF (1988). Toward a symbolic representation of intensity changes in images. IEEE Transactions on Pattern Analysis and Machine Intelligence. PAMI-10(5):610–625.

Levoy M (1988). Display of surfaces from volume data. IEEE Computer Graphics and Applications. 8(3):29–37.

Lin WJ (1991). Boundary Estimation in Ultrasound Images. PhD thesis, U. of North Carolina, Chapel Hill, NC, 1991. To appear.

Meer P, Sher CA and Rosenfeld A (1990). The chain pyramid: Hierarchical contour processing. IEEE Transactions on Pattern Analysis and Machine Intelligence. PAMI-12:363–376.

Nassiri DK and Hill CR (1986). The use of angular acoustic scattering measurements to eliminate structural parameters of human and animal tissues. The Journal of the Acoustical Society of America. 79(6):2048–2054.

Shneier M (1981). Two hierarchical linear feature representations: Edge pyramids and edge quadtrees. Computer Graphics and Image Processing. 17:211–224.

Shattuck DP, Weinshenker MD, Smith SW and von Ramm OT (1984). Explososcan: A parallel processing technique for high speed ultrasound imaging with linear phased arrays. Journal of Acoustic Society of America. 75(4):1273–1282.

Tanner MA and Wong WH (1987). The calculation of posterior distributions by data augmentation. Journal of the American Statistical Association. 82(398):528–541.

von Ramm OT, Smith SW, Sheikh KH, and Kisslo J (1988). Real-time, three-dimensional echocardiography. Circulation. 78. Supplement II.

Wagg RC, Dalecki D and Christopher PE (1989). Spectral power determinations of compressibility and density variations in model media and calf liver using ultrasound. The Journal of the Acoustical Society of Americ. 85(1):423–431.