Incremental 3D Ultrasound Imaging from a 2D scanner

Ryutarou Ohbuchi
Henry Fuchs
Department of Computer Science,
Sitterson Hall,
University of North Carolina at Chapel Hill,
Chapel Hill, NC 27599-3175

Abstract

We have been developing an interactive system that will display 3D structures from a series of two-dimensional 2D images acquired incrementally by a conventional 2D ultrasound echographic (2DE) scanner. The user-guided scanner head is mounted on a mechanical arm that tracks the position and orientation of an image frame. We expect in the coming year to complete a version of the system in which the 3D image will be reconstructed incrementally in real-time during the scanning process. This paper reports on the design and implementation of the system, visualization algorithm, and the results of an experiment with a doll phantom to test its feasibility.

1 Introduction

We have been conducting research toward a system that acquires and displays 3D images in real-time, using ultrasound echographics scanning as the imaging modality. In the following we will sketch the 'ultimate' system we are aiming at, then introduce an interim version of it, an *incremental, interactive* 3D echography (3DE) using conventional 2D scanner images integrated over time ans space.

1.1 The 'ultimate' system

The 'ultimate' 3DE system we are aiming at will *acquire*, *display*, and *manipulate* a 3D volume image in real-time. Real-time-ness is very important for an application such as cardiac diagnosis, where static 3DE display, or dynamic display by gated acquisition might miss certain kinetic features. In real-time system, scanhead can be guided by doctor to gain best acoustic window or best view-angle. These can be crucial advantage of system with real-time acquisition and display over non-real-time imaging modalities.

Among the various medical imaging modalities, ultrasound echography is the closest to achieving 3D real-time acquisition. To acquire real-time 3DE image, the 'ultimate' system will use a new 3DE scanner being developed by Dr. Olaf. T. von Ramm's group at Duke University [Shat84]. Due to the velocity of sound limitation (about $1540m/s$ in water), scanning a 3D volume with reasonable resolution ($128 \times 128 \times 128$ or more) in real-time (30 3D-frames/s) requires parallel processing. The new scanner will use a single transmit/multiple-receive scheme called *Explososcan* to increase data acquisition bandwidth. The first implementation will use a 16×16 2D-array transducer along with digital delay-lines implemented in VLSI chips for 3D beam steering and focusing with 64-way multiple simultaneous reception.

Next issue is the visualization of such data. We at UNC-Chapel Hill have been working on the display system based on volume rendering that display 3DE data in real-time. Such a display system has to cope with the challenge of very high data bandwidth. We must also find effective ways to render and manipulate such 3DE images.

The real-time 3DE scanner above will produce on the order of $2-4 \times 10^6$points per frame, or about $60-120 \times 10^6$ points per second. Visualizing a 3D data volume of this bandwidth requires very large computational power, on the order of Giga floating point operations per second, if straightforward algorithm is used. We are approaching this issue through the parallelism, and algorithm efficiency gained by exploitation of various forms of coherences.

Effective visualization method is another major issue. It involves standard problems associated with visualization of 3D data, such as obscuration. To look at the inside of the left ventricle, images of fat tissue, part of myocardium, etc. must be removed to reveal the object behind. This usually requires extensive manipulation of the 3D image data. On top of those are the special character of the ultrasound data, e.g. speckle noise, gain variation, shadowing, specular artifact, etc. that can hinder the 3D imaging.

We expect that *real-time, interactive* image acquisition, manipulation, and display will greatly help over-

TH0311-1/90/0000/0360/$01.00 © 1990 IEEE

come the difficulties in visualizing 3D ultrasound echography. As mentioned above, with real-time acquisition, better acoustic widow can be sought interactively to improve image quality or view angle. Rendering parameters can be changed interactively to suit special objective of imaging. Unconventional view can be gained by changing the viewpoint.

If realized, real-time 3DE scanner and display as sketched above can be a powerful tool for diagnosis, operation planning, intra-operative guidance, etc. If combined with such technology as a real-time optical motion tracker and a head-mounted display, both of which are being developed at UNC-Chapel Hill and at other places, the doctor will be able to view, for example, the 3D image of a fetus or a beating heart overlayed on a patient's chest, with stereopsis and motion parallax.

1.2 Incremental, Interactive 3DE

Visualizing real-time 3DE input is different from rendering off-line data from other modality such as X-ray CT. To study the issues of real-time 3DE imaging before either the real-time 3D scanner or a powerful enough parallel multicomputer, we have been developing an interactive system that will display 3D structures as a series of 2D images incrementally acquired by a conventional 2DE scanner. Currently a user-guided scanhead is mounted on a mechanical arm that tracks the position and orientation of a 2DE. The 3D viewing image is generated as the object is scanned, and the scanning of typical examination volume will take on the order of a few seconds, though the current implementation on SUN4 takes tens of minutes.

Although the basic idea of tracking 2DE image location and orientation for the reconstruction of a 3D volume have been applied for ultrasound many times before [Brin78] [Ghos82] [Hott89] [Lalo89] [Raic86] [McCa88] [Nikr84] [Stic84], our approach is a departure from those; we aim at *incremental acquisition and rendering*, where the 3D image is renderd from the incrementally changing 3D reconstruction result as a new 2DE image slice is acquired and added. This is achieved by a new incremental volume reconstruction and rendering algorithm, as well as the advances in the computing power available.

Although this is intended as a research vehicle for future real-time 3DE, incremental 3DE may prove to be useful medical imaging modality. In the following sections, previous work is reviewed, followed by the description of the design and implementation of our incremental 3DE system, including its incremental reconstruction and rendering algorithms. Result of early experiment of scans of a doll phantom and forearm are reported.

1.3 Previous Work

Acquisition of 3DE data can be done directly by a real-time 3DE scanner, as in [Shat84], or it can be reconstructed from the data of lesser dimension, e.g. 2DE slices. In the latter case, the location and orientation of 2DE slices must be available to reconstruct 3DE data.

Reconstruction of 3D data from a collection of ultrasound 2DE slices have been tried many times. Technical issues in acquisition are; 1) tracking the location and/or orientation of the scanhead, 2) storing multiple image slices for later reconstruction (since none of them are real-time), and 3) synchronizing tracking and images acquisition. Then, there are issues of reconstruction and presenting the data in a usable way, such as estimation of the volume of the heart chamber.

The majority of the prior studies [Brin78] [Ghos82] [Raic86] [Nikr84] [Stic84] had estimation of the volume of the heart as their primary objective. Thus, typical reconstruction is only geometric. The typical process involved manual trace of the 2DE image taken from the video-taped images by human, then digitizing the trace into computer, which resulted in a geometrical reconstruction. In such a study, the display of the reconstruction result is usually a wire frame, or a stack of contours. All the backscatter information is lost in the process or reconstruction.

More recent studies by [Lalo89] [McCa88] actually reconstructed 3D *images*, not only the geometrical information but with the backscatter information. [Lalo89] describes a study on mammograms using a special 2DE scanner which can acquire and store 45 consecutive parallel slice of 1mm interval. Actual data acquisition took approximately 15 minutes, but the reconstruction and rendering is done off-line. Reconstruction used is essentially an image reformatting with cubic spline interpolation. Volume rendering is used for visualization, after reformatting. [McCa88] performed the gated acquisition of a heart's image over a cardiac cycle. It also used video-tape to store 2DE images and used volume rendering to generate images. Upon reconstruction, repetitive low-pass filtering is done on 3D volume image to suppress aliasing artifact by filling spaces between radial slices.

The majority of reported projects used various form of mechanical tracking systems to acquire scanhead location and orientation. Nikravesh et.al used a 6 DOF mechanical arm [Nikr84], Stickels et.al. used 3 DOF arm [Stic84], while Raichelen's group used 1 DOF 'slider'. One interesting variation of 1 DOF tracking is the rotation tracking where a sector scanner is rotated, as in Ghosh et.al. [Ghos82] and later by McCann et.al. [McCa88]. This is done to gain access to the 3DE image of heart through limited acoustic window. The resulting image slices are intersecting planes around rotation axis. This mechanical tracking to achieve 3D image acquisition using 2DE scanner is an extension of the well proven method of achieving compound 2DE by the combination of 1D echographic scanner and a motion tracker.

On the somewhat exotic side, Brinkeley et.al. used an elaborate 6 DOF acoustic tracking system [Brin78], where two spark gaps (sound sources) mounted on the scanhead and carefully arranged microphones tracked the location and orientation of the scanhead. [Mill90] used optical tracking, where images of a 'crown' of tiny lumps were recorded with two TV cameras as the 2DE scanning was done, which yielded 6 DOF tracking information by postprocessing.

2 Methods

The incremental 3DE system visualizes the 3D volume progressively covered by the moving 2D image slice. Location of each 2DE slice acquired is tracked by the mechanical arm motion tracker, as an image slice is digitized from the 2DE scanner. Image reconstruction, instead of geometric reconstruction, is used to preserve image information. The reconstruction and rendering take place for each new 2D image slice, so that the user will see the rendered image build up incrementally. Since scanning of the 3D volume by manually moving 2D scanhead takes a few seconds, the object must be relatively stable to avoid image registration problem. Examples of the possible imaging targets are liver, women's breast and calm fetus in the early month of pregnancy. The design and implementation of this incremental 3DE scanner is presented in this section.

2.1 Incremental 3D scanner system

Figure 1 shows the hardware setup for our incremental 3DE system. The mechanical arm, which came from previous generation 2DE scanner Rohe ROHNAR 5580, has 3 degree of freedom; x, y, and θ. The arm moves in a x-y plane, where the location of the 'wrist' of the arm is given as (x, y), and the angle of the 'hand' around the wrist is given as θ. These values are transduced by potentiometers into voltages, then buffered and sent to the Data Translation DT-1401 A/D converter (12bit resolution) board housed in a SUN-4/280. Sampling rate for the tracking much higher than necessary at around 800samples/s.

Each 2DE image slice acquired by the ultrasound scanner Advanced Technology Laboratory model Ultra Mark 4 is video-digitized at a rate of 30frames/sec by Matrox MVP/S real-time frame grabber. A 3.5MHz linear scanhead is mounted on the arm's wrist, and (x, y, θ) values are sampled for each frame acquired.

The reconstruction process ***incrementally reconstructs*** 3D volume data from 2DE slices, as a new slice is added. Reconstruction result is then passed to rendering stage, which is also incremental as far as the shading parameter and viewpoint are fixed. Both reconstruction and rendering algorithm are described in the following section.

In this set up, if sampled at 2mm interval parallel slices, 100 2DE images cover a volume of 20cm thickness. This will take 3s to acquire, assuming 30frames/s frame rate. Actual acquisition time and the resulting image quality depend on the way the user moves the scanhead; if moved too fast, the anatomical details will be missed.

The current system reconstructs and renders on the SUN-4 workstation. The transfer of 512 × 480pixel, 1byte/pixel frame from video digitizer to SUN-4 memory takes 0.6s. This is before lengthy reconstruction and rendering process. Thus, the system does not currently perform at interactive speed. Currently, the acquired image slices are stored in file along with the arm coordinate values for later rendering and reconstruction experiments.

In the second stage system, which we hope to comelete in the coming year, the video digitizer output will be directly fed, bypassing VME bus and SUN-4, to the Pixel-Planes 5 [Fuch89] through a custom interface board. To allow this, we have chosen the video digitizer board with a special proprietary port that outputs video D/A converter output at the rate of 10MByte/s. With this setup, full speed (30 2D-frames/s) acquisition, reconstruction, and rendering of the image is expected. The algorithm running on the Pixel-Planes 5 will be a parallelized version of the single processor algorithm currently running on the SUN-4.

2.2 Visualization Algorithm

As mentioned above, visualization of incrementally acquired 2DE image slices can take place as an image slice arrives, without waiting for all the slices to arrive. The *reconstruction* is an incremental process, which may be unique to this system. For rendering, we have developed a volume rendering algorithm based on ray-casting into a regular grid, as found in [Levo88], as a basis.

Fig 2 shows the pipeline for the incremental volume visualization. First, 2DE image slices are incrementally *reconstructed* into a 3D scalar field sampled at 3D regular grid points. It is stored in a 3D *reconstruction buffer* in object space. From this point on, the process is a analogous to standard volume rendering pipeline, except that this is an incremental algorithm. The 3D scalar field is *classified* (non-binary classification) and *shaded*, yielding a 3D *shade buffer*, also in

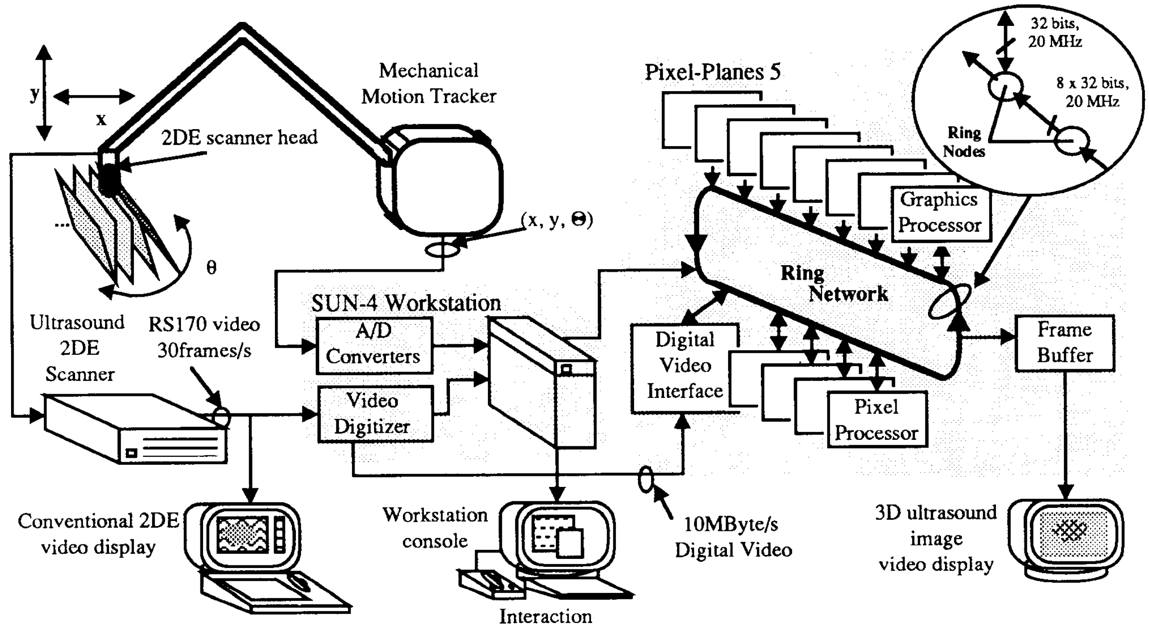

Figure 1: Overall structure of the incremental, interactive 3D ultrasound scanner system.

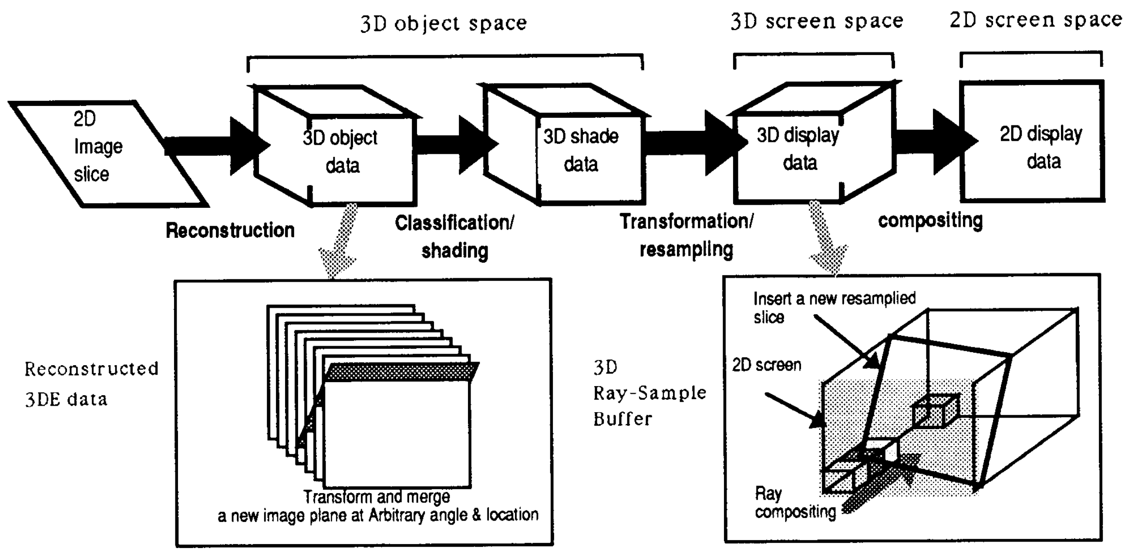

Figure 2: Incremental reconstruction and volume rendering pipeline.

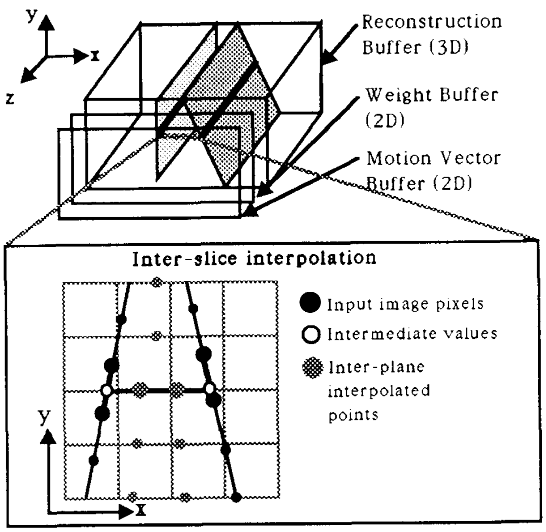

Figure 3: Inter-slice interpolation reconstruction

object space. Next step is the ray-casting, which essentially *transforms* object space data into 3D screen space with proper *resampling*. Filtered resampling is necessary to avoid aliasing in the final image. Finally, sampled values in 3D screen space are *composed*, to yield 2D viewing image.

2.2.1 Incremental Reconstruction

In addition to the geometrical constraint of 3 degree-of-freedom (x, y, θ) given by the arm, we made two more assumptions for reconstruction; *1) the scalar field we are sampling (e.g. human body) is a continuous function of the space, 2) locations of 2D slices in sequence have continuity*. First assumption means that the final image must not look like a set of intersecting planes; the space between two slices must be filled with reasonable values. Second assumption means that the consecutive slices are geometrically next to each other. Also, it means that the temporal sampling rate of 2D image acquisition, hence spatial sampling rate of a sequence of slices are high enough to sample changes in the 3D scalar field (e.g. human body) we are imaging.

Basic method of interpolated reconstruction is forward mapping of input sample values into the reconstruction buffer, using a certain low-pass filter kernel (See Figure 3). An input pixel value is distributed among the neighboring grid points weighted by the filter coefficients. We have chosen a triangular function for the filter kernel. The filter kernel is made asymetric, since the resolutions of the ultrasound scanner are different from axis to axis. The filter kernel rotates along with the image slice, and its sizes are specifiable by the user. The coefficients of a filter kernel are also accumulated, this time in a 2D *weight buffer* for normalization of the value as they are used in shading. A 2D, instead of 3D weight buffer is enough, since there is only one axis of rotation.

This is sufficient for parallel slices of predetermined interval; kernel size can be matched to the interval. But for a manually guided scan widely spaced. Theoretically, a filter kernel with infinite fall-off can be used to reconstruct such an image. But this is not practical, especially for its computational cost. We chose to adaptively *fill-in* spaces between slices. The current algorithm bi-linearly interpolates between current and previous slices. First linear interpolation is along the slice to obtain values at every integral y coordinate on the slice, and the second linear interpolation is along the X-axis to fill space between slices. For the latter, distance along X-axis is split into equal intervals, and at every interval, values interpolated along X-axis is splat using filter kernel. This *adaptive inter-slice interpolation* keeps kernel size, and thus computational cost down to a realistic value.

There is another issue associated with incremental scanning. A volume can be scanned many times over, and older values from older slices should somehow be replaced by new values. One idea is the *decay method*, which lets images decay gradually over time, as images on a classical radar screen with long persistence phosphor. This can be implemented by keeping a 2D map of time-stamps for every Z-lines at (x, y). Another is *motion vector method*, which somehow detect if the scanhead is scanning forward or backward (relative to last sweep past that voxel) in the volume. If it is scanning forward, the values and weights are simply accumulated. If it is scanning backward, new values and weights start replacing older values and weights. The current implementation use the latter, motion vector method, by keeping the 2D motion vector for each pixel (actually a Z-line) in the 2D *motion vector buffer*. If previous motion vector of a pixel is more than 180^o different from the motion vector of the current sweep, it is determined as scanning backward, and the values in the reconstruction buffer as well as weights in the weight buffer are replaced. The replacement is actually a weighted average of new and old values.

2.2.2 Incremental Volume Rendering

The incremental volume rendering algorithm tries to save computation by taking advantage of three assumptions; 1) incremental acquisition of 2DE slices, 2) shading parameter will not change for every few frames, 3) viewpoint will not change every few frames. If these conditions are met, an incremental rendering can be done at real-time rate in the second generation system. By incrementally shading and ray-sampling per inserted 2D slice, computation is proportional to the area of the inserted image, instead of the reconstruction volume.

After reconstruction, classification is done by table lookup, and the shading values, color and opacity, are computed from the 3D scalar field in the reconstruction buffer. This happens incrementally, only in a subset of volume between current and previous image slices inserted. Currently, two shading algorithms are implemented; 1) *image value shading* which directly maps from input scalar value to the color value by table lookup, and 2) *gradient-Phong shading* which performs Phong shading with diffuse and specular components, where surface normal is approximated by finite difference. The shading result is stored in the *shade buffer*.

The ray-sampling stage performs actual ray-casting, where a ray is cast from a pixel into 3D shade buffer, and the shading values are sampled at uniform interval along the ray. We used perspective projection to give better 3D perception. Ray sample is the result of tri-linear interpolation to minimize aliasing in the resulting images. This stage also works incrementally, and only the same sub-volume processed in shading stage is sampled. Ray sample values are saved in a 3D array called *'Ray-sample cache'* in 3D screen space. Since the ray-sampling is a costly process, saving and resuing ray-sample values saves a lot of computation.

In general, the composition stage has to be done on entire ray-samples in 3D screen space even if only a 2D slice of resampled ray values are inserted into ray-sample cache, and can not be done incrementally. The current implementation has multiplicative composition as in [Levo88], as well as additive composition, and Maximum Intensity Projection, which takes the maximum sample value along the ray as a pixel value.

If the viewpoint is changed, ray-sampling and compositing have to be done essentially on the entire data. Or, if the shading parameters are changed, the shading have to be done in addition to ray-sampling and compositing. In such cases, various forms of coherence, such as image coherence and object coherence as well as temporal coherence can be used to reduce requirements on computation. Current implementation is a proof of a concept, and includes none of these optimizations.

In the second stage system using Pixel-Plane 5, reconstruction, shading, and ray-sampling will take place in the 16 to 32 Graphics Processors. The last stage, the compositing, will take place in the Pixel Processors. The Pixel Processors' memory is used as ray-cache. With its parallelism, Pixel Processors will be able to compose cached ray-samples in 1/30s. For non-incremental cases, the second stage system is estimated to take a second to generate image from viewpoint change, and a few second to generate image from shading parameter change.

Figure 4: Experimental setup. The 2DE scanner, whose linear scanhead mounted on the mechanical tracking arm, and a phantom in the watertank is seen.

3 Experiment

An in-vitro experiment is done to test visualization algorithm, as well as the image acquisition system setup. The 2DE scanner used is the Advanced Technology Lab. Mark-4, with 3.5MHz linear scanhead. Figure 4 shows the overall setup, including the watertank with a plastic baby doll as a phantom.

We built another phantom for geometric calibration of the system. The scale (i.e. mm/pixel) of the image and the location of the image relative to the arm must be known for reconstruction. The design of the phantom is suggested by Mills, and is analogous to that of the one found in Joynt's [Joyn82], though they are developed independently.

For the phontom test of the scanner system, we chose a plastic doll of a baby, with height of 16.5cm, which roughly corresponds to the fetus in about 18weeks of pregnancy. Figure 5 shows the doll, as well as two slices out of 2DE images acquired. We suspended the doll by a fine wire, which is visible in the head slice image. We scanned the doll at approximately parallel, constant interval (2mm, 3mm, and 5mm) slices, as well as arbitrary location and angle slices. In the constant interval scan, the program tracked the x coordinate, and digitized the image after a given translation limit (e.g. 2mm) was surpassed. Images and coordinates are stored in the file for later reconstruction and rendering studies. We also scanned a forearm of a volunteer at

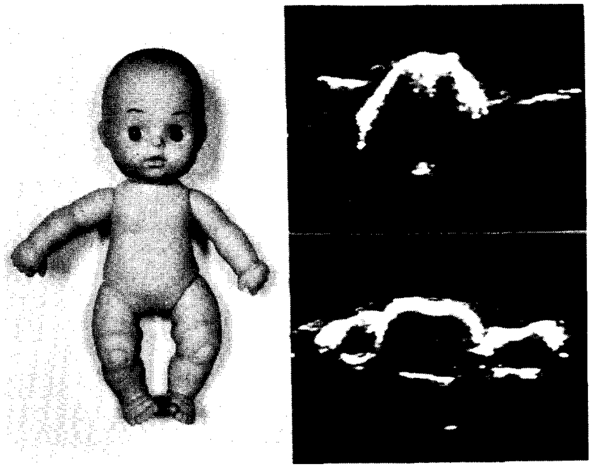

Figure 5: Left : Picture of the doll scanned; Right : Images of the torso and head slices.

3mm slice, for an example of real tissue scan.

4 Result

Figure 6 shows the baby doll image reconstructed from 90 approximately parallel slices of 2mm interval. Each 2DE image acquired was of 256×405 pixels, which corresponds to $98 \times 195mm$ in real size. This is windowed to 245×146 pixels, and horizontal axis is sub-sampled by 2 to 1. Thus, 2DE image slice size used for this reconstruction is 122×146. This is mainly to reduce the necessary memory for reconstruction buffer. For each slice inserted in 2mm interval, average of 3 additional points are inserted for inter-slice interpolation, to fill the space.

In the reconstructed image, the diagonal line on the upper-right is the image of the string. The almost horizontal 'cloud' around the feet is a scanning artifact, energy from the previous transmission from the neighboring transducer element bounced off the bottom of the watertank. Artifact visible as horizontal stripe pattern is likely to have resulted from tracking, acquisition, and reconstruction error. Various features such as eyes and nose (feature size of about 5mm) are recognizable, especially if seen in a rotating movie loop. Notice also that at the right shoulder of the doll, both the front and back interface of the 'skin' can be seen.

Figure 7 shows a volunteer's right arm, scanned in thumb up position in the watertank with the same setup, reconstructed and rendered. Left is the anterior view, and the right is the posterior view. It is about 18cm in actual length. In the reconstructed image, some muscle/fat interfaces and muscle/bone interfaces are visible. With a movie loop, more subtle and complex 3D structures become apparent.

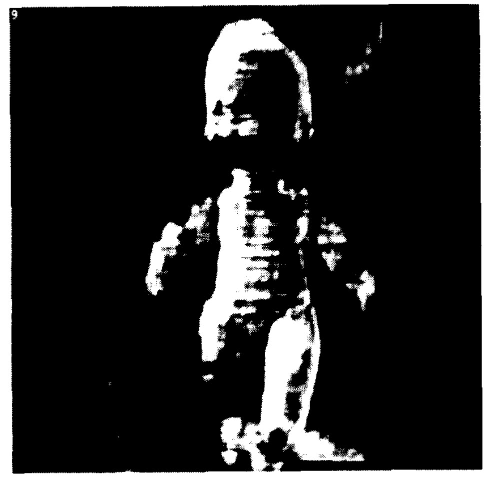

Figure 6: Baby doll phantom visualized from 90 approximately parallel slices of 2mm spacing.

5 Conclusion

We have established a basis for the incremental, interactive 3DE, which visualizes a 3DE image from consecutive 2DE image slices acquired in real-time. Though currently not interactive, the our next generation system running on Pixel-Planes 5 should be able to incrementally reconstruct and render a 3DE image at interactive rates.

The mechanical tracking arm and the real-time image digitzer allowed us to acquire multiple image slices relatively quickly and easily. digitizing process as post-processing. The incremental reconstruction algorithm with 3D forward mapping and adaptive linear interpolation performed pretty well. As seen in the reconstructed image of the doll phantom, it can reconstruct and present small features in the rendered image. We have also tried the reconstruction and rendering of the human forearm, by the same method. A higher order inter-slice interpolation method, such as cubic spline interpolation, can increase image quality at the expense of speed.

We plan to improve this system in the following ways. For reconstruction, higher order inter-slice interpolation functions as well as replacement strategy for aging data have to be studied. Rendering stage currently has virtually no optimization implemented. Algorithm developed for optimized parallel rendering algorithm for real-time volume rendering will be implemented in the incremental, interactive 3DE system.

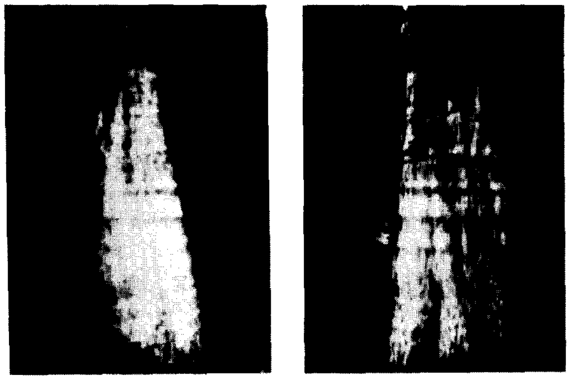

Figure 7: Volunteer's forearm visualized from 60 approximately parallel slices of 3mm spacing.

Acknowledgement

The authors would like to thank Vern Katz, M.D. for the use of the ultrasound scanner, and Jeff Butterworth for assisting with the tracking arm and video digitizer interface. We also would like to thank the Wake Radiology Association for donating ROHNAR 5580 to us. This research is supported by NSF grant number CDR-86-22201.

References

[Brin78] Brinkley, J.F., Moritz, W.E., and Baker, D.W., "Ultrasonic Three-Dimensional Imaging and Volume From A Series Of Arbitrary Sector Scans", Ultrasound in Med. & Biol., Vol.4, pp317-327 (1978).

[Clea88] Cleary, J.G., Wyvill, G., "Analysis of an algorithm for fast ray tracing using uniform space subdivision", Visual Computer, Vol.4, pp65-83, (1988).

[Fuch89] Fuchs, H., Poulton, J., Eyles, J., Greer, T., Goldfeather, J., Ellsworth, D., Molner, S., Turk, G., and Israel, L., "Pixel Planes 5: A Heterogeneous Multiprocessor Graphics System Using Processor-Enhanced Memories", Computer Graphics, Vol.23, No.3, pp79-88 (1989).

[Ghos82] Ghosh, A., Nanda, C.N., and Maurer, G., "Three-dimensional Reconstruction of Echo-cardiographic Images Using The Rotation Method", Ultrasound in Med. & Biol., Vol.8, No.6, pp655-661 (1982).

[Hott89] Hottier, F., Paris Phillips Research Lab. (Private communications)

[Joyn82] Joynt, L. and Popp, R.L., "The Concept of Three Dimensional Resolution in Echocardiographic Imaging", Ultrasound in Med.& Biol. Vol.5, No.3, pp237-247 (1982).

[Lalo89] Lalouche, R.C., Bickmore, D., Tessler, F., Mankovich, H.K., and Kangaraloo, H., "Three-dimensional reconstruction of ultrasound images", Proc.SPIE'89, Medical Imaging, pp.59-66 (1989).

[Levo88] Levoy, M., "Display of Surfaces from Volume Data", IEEE CG&A, Vol.8, No.5, pp29-37, (May,1988).

[McCa88] McCann, H.A., Sharp, J.S., Kinter, T.M., McEwan, C.N., Barillot, C., and Greenleaf, J.F., "Multidimensional Ultrasonic Imaging for Cardiology" Proc.IEEE, Vol.76, No.9, (Sep.1988).

[Mill90] Mills, P.H., Fuchs, H, "3D Ultrasound Display using Optical Tracking", Proc. 1'st Conference on Visualization for Biomedical Computing, (in these proceedings), Atlanta, GA, May.1990.

[Nikr84] Nikravesh, P.E., Skorton, D.J., Chandran, K.B., Attarwala, Y.M., Pandian, N., and Kerber, P.E., "Computerized Three-Dimensional Finite Element Reconstruction of the Left Ventricle from Cross-sectional Echocardiograms", Ultrasonic Imaging, Vol6, pp48-59 (1984).

[Raic86] Raichelen, J.S., Trivedi, S.S., Herman, G.T., Sutton, M.G., and Reichek, N. "Dynamic Three Dimensional Reconstruction of the Left Ventricle From Two-Dimensional Echocardiograms", Journal. Amer.Coll.of Cardiology, Vol.8, No.2, pp364-370 (Aug.1986).

[Sabe88] Sabella,P., "A Rendering Algorithm for Visualizing 3D Scalar Fields", Computer Graphics, Vol.22, No.4, pp51-58, (1988).

[Shat84] Shattuck,D.P., Weishenker, M.D., Smith, S.W., and von Ramm, O.T., "Explososcan: A Parallel Processing Technique for High Speed Ultrasound Imaging with Linear Phased Arrays", pp1273-1282, JASA, Vol.75, No.4, (1984).

[Stic84] Stickels,K.R. and Wann,L.S., "An Analysis Of Three-Dimensional Reconstructive Echocardiography", Ultrasound in Med. & Biol., Vol.10, No.5, pp575-580, (1984).

[Upso88] Upson,C. and Keeler,M., "VBUFFER: Visible Volume Rendering", Computer Graphics, Vol.22, No.4, p59-64, (1988).