

---

**STRADX: REAL-TIME ACQUISITION  
AND VISUALISATION OF  
FREEHAND 3D ULTRASOUND**

R. W. Prager, A. H. Gee and L. Berman

**CUED/F-INFENG/TR 319**

April 1998

University of Cambridge  
Department of Engineering  
Trumpington Street  
Cambridge CB2 1PZ  
England

Email: [ahg/rwp@eng.cam.ac.uk](mailto:ahg/rwp@eng.cam.ac.uk), [lb@radiol.cam.ac.uk](mailto:lb@radiol.cam.ac.uk)

---

# Stradx: Real-time Acquisition and Visualisation of Freehand 3D Ultrasound

Richard Prager, Andrew Gee and Laurence Berman\*  
University of Cambridge

Department of Engineering  
Trumpington Street  
Cambridge CB2 1PZ

\*Department of Radiology  
Addenbrooke's Hospital  
Cambridge CB2 2QQ

## Abstract

Conventional freehand 3D ultrasound is a multi-stage process. First, the clinician scans the area of interest. Next, the ultrasound data is used to construct a 3D voxel array, which can then be visualised by, for example, any-plane slicing. The strict separation of data acquisition and visualisation disturbs the interactive nature of the ultrasound examination. Furthermore, some systems require the clinician to wait for an unacceptable amount of time while the voxel array is constructed. In this paper, we describe a novel freehand 3D ultrasound system which allows exceptionally accurate acquisition of the raw data and immediate visualisation of arbitrary slices through the data. Minimal processing separates the acquisition and visualisation processes, with the result that fine detail present in the B-scans is preserved in the synthesised slices.

## 1 Introduction

Conventional diagnostic ultrasound imaging is performed with a hand-held probe which transmits ultrasound pulses into the body and receives the echoes. The magnitude and timing of the echoes are used to create a 2D grey scale image (B-scan) of a cross-section of the body in the scan plane.

Using a technique called **freehand 3D ultrasound imaging** [2, 14, 18, etc.], it is possible to construct 3D data sets from a series of 2D B-scans — see Figure 1. A 3D freehand examination can be broken into three stages: scanning, reconstruction and visualisation. Before scanning, some sort of position sensor is attached to the probe. This is typically the receiver of an electromagnetic position sensor [1, 3, 5, 8, 10], as illustrated in Figure 1, although alternatives include acoustic spark gaps [6], mechanical arms [11] and optical sensors [17, 19]. Measurements from the position sensor are used to determine the positions and orientations of the B-scans with respect to a fixed datum, usually the transmitter of the electromagnetic position sensor. In the next stage, the set of acquired B-scans and their relative positions are used to fill a regular voxel array. Finally, this voxel array is visualised using, for example, any-plane slicing, volume rendering or surface rendering (after segmentation).

Freehand systems can be used to obtain arbitrary volumes of data, since the motion of the probe is unconstrained. They are also cheap, requiring only existing, conventional ultrasound systems and relatively inexpensive additional components. For these reasons, research into freehand systems is very active (this paper's bibliography provides a representative survey), and several commercial systems have recently become available.

There are, however, several problems with the conventional, voxel-based approach. While it is straightforward to construct a voxel array from regularly sampled data, such as the parallel slices produced by CT and MRI scanners, the same cannot be said for freehand 3D

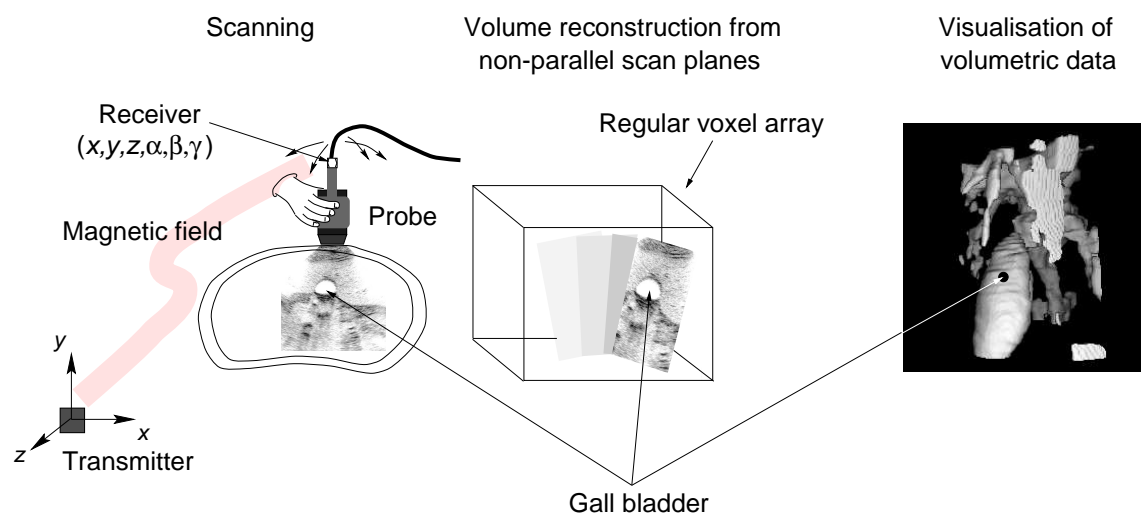


Figure 1: **Conventional freehand 3D ultrasound imaging.** This is a multi-stage process, involving scanning, reconstruction and visualisation. The figure illustrates an examination of a gall bladder.

ultrasound. Figure 2 illustrates some of the problems. Depending on the scanning pattern, voxels may be empty or intersected by one or more B-scans. Since empty voxels create artifacts in renderings and confuse automatic segmentation algorithms, they are usually filled-in using some sort of interpolation scheme. The better schemes [11] take into account the finite width of the ultrasound beam — see Figure 3. Voxels intersected by multiple B-scans are often set using a simple average of the contributions from each B-scan: this has the beneficial effect of suppressing speckle noise [1, 4, 5, 9, 10, 15]. Alternatively, a maximum intensity approach can compensate for drop-out and shadowing in the B-scans [9].

Whatever approach is used to construct the voxel array, it is clear that there are many parameters to set. What size should the voxels be? What should be done with those voxels intersected by multiple B-scans? What should be the size and shape of the interpolation kernel? Another problem is that a lot of filtering takes place during the construction of the voxel array: it is often said that slices through 3D ultrasound data sets appear processed beyond recognition. The final problem is the time it takes to construct the voxel array: the better interpolation schemes (which make use of steerable kernels) are computationally expensive. There may be a delay of several minutes after scanning before the clinician is able to visualise the data.

In this paper, we describe the Stradx<sup>1</sup> freehand 3D ultrasound system, which aims to avoid the pitfalls of a voxel-based approach. The Stradx system deals exclusively with raw 3D ultrasound data, just B-scan images and positions. At no stage is a voxel array constructed. Great care is taken to label each B-scan with the most accurate position and orientation possible, as

<sup>1</sup>The name “Stradx” derives from an X-windows implementation of an earlier system developed under the UK Engineering and Physical Sciences Research Council’s “Stradivarius” project. The Stradx software, along with full documentation, can be downloaded from <http://svr-www.eng.cam.ac.uk/~rwp/stradx/>.

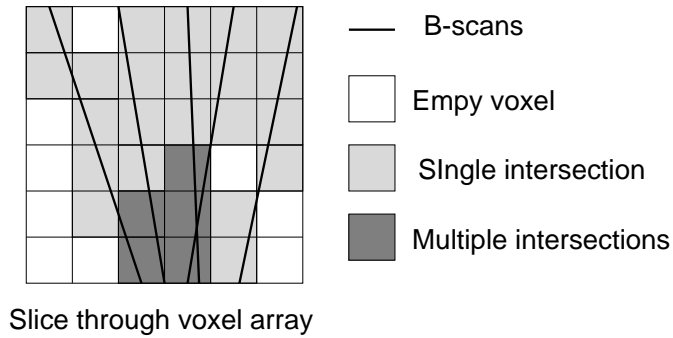


Figure 2: **Constructing voxel arrays from freehand 3D ultrasound data.** Depending on the scanning pattern, voxels may be empty or intersected by any number of B-scans.

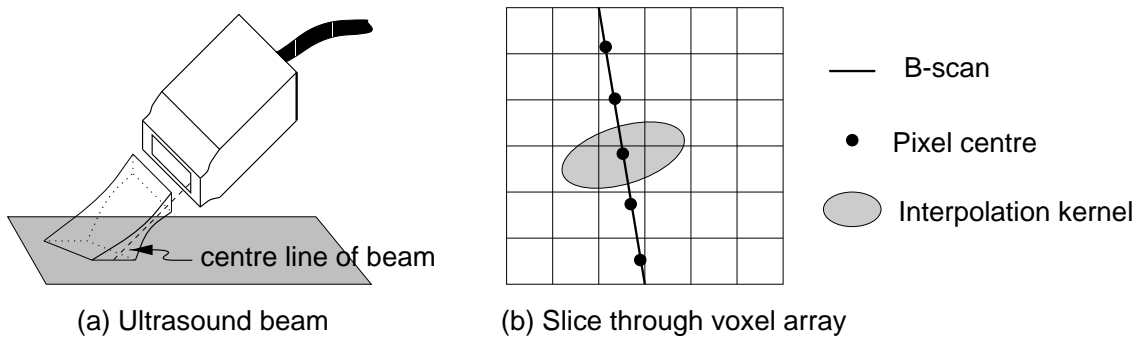


Figure 3: **Interpolation for freehand 3D ultrasound.** The ultrasound beam (a) has a finite thickness, so each pixel in the B-scan samples a region of space which extends some distance normal to the B-scan. The interpolation kernel (b) should resemble this sampling function. Each B-scan pixel influences the contents of voxels lying within its interpolation kernel.

described in Section 2. In Section 3, we describe how visualisation by panoramic imaging or any-plane slicing is possible immediately following, or even during, the examination, without the need to set any parameters. Finally, we draw some conclusions in Section 4.

## 2 Acquiring 3D ultrasound data

At the heart of the Stradx system lie two asynchronous processes. One process acquires B-scan images from the ultrasound machine as fast as possible (every 40ms for a PAL video stream) and places them in a circular buffer in the main memory of the computer. The other performs a similar service with the position sensor readings. These are acquired every 30ms or so, and placed in another circular buffer in the computer's main memory.

Since Stradx does not require a voxel representation of the data, it remains only to match the images with the positions. A simple approach might be to label the most recent image with the most recent position reading. However, greater accuracy can be achieved by interpolating the position between the readings just before and just after each image is acquired. In order to produce the real-time tools in the Stradx system, we would like to perform this temporal matching and interpolation on the fly, as the images and positions are coming in.

### 2.1 Matching images and positions

To match images and positions, we need to know the time that each image is grabbed, and the time each position reading is taken, as accurately as possible. The hardware we use (Silicon Graphics Indy and O2 workstations) provides a nanosecond time-stamp on each image acquired by the frame-grabbing card. We can record a similar time-stamp when we send a character to the position sensor to request a reading. We have found that there is an approximately constant time offset between the point just after sending the character, and the point when the sensor actually takes its reading. This enables us to time-stamp the position readings as well. The constant time offset is determined using an automatic calibration procedure described in Section 2.2.

Now it just remains to linearly interpolate between the relevant pair of readings to find the position and orientation of each B-scan. This is where we exploit the two circular buffers. Instead of picking the most recent image out of the circular buffer, we select the most recent image that lies within a specific age range — see Figure 4. In this way, there is a good chance that a position reading will be available on either side of it. The age range is calculated to span about one and a half times the image sampling period, so there will always be an eligible image available.

It is hard to predict the correct age offset of the image sampling window, since it depends on the latency of the position sensor. This parameter is therefore determined automatically using an adaptive algorithm. The system maintains a record of the time offset between the current image and the reading at the centre of the position circular buffer, averaged over the last 100 images. If more than five out of the last 100 images are unable to be matched with position readings, then this record is used to correct the age offset of the image sampling window.

Given a suitable age offset, the most recent image in the image sampling window is matched to one of the four possible intervals in the position circular buffer using binary search. Once the precise position and orientation of the B-scan has been calculated by linear interpolation, the raw position data stream is discarded and the B-scan image, along with its

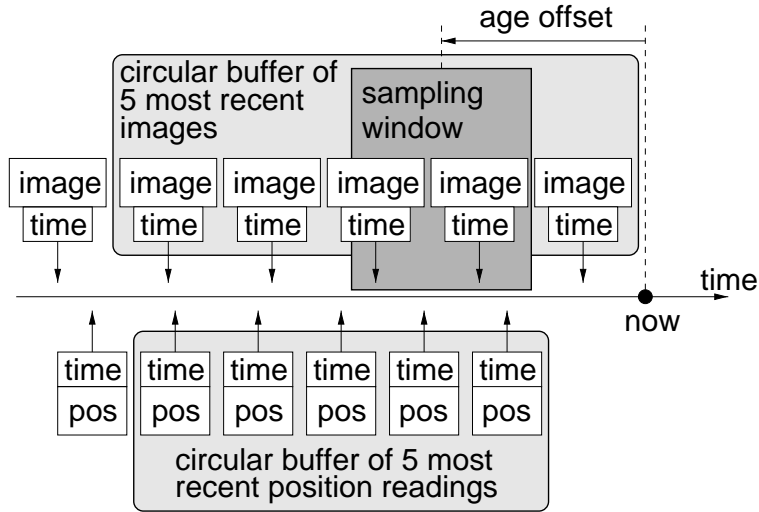


Figure 4: **Matching images and positions.** The five most recent images and positions are stored in circular buffers (for clarity, the circular buffers have been drawn unwrapped along a straight time line). By selecting images which lie within a certain age range, it is possible to find their positions accurately by linear interpolation between two position readings.

position and orientation, is provided for use by the rest of the Stradx system. Care must be taken when interpolating between two orientation angles, since they wrap around at  $\pm 180^\circ$ .

## 2.2 Temporal and spatial calibration

There are two calibration processes which are central to the accuracy of the Stradx system. The first of these involves estimating the latencies in the position sensing device and the ultrasound machine. As explained in Section 2.1, images are automatically time-stamped by the frame-grabbing card. However, we do not know whether there are significant latencies in the ultrasound machine and the frame-grabbing hardware, which might cause the image to be time-stamped some time after it is actually acquired by the ultrasound probe. Similarly, there is an unknown latency in the position sensor readings: we take a time-stamp whenever we request a reading, though the delivered reading generally comes from a buffer inside the position sensing device and might already be fairly old. The temporal calibration procedure involves estimating the difference between the image and position latencies. This difference is then added to the position sensor time-stamps, so that the image and position time-stamps can be meaningfully compared.

The temporal calibration is performed by applying a step input simultaneously to both the image and position streams. In practice, this is achieved by holding the probe still against the skin and then suddenly jerking it away, producing step changes in both the image content and position readings. Any discrepancy in the timings of the observed disturbances can be used to adjust the offset added to the position sensor time-stamps.

The motion of the probe is monitored by looking at the amount it has translated between successive position readings (the orientation angles are ignored). The calibration software asks the user to hold the probe steady for a period of 2.5 seconds, where “steady” means that

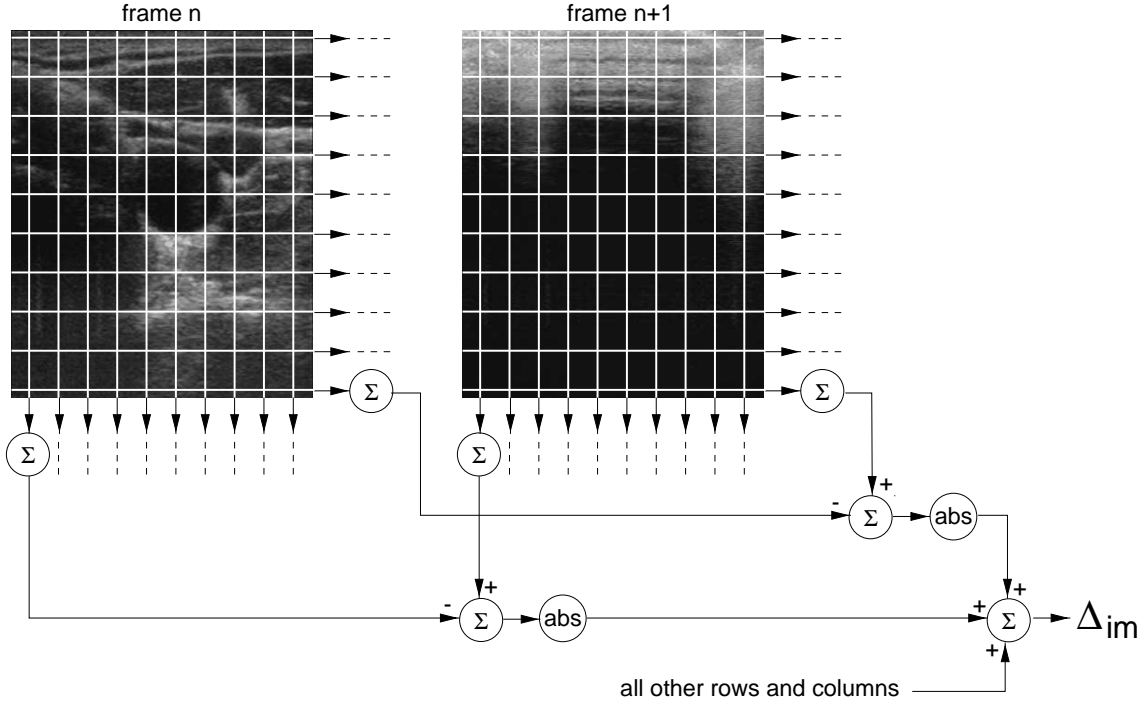


Figure 5: **Detecting sudden changes in the image.** The difference measure  $\Delta_{im}$  is robust to speckle noise, quick to compute and sensitive to changes in all parts of the image, as long as the image contains some sort of large scale structure.

the probe should not move by more than 1mm between successive position readings. When the user has steadied the probe for the required time, the system primes itself, asking the user to jerk the probe suddenly. The subsequent motion is accepted if the probe moves by more than 5cm in 100ms. The system records the time of the step, which is deemed to take place mid-way between the two position readings which first differed by more than 1mm.

While this is happening, the system is also examining the incoming image stream. It compares successive images and records a measure of their difference in a circular buffer. The difference measure must be robust to speckle noise, quick to compute and sensitive to changes in all parts of the image. To this end, Stradx sums the pixel intensities along ten evenly spaced rows and columns spanning the image. These sums are subtracted from their values in the previous frame, and the absolute values are accumulated to give the difference measure  $\Delta_{im}$  — see Figure 5. Sudden changes in the image produce significant jumps in  $\Delta_{im}$ .

The time of the step is estimated from the image stream as follows. Immediately following the jerk, the system looks in the circular buffer of difference measures and picks out those which were recorded while the probe was being held steady. From these, the system notes the maximum steady-state value of  $\Delta_{im}$ . Care must be taken at this stage, since the image time-stamps cannot be reliably compared with the position time-stamps: there is an unknown latency which we are currently trying to estimate. So when deciding which images were acquired with a steady probe, the system allows for some predetermined maximum latency, which Stradx generously considers to be 0.5s. After the calibration process is primed (again, allowing for up to 0.5s latency), the system looks for the first  $\Delta_{im}$  that exceeds the maximum

steady-state value by 10%. The step is deemed to take place mid-way between the two images that produced this increased  $\Delta_{\text{im}}$ .

The constant offset added to the position sensor time-stamps can now be adjusted by looking at the delay separating the image and position disturbances. Note, however, that if images are acquired every  $t$  seconds, and positions every  $T$  seconds, then the offset can be estimated only to an accuracy of  $\pm(T + t)/2$  seconds. For this reason, it is advisable to perform several temporal calibrations to establish the range of results, and then accept a value that lies at the centre of this range.

The second calibration process is a spatial one. The position sensor indicates the position of the small receiver, mounted on the probe, with respect to the fixed transmitter. What we really want to know is the position of each B-scan pixel with respect to the transmitter. The calibration process involves deducing the six degree-of-freedom rigid body transformation between the position sensor receiver and the corner of the B-scan plane, and also the  $x$  and  $y$  dimensions of the pixels (in mm/pixel). This is a fairly complex process which is usually performed by scanning and reconstructing some known object, and then using the discrepancy between the reconstructed shape and the known shape to re-estimate the calibration parameters [1, 3, 7, 8, 17]. Stradx uses a novel, state-of-the-art spatial calibration technique which we describe in full elsewhere [12, 13].

### 3 Visualising 3D ultrasound data

Stradx provides two visualisation tools, both of which are available immediately after the data has been acquired, without having to wait for a voxel array to be constructed. In fact, both visualisation tools can be used even at the same time as the data is being acquired. The first tool, any-plane slicing, displays the ultrasound data on any plane selected by the user through the 3D data set. The second tool, panoramic imaging, allows many B-scans to be stitched together to form a seamless, wide-angle composite.

#### 3.1 Any-plane slicing

Any-plane slicing of regular voxel arrays is straightforward and fast. The slice plane is selected interactively, and then the voxels which straddle the slice plane are rapidly picked out of the array (using integer arithmetic) and sent to the display. Any-plane slicing of scattered B-scan data is not so straightforward. A naive reslicing algorithm might extract intensity values along the lines of intersection of each B-scan with the slice plane, and paint these intensities onto the slice plane — see Figure 6. The problem with this simple approach is that the resulting slice comprises a set of fragmented line segments, as shown in Figure 6(c).

To improve the quality of the slice we clearly need to interpolate between the line segments. Figure 7 illustrates an interpolation scheme which takes into account the anisotropic resolution of the 3D ultrasound data. Since the ultrasound beam has a finite thickness, perhaps as much as 10 mm [16], we should think not of a single, thin B-scan in space, but a continuum of repeated B-scans smeared across the width of the ultrasound beam<sup>2</sup>. The intersection of the slice plane with the smeared B-scan is now a polygon. Intensity values can be extracted within this polygon and drawn at the appropriate position on the slice plane. Repeating

---

<sup>2</sup>For simplicity, we assume that the width of the ultrasound beam is constant, though in reality the width varies with distance from the probe face [16] — see Figure 3(a).



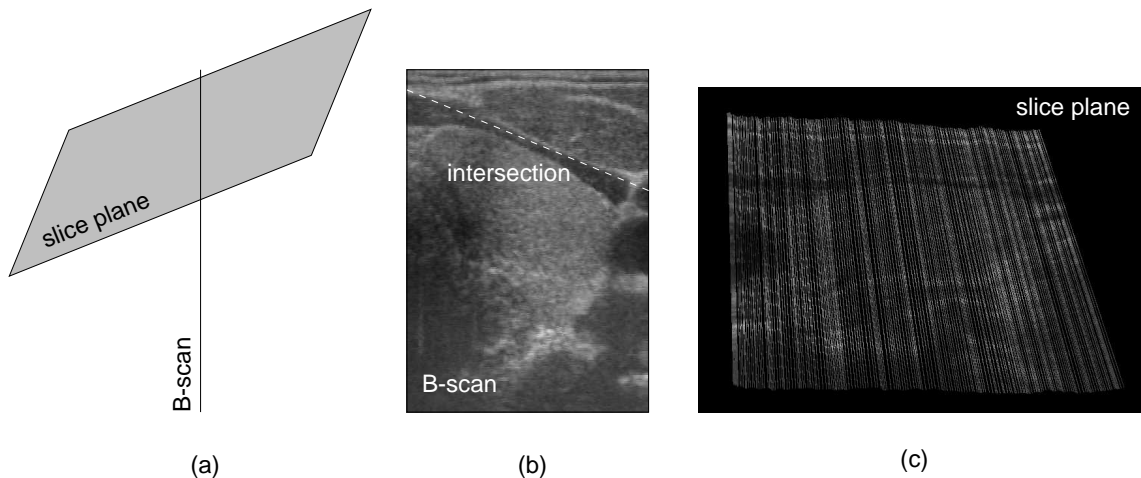


Figure 6: **Reslicing thin B-scans.** The slice plane intersects each B-scan along a line, as shown in (a) and (b). Intensities can be extracted from the B-scans along these lines and drawn on the slice plane. However, this produces an unsatisfactory, fragmented image, as shown in (c).

this process for all B-scans tiles the slice plane with a set of overlapping polygons filled with intensity data.

It is only left to decide which intensity to display where two or more polygons overlap on the slice plane. Intuitively, we would like to see data gathered along the centre line of the ultrasound beam, and not the periphery — see Figure 3(a). This can be achieved by mapping the intensity data not onto a flat polygon in the slice plane, but a wedge, such that the orthographic projection of the wedge onto the slice plane is the same as the original polygon — see Figure 8. To an observer viewing the slice plane from above, a single wedge looks exactly the same as the flat polygon. However, intensities associated with the centre of the ultrasound beam are rendered close to the viewer, while those from the extremities of the beam are further away. If the graphics system is instructed to render the wedges using orthographic projection *with hidden surface removal*, then the slice will reveal only those portions of the polygons that are as close as possible to their centre lines.

Note that the voxel-less reslicing technique is effectively free of parameters. We say “effectively”, since it is necessary to specify the beam thickness. However, the appearance of the reslice image is entirely insensitive to this parameter, provided it is set high enough to fill all the gaps in the slice plane. In practice, we tend to acquire B-scans separated by about 0.5mm, so we just set the thickness parameter to 0.5mm and leave it there.

Central to the success of this approach is the need to develop a *fast* implementation. To this end, we exploit standard graphics accelerator hardware, as found in many of today’s desktop computers. The wedges can be rendered using one of two techniques. The first, 2D texture mapping, defines each wedge as two polygons and projects the appropriate grey level pattern from the B-scan onto the polygons. This is the preferred technique with hardware that offers flexible, accelerated texture mapping. Alternatively, we can extract adjacent rows of pixels from the B-scans, as in Figure 8(a), and place each pixel at its correct position on the wedge. These points can then be rendered as the vertices of a dense mesh of triangles,

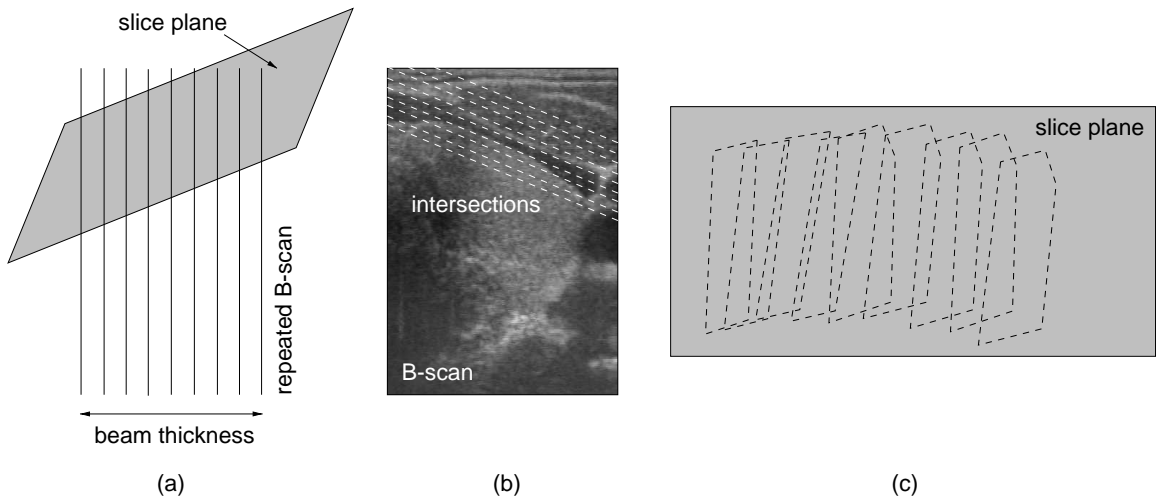


Figure 7: **Reslicing thick B-scans.** In practice, the ultrasound beam is not well focused out of the plane of the B-scan. In effect, the B-scan can be imagined repeated in space over the extent of the beam thickness, as shown in (a). The intersection of each B-scan with the slice plane is now a polygon, as shown in (b). Intensities can be extracted from within the polygons and drawn on the slice plane. This typically produces a set of filled, overlapping polygons, as shown in (c).

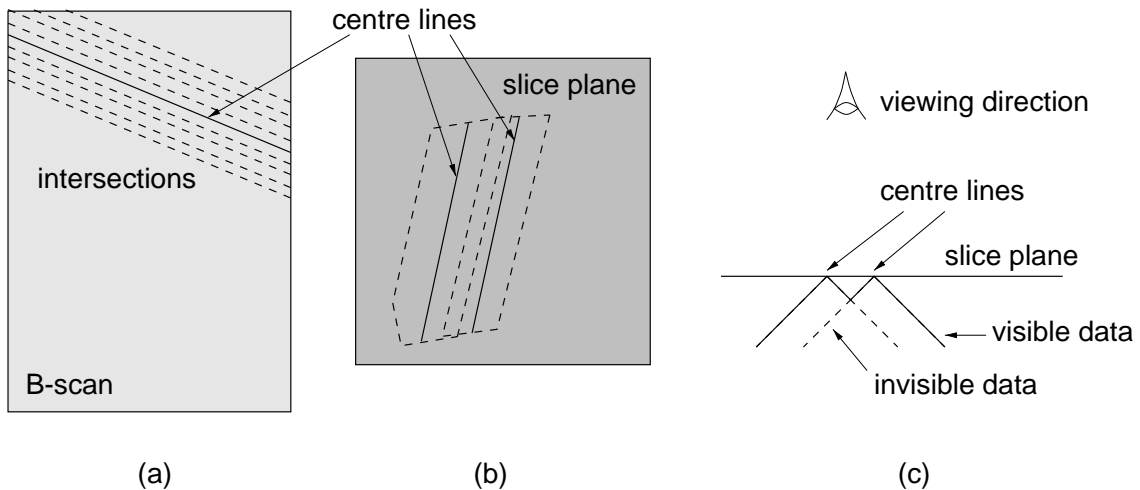


Figure 8: **Correct treatment of overlapping polygons.** Where polygons overlap, the slice should show data arising from the centre of the ultrasound beam, and not from the edges. This can be accomplished by “folding” each polygon around its centre line to form a wedge, such that the centre line is close to the viewer and the edges further away. If the graphics system is instructed to render the collection of wedges with hidden surface removal, then the slice will reveal only those portions of the polygons that are as close as possible to their centre lines.

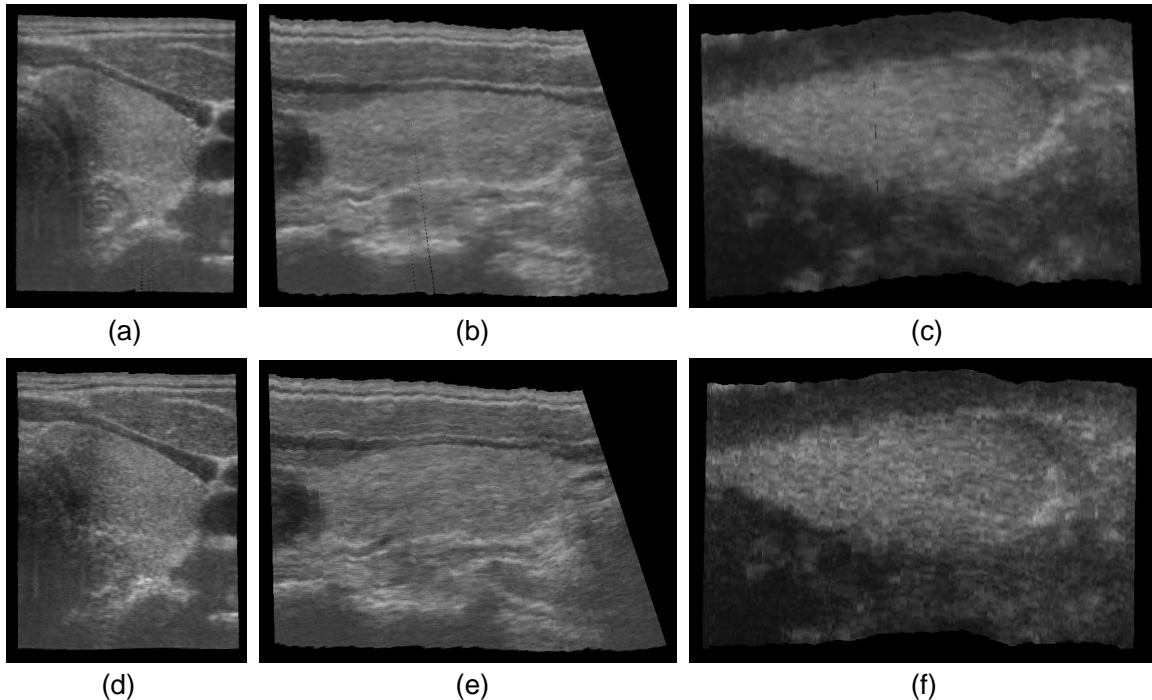


Figure 9: **Any-plane slicing of a human thyroid gland.** The figure shows results of a voxel-based (a)–(c) and voxel-less (d)–(f) approach. Slices (a) and (d) are virtually parallel to the B-scans. Slices (b) and (e) are orthogonal to these, but the skin is still at the top: similar B-scans could have been obtained by rotating the probe through  $90^\circ$ . Slices (c) and (f) are located a constant depth below the skin, and could not have been obtained using conventional 2D ultrasound.

using Gouraud shading to fill in the gaps.

Given suitable (but not necessarily extravagant) graphics hardware, the slice can be rendered very rapidly. For instance, using the Gouraud technique, several hundred B-scans can be resliced on a Silicon Graphics O2 workstation in about one second. Moreover, once the slice plane has been defined, the slice can be displayed *in real time as the ultrasound data is acquired*, providing a powerful interactive tool for the clinician. This is possible because the interpolation strategy requires information from only one B-scan at a time, in any order.

Orthogonal slices rendered using the Gouraud scheme are shown in Figures 9(d)–(f). For comparison, Figures 9(a)–(c) show similar slices through a regular voxel array constructed from the same data. The voxels were cubic, with the same dimension as a pixel in the original B-scans. Gaps in the voxel array were filled by averaging intensities in a  $5 \times 5 \times 5$  local neighbourhood. Even with this simple interpolation scheme, the voxel array took a few minutes to construct on a powerful workstation. Close inspection of the two sets of slices shows that the grey scale texture is better preserved by the voxel-less scheme, since there is *no* averaging of the raw intensity data.

### 3.2 Panoramic ultrasound

The field of view of standard B-scans is very limited. Stradx offers a panoramic imaging facility which extends this field of view by any amount, allowing long structures to be displayed in

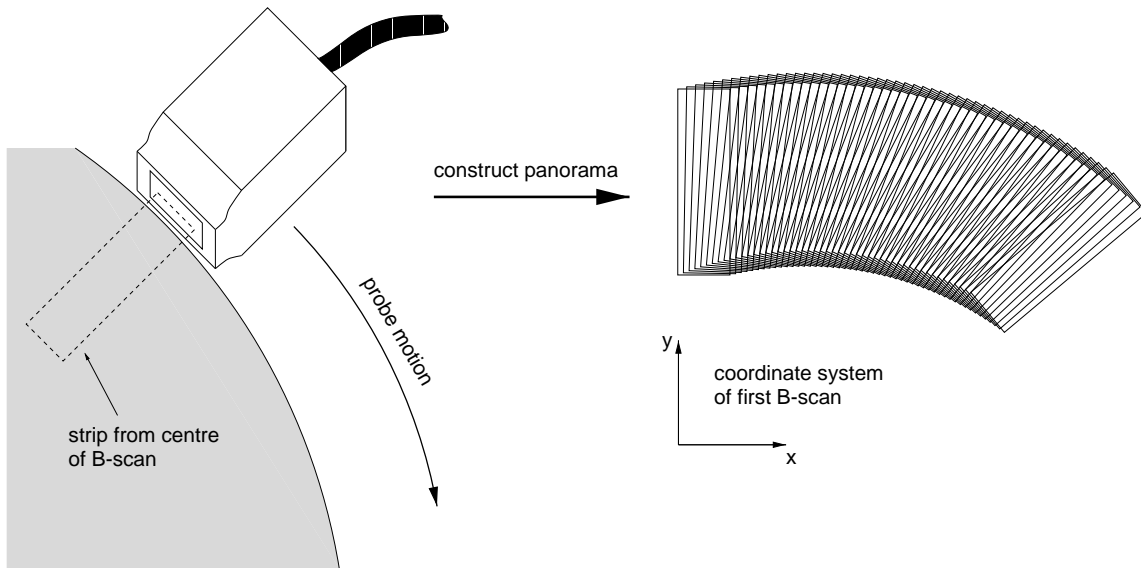


Figure 10: **Panoramic ultrasound.** The probe is moved roughly in the plane of the B-scans over the region of interest. Thin strips from the centre of the B-scans are used to construct a panoramic image in real-time.

a single image. Raw data for a panoramic image is acquired by translating the probe in the plane of the B-scans — see Figure 10. The acquired B-scans are then accurately registered and stitched together to produce a seamless composite. This can be performed in real-time, so the clinician sees the panorama being constructed as the probe is moved over the region of interest.

Recent interest in this type of imaging has been triggered by the release of the Siemens Sonoline Elegra ultrasound machine, which offers a panoramic imaging facility called “Si-eScape”. The Siemens machine achieves accurate registration by correlating consecutive images in real-time. This requires special-purpose, dedicated hardware that contributes to the high cost of the machine. We achieve a similar result using information provided by the position sensing device. Consider B-scans  $n$  and  $n + 1$ . Since we know the 3D position and orientation of both scans, it is straightforward to calculate the rigid body transformation  $\mathbf{T}$  between the local coordinate systems of the two images. We can then map pixels in B-scan  $n + 1$  into B-scan  $n$ ’s coordinate system, and display both images on the same axes.

There is one slight complication. Unless the probe is moved perfectly in the plane of the B-scans,  $\mathbf{T}$  will involve not only a within-plane translation and rotation, but also an out-of-plane transformation. We need to “flatten” this transformation to warp the surface traced out by the B-scans onto a flat plane. This can be achieved as follows. We use  $\mathbf{T}$  to transform the centre line of B-scan  $n + 1$  into B-scan  $n$ ’s coordinate system. The centre and endpoints of the centre line will not generally have zero  $z$ -coordinates in the new coordinate system. We flatten the transformation by setting these coordinates to zero. We then calculate the angle  $\theta$  and the displacement  $(\delta x, \delta y)$ , such that a within-plane rotation  $\theta$  followed by a translation  $(\delta x, \delta y)$  maps the centre of B-scan  $n + 1$  onto its projected position in B-scan  $n$ ’s coordinate system — see Figure 11. We can easily construct a planar Euclidean transformation matrix

$\mathbf{T}'$  from  $\theta$  and  $(\delta x, \delta y)$ . The panoramic image is then constructed using  $\mathbf{T}'$  to map pixels in B-scan  $n + 1$  into B-scan  $n$ 's coordinate system.

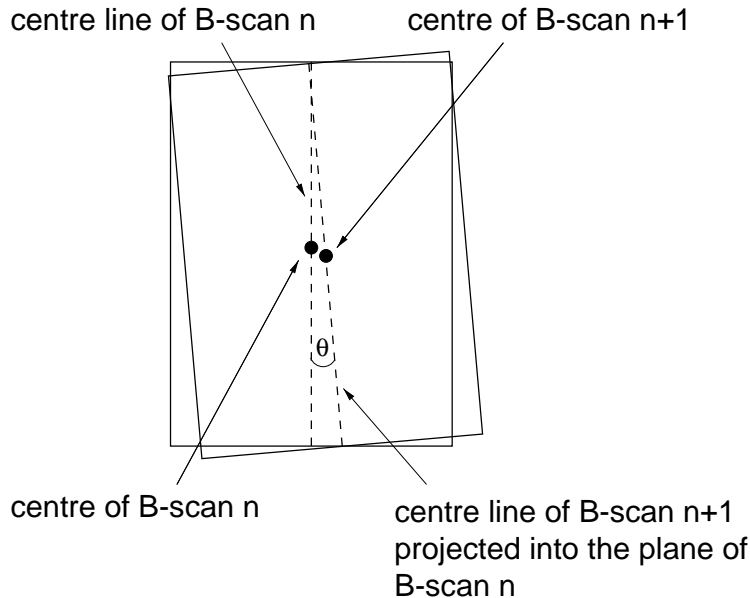


Figure 11: **Registering consecutive B-scans.** The position sensor readings are used to transfer the centre line of B-scan  $n + 1$  to B-scan  $n$ 's coordinate system. In general, the centre line will lie outside the plane of B-scan  $n$ , so its centre and endpoints are projected onto the plane by simply setting their  $z$ -coordinates to zero. It is then straightforward to measure the angle  $\theta$  and deduce the translation  $(\delta x, \delta y)$ , such that a within-plane rotation  $\theta$  followed by a translation  $(\delta x, \delta y)$  maps the centre of B-scan  $n + 1$  onto its projected position in B-scan  $n$ 's coordinate system.

Of course, we do not need to render every B-scan in its entirety to construct the panoramic image. Consecutive B-scans will share a large overlap region, and it suffices to use only a narrow strip from the centre of each B-scan. The panorama is constructed in the coordinate system of the *first* B-scan.  $\mathbf{T}'$  is calculated for each pair of consecutive B-scans, and the frame-to-frame transformations are concatenated to refer the current B-scan back to the coordinate system of the first B-scan.

Where adjacent strips overlap, we would like to see pixels arising from the centre of the strips and not the edges. This can be achieved using the wedge-based technique developed for any-plane slicing — see Figure 8. Each wedge is folded about its centre line before being painted onto the screen: hidden surface removal sees to the rest.

Note that this technique is effectively free of parameters. It is necessary to specify the width of each strip, but this can be determined automatically, using  $\mathbf{T}'$  to calculate the minimum width such that there are no gaps between adjacent strips. It is the wedge-based technique that really determines the parts of each strip that are displayed, so using slightly too large a strip width only affects the rendering speed, not the image quality.

Using the Gouraud technique, the panorama is easily rendered on a Silicon Graphics Indy workstation at the video frame rate (25 strips per second), at the same time as the clinician guides the probe over the area of interest. Typical panoramic images constructed using this

technique can be found in Figure 12.

## 4 Conclusions

We have described a freehand 3D ultrasound system which allows immediate visualisation of the data without constructing a voxel array. Great care is taken to determine the positions and orientations of the B-scans as accurately as possible. Any-plane slicing and panoramic imaging are accomplished with minimal processing of the raw data, with the aim of preserving the fine detail present in the B-scans.

In the future, we aim to extend the system to handle colour Doppler data as well as grey scale images. We are also developing novel, semi-automatic techniques for segmenting the B-scans and reconstructing the 3D shape of the segmented object from the resulting multi-planar contours.

## Acknowledgements

Stradx was developed from an earlier data acquisition system written by Patrick Gosling. Figure 1 was provided by Jonathan Carr. The reslices through the voxel array in Figure 9(a)–(c) were produced by Robert Rohling using the 3DViewnix visualisation package.

## References

- [1] C. D. Barry, C. P. Allot, N. W. John, P. M. Mellor, P. A. Arundel, D. S. Thomson, and J. C. Waterton. Three-dimensional freehand ultrasound: Image reconstruction and volume analysis. *Ultrasound in Medicine and Biology*, 23(8):1209–1224, 1997.
- [2] J. Deng, J. E. Gardener, C. H. Rodeck, and W. R. Lees. Fetal echocardiography in 3-dimensions and 4-dimensions. *Ultrasound in Medicine and Biology*, 22(8):979–986, 1996.
- [3] P. R. Detmer, G. Bashein, T. Hodges, K. W. Beach, E. P. Filer, D. H. Burns, and D.E. Strandness Jr. 3D ultrasonic image feature localization based on magnetic scan-head tracking: in vitro calibration and validation. *Ultrasound in Medicine and Biology*, 20(9):923–936, 1994.
- [4] A. Hernandez, O. Basset, P. Chirossel, and G. Gimenez. Spatial compounding in ultrasonic imaging using an articulated scan arm. *Ultrasound in Medicine and Biology*, 22(2):229–238, 1996.
- [5] S. W. Hughes, T. J. D’Arcy, D. J. Maxwell, W. Chiu, A. Milner, J. E. Saunders, and R. J. Sheppard. Volume estimation from multiplanar 2D ultrasound images using a remote electromagnetic position and orientation sensor. *Ultrasound in Medicine and Biology*, 22(5):561–572, 1996.
- [6] D. L. King, D. L. King Jr., and M. Y. Shao. Evaluation of in vitro measurement accuracy of a three-dimensional ultrasound scanner. *Journal of Ultrasound in Medicine*, 10:77–82, 1991.

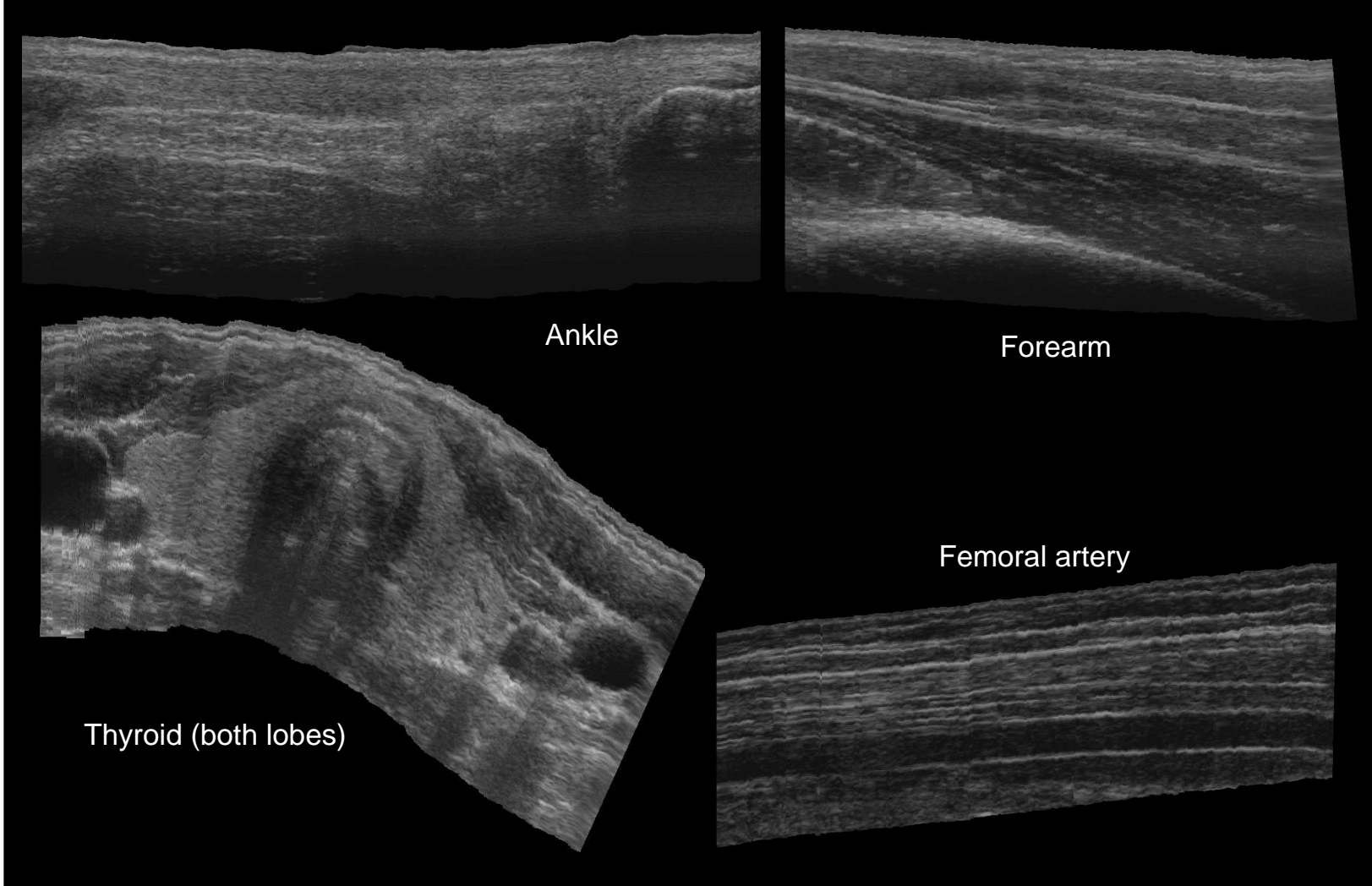


Figure 12: Typical real-time panoramic images.

- [7] D. F. Leotta, P. R. Detmer, O. H. Gilja, J. M. Jong, R. W. Martin, J. F. Primozych, K. W. Beach, and D. E. Strandness. Three-dimensional ultrasound imaging using multiple magnetic tracking systems and miniature magnetic sensors. In *Proc. IEEE Ultrasonics Symposium, 1995*, pages 1415–1418, 1995.
- [8] D. F. Leotta, P. R. Detmer, and R. W. Martin. Performance of a miniature magnetic position sensor for three-dimensional ultrasound imaging. *Ultrasound in Medicine and Biology*, 24(4):597–609, 1997.
- [9] A. Moskalik, P. L. Carson, C. R. Meyer, J. B. Fowlkes, J. M. Rubin, and M. A. Roubidoux. Registration of three-dimensional compound ultrasound scans of the breast for refraction and motion correction. *Ultrasound in Medicine and Biology*, 21(6):769–778, 1995.
- [10] T. R. Nelson and T. T. Elvins. Visualization of 3D ultrasound data. *IEEE Computer Graphics and Applications*, pages 50–57, November 1993.
- [11] R. Ohbuchi, D. Chen, and H. Fuchs. Incremental volume reconstruction and rendering for 3D ultrasound imaging. *Proceedings of SPIE — The International Society for Optical Engineering*, pages 312–323, 1992.
- [12] R. W. Prager, R. Rohling, A. Gee, and L. Berman. Rapid calibration for 3-D freehand ultrasound. *Ultrasound in Medicine and Biology*, in press.
- [13] R. W. Prager, R. Rohling, A. Gee, and L. Berman. Automatic calibration for 3-D freehand ultrasound. Technical Report CUED/F-INFENG/TR 303, Cambridge University Department of Engineering, September 1997.
- [14] R. N. Rankin, A. Fenster, D. B. Downey, P. L. Munk, M. F. Levin, and A. D. Vellet. Three-dimensional sonographic reconstruction: techniques and diagnostic applications. *American Journal of Roentgenology*, 161(4):695–702, 1993.
- [15] R. Rohling, A. Gee, and L. Berman. Three-dimensional spatial compounding of ultrasound images. *Medical Image Analysis*, 1(3):177–193, April 1997.
- [16] M. L. Skolnick. Estimation of ultrasound beam width in the elevation (section thickness) plane. *Radiology*, 180(1):286–288, 1991.
- [17] A. State, D. T. Chen, C. Tector, A. Brandt, H. Chen, R. Ohbuchi, M. Bajura, and H. Fuchs. Case study: Observing a volume rendered fetus within a pregnant patient. In *Proc. IEEE Visualization, 1994*, pages 364–368, 1994.
- [18] H. Steiner, A. Staudach, D. Spitzer, and H. Schaffer. Three-dimensional ultrasound in obstetrics and gynaecology: technique, possibilities and limitations. *Human Reproduction*, 9(9):1773–1778, 1994.
- [19] J. W. Trobaugh, D. J. Trobaugh, and W. D. Richard. Three-dimensional imaging with stereotactic ultrasonography. *Computerized Medical Imaging and Graphics*, 18(5):315–323, 1994.