Three dimensional displays of tomographic images using shaded surfaces

Christopher John Gibson

THREE DIMENSIONAL DISPLAY OF TOMOGRAPHIC IMAGES USING SHADED SURFACES

by

Christopher John Gibson

A thesis submitted to the Faculty of Science of the University of Durham for the degree of Doctor of Philosophy

April 1988

Three Dimensional Display of Tomographic Images using Shaded Surfaces

C. J. Gibson

Abstract

Several medical imaging techniques are capable of producing tomographic images, corresponding to cross-sections through the body. A stack of adjacent sections contains three dimensional information about the organs of interest, and this can be presented on a two dimensional screen using shaded surface techniques.

In order to facilitate the routine use of such images, algorithms and techniques were developed on a conventional medical imaging computer system in a hospital environment. Several object representation schemes were compared, and two new schemes were devised. The 'solid binary object' technique facilitated exploration of the interior of an object, while the 'ordered surface list' technique enabled real time display of object surfaces.

Several shading algorithms were compared, and a local polynomial fitting routine was devised. This was found to be superior to other methods using objective evaluation of the accuracy of surface normal estimations, and subjective evaluation of the corresponding image appearance.

The techniques developed were applied to a variety of data obtained using xray computed tomography, nuclear magnetic resonance and emission computed tomography. For display of myocardial tomograms, a technique was devised for superposition of colour coded coronary arteries, showing their relationship to observed perfusion defects. For display of time varying images of the heart, a rapid display routine was developed to enable ventricular wall motion to be evaluated from any angle. Colour display techniques were also applied to this data to produce single images which incorporated kinetic as well as morphological information.

The results obtained have confirmed that shaded surface images can be produced using computers currently available in hospital imaging departments. Interactive object modification and real time object display can be achieved without requiring special hardware.

Contents

Contents

Acknowledgements

I should like to thank the following people for their assistance in the preparation of this thesis :-

Dr. C. Preece for supervision throughout the project, and for his advice on all matters from initial planning to final presentation.

Dr. B. L. Diffey for advice and comments on several techniques described in this work.

Professor K. Boddy and the Regional Medical Physics Department for encouragement and financial support.

Dr. E. D. Williams and Dr. E. E. Laird for providing tomographic images of myocardial perfusion, and for suggesting the idea of arterial superposition.

Dr. J. K. Haywood, Mr. D. Mott, Mr. T. A. Reeder and Dr. G. Johnson for providing additional tomographic image data.

Mr. P. Grencis for photographing the figures, and Mrs. S. Lumsden for typing the manuscript.

Most of all I should like to thank Yvonne, who worked as hard as I did, and who deserves a good deal of the credit.

Symbols and Abbreviations

Bold print indicates a vector or matrix quantity, ordinary print indicates a scalar.

2-D	Two dimensional.
3-D	Three dimensional.
θ	Angle Theta, used in viewing transformations.
ϕ	Angle Phi, used in viewing transformations.
a	Semi axis of ellipse.
a	**Apex of the left ventricle, defined on the arterial tree.**
A	Angle used in coordinate rotation matrices.
A	**Apex of the left ventricle, defined on the myocardium.**
$a_{0..2}$	Parameters used in the polynomial fitting procedure.
$A_{1..4}$	Angles used in the coronary artery fitting procedure.
b	Semi axis of ellipse.
$B_{1..2}$	Angles used in coronary artery fitting procedure.
BTF	Back-to-front traversal (of an object array).
C	Coordinate of the centre of the object array.
C_m	Number of comparisons between algorithms.
c_p	Cos ϕ
c_t	Cos θ
CPU	Central processing unit of a computer system.
d	Distance from the observer to a visible voxel.
d_o	Minimum distance to a visible voxel.
d_1	Maximum distance to a visible voxel.
D_r	Intensity of directional illumination.
E	Error angle used to assess gradient estimators.
ECAT	Emission Computed Axial Tomography.
ED	End Diastole.

ES	End Systole.
f(d)	Function describing attenuation of isotropic light.
F(x',y')	Function in observer space describing visible surfaces.
F_w	Factor by which object width increases after smoothing.
g(d)	Function describing attenuation of directional light.
G(x',y')	Function in observer space describing visible surfaces.
Ga-67	Gallium-67
h(**i**,**s**,**o**)	Function describing reflection of directional light.
HLS	Hue, lightness and saturation of a colour.
i	**Unit vector along the direction of incident ilumination.**
I	Angle of incidence.
I_s	Intensity of isotropic ilumination.
J	Angle between reflected ray and the viewing direction.
k	Integer counter used in schematic image display algorithms.
l	Direction cosine of the viewing direction along the x axis.
L	Length of the surface normal vector, used for normalisation.
l_{na}	Distance from point **n** to point **a**.
L_{NA}	Distance from point **N** to point **A**.
lsb	Least significant byte in a 16 bit word.
m	Direction cosine of the viewing direction along the y axis.
m	**Intermediate point used in fitting coronary arteries.**
M	Parameter used to modify the cosine shading law.
m_p	Number of points in a closed outline.
msb	Most significant byte in a 16 bit word.
n	Direction cosine of the viewing direction along the z axis.
n	**Coronary artery node point, defined on the arterial tree.**
N	Linear sample size for a cubic array of N^3 voxels.
N	**Coronary artery node point, defined on the myocardium.**

n_p	Number of points in a closed outline.
NMR	Nuclear Magnetic Resonance.
$\mathbf{o}$	**Unit vector along the observer's viewing direction.**
O(q)	A process requiring q operations.
p	Parameter used to modify the cosine shading law.
P	Number of preferences expressed when comparing images.
r(I)	Function describing reflection from a transparent surface.
$\mathbf{r}$	**Row vector specifying a point in object coordinates.**
$\mathbf{r}'$	**Row vector specifying a point in observer coordinates.**
R	The number of different 3-D representations of object(s).
$\mathbf{r_n}$	**Row vector specifying a given point in object coordinates.**
$\mathbf{r_{n+1}}$	**Row vector specifying adjacent point in object coordinates.**
$\mathbf{r_n}'$	**Row vector specifying a given point in observer coordinates.**
$\mathbf{r_{n+1}}'$	**Row vector specifying adjacent point in observer coordinates.**
RCA	Right coronary artery.
RGB	Red, green and blue components of a colour.
s	Scaling factor for coordinate stretching.
$\mathbf{s}$	**Unit vector along the local surface normal.**
$\mathbf{s}'$	**Averaged surface normal vector.**
S	Shade value stored in pixel of shaded surface image.
$\mathbf{S}$	**Unnormalised local surface normal.**
S_c	Cosine shade value.
S_d	Depth shade value.
$\mathbf{s_e}$	**Estimated (normalised) surface normal on an ellipse.**
$\mathbf{S_e}$	**Estimated (unnormalised) surface normal on an ellipse.**
S_{mc}	Modified cosine shade value.
s_p	Sin Ø
S_p	Polynomial shade value.

s_t	Sin θ
$\mathbf{s}_t$	**True (normalised) surface normal on an ellipse.**
S_t	**True (unnormalised) surface normal on an ellipse.**
T	**A 4 x 4 transformation matrix.**
$T_{1..6}$	**Transformation matrices used to derive viewing transforms.**
T_{ij}	Element at row i, column j of matrix T.
T_x	**Transformation to rotate about the x axis.**
T_y	**Transformation to rotate about the y axis.**
T_z	**Transformation to rotate about the z axis.**
Tc-99m	Technetium-99m.
Tl-201	Thallium-201.
u	Normalisation value used in the general viewing transformation.
v	Normalisation value used in the general viewing transformation.
V_o	Object variance before smoothing.
V_1	Variance of a function.
V_2	Variance of a function.
W	Coefficient of concordance.
x	Coordinate in object space.
x	**Unit vector along the x axis.**
x'	Coordinate in observer space.
X	Dummy coordinate parallel to x', origin at the current pixel.
x_o	Coordinate of the origin of a translated coordinate system.
x_a	Coordinate of point a.
X_a	Coordinate of point **A**.
x_m	Coordinate of point **m**.
x_n	Coordinate of point **n**.
X_n	Coordinate of point **N**.
y	Coordinate in object space.

y'	Coordinate in observer space.
y_o	Coordinate of the origin of a translated coordinate system.
y_a	Coordinate of point **a**.
Y_a	Coordinate of point **A**.
y_m	Coordinate of point **m**.
y_n	Coordinate of point **n**.
Y_n	Coordinate of point **N**.
z	Coordinate in object space.
z'	Coordinate in observer space.
Z	Dummy coordinate parallel to z', origin at the current pixel.
z_o	Coordinate of the origin of a translated coordinate system.
z_a	Coordinate of point **a**.
Z_a	Coordinate of point **A**.
z_m	Coordinate of point **m**.
z_n	Coordinate of point **n**.
Z_n	Coordinate of point **N**.
Z_s	Normal standard deviate.

Index to Figures

1. Introduction

Medical imaging techniques form some of the most powerful diagnostic tools available to the clinician. Photons in the visible and infrared regions of the electromagnetic spectrum have very limited penetration into the body, and are therefore of most value for examination of the skin, and superficial organs such as the ears and eyes. Nevertheless, the hue of reflected light can indicate systemic disorders (e.g. the 'rubor' of the early physicians), and surface temperature may suggest vascular abnormalities. Transillumination techniques can be used with small organs such as the breast and neonatal head.

The discovery of xrays was followed by their immediate application to medical diagnosis, and led to the development of the medical specialty of radiology. Conventional radiographs are formed by transmission of xrays through the body, and contrast in the image depends on the existence of differences in xray attenuation between normal and pathological tissues. The production of man-made short lived radionuclides has enabled the development of radionuclide imaging techniques in which small quantities of gamma ray emitting radionuclides are administered to the patient, usually by intravenous injection. The subsequent distribution of radioactive material depends on the chemical form of the radionuclide, and on the current physiological state of the body. Detection of the emitted gamma rays produces images in which contrast is determined by the function of the organ being imaged.

In both radiography and radionuclide imaging, the imaging parameter (xray attenuation coefficient and concentration of radionuclide respectively) is a three dimensional scalar function. Conventional

imaging techniques 'project' this distribution onto a two dimensional plane, thereby losing information about the distribution along the direction of projection. Frequently several views are obtained at standard projections (e.g. anterior, lateral or oblique to the body) to provide at least some information about the true distribution. From such views a skilled observer may be able to locate a lesion in three dimensions, providing that it is visible on at least two views.

However, this is not an easy task, and the projection images suffer an irrevocable loss of contrast caused by the summation of imaging parameter values along the projection direction. As a consequence tomographic imaging techniques have been developed to reconstruct the original three dimensional function. The data in each pixel of a projection image represents a line integral through the imaging parameter distribution along the line of projection. This representation is reasonably exact for xray attenuation, but is only approximate for radionuclide concentration because of the attenuation of gamma rays within the body. Nevertheless, if a large number of such line integrals are obtained at many angles around a three dimensional function, then that function can be reconstructed by back-projection. The images obtained are transverse sections through the body, and are known as 'computed tomograms', because the reconstruction procedure inevitably makes use of a digital computer. The term 'CT' image has come to mean a tomogram obtained using transmitted xrays, whilst those obtained using emitted gamma rays are usually referred to as 'ECAT' images (emission computed axial tomograms).

Several other imaging techniques are also used in medicine. Partly because of their physical nature, and partly because of the known advantages, attention has concentrated on the development of tomographic

rather than projection techniques.

Positron emitting radionuclides can be used in a manner identical to conventional gamma ray emitting radionuclides. They have the advantage that the pair of 511 keV photons arising from positron annihilation are emitted in opposite directions and can be imaged using uncollimated coincidence detectors. Unfortunately the short half life of many positron emitters has restricted this technique to only a few centres, with an on-site cyclotron.

The human body is a reasonably good transmitter of sound waves with frequencies around 2-10 MHz. If pulses of such sound are directed into the body from a transducer placed in contact with the skin, echoes are received from interfaces between structures with different values of acoustical impedance. This technique is known as ultrasonic scanning, and is widely used, particularly in obstetrics, because of the non-ionising nature of ultrasound. The images obtained are inherently tomographic, since the spatial location of each echo forming the image is determined uniquely by the position of the transducer and the time between transmission of the pulse and receipt of the echo. In a typical device a linear array of transducers is used to create a real time image of the echo distribution in a plane defined by the orientation of the transducer array and the direction of transmission into the body. However, there are problems in obtaining a complete sampling of the three dimensional distribution. The transmission of ultrasound into the body can only be achieved using close contact between the transducer and the skin. To facilitate this the transducer is hand-held, and a coating of oil is spread on to the skin. Although this enables the observer to make rapid changes to the transducer orientation, and hence to alter the section displayed, it makes it difficult to obtain the set of parallel

equally spaced sections required for complete three dimensional sampling.

The observation that certain nuclei, including hydrogen, have a net spin and hence an associated magnetic moment, has led to the development of nuclear magnetic resonance (NMR) imaging. This technique is unique in many ways, not least because the signal strength is determined by several components: the proton density, the spin-lattice relaxation time and the spin-spin relaxation time. In a sense therefore, the imaging parameter forms a three dimensional vector field. In practice a chosen sequence of excitation pulses and field gradients produces a scalar image in which contrast is mainly determined by one or other of these components. As well as reconstruction from line integral projections, in a manner analogous to CT or ECAT images, the imaging parameter can be reconstructed directly, both along any two dimensional plane, and in three dimensions. Of all the medical imaging techniques this is the most truly three dimensional.

The advantages of cross-sectional imaging are increased contrast and complete sampling. However, the interpretation of such images can be difficult, because of the complex shape of many body organs. This has required the development of particular skills for interpreting tomograms, and has led to an increased interest in cross-sectional anatomy. However, the fact that large numbers of such tomographic images are now being produced has stimulated consideration of alternative display techniques. Various methods have been proposed for three dimensional display and these are reviewed in Section 2. The choice of technique is influenced both by the available computing and display hardware, and by the degree of operator interaction required. Three dimensional display techniques will find their greatest

application if they can be made available in the clinical environment, without requiring significant extra investment or resources. This project was initiated in 1983 with the aim of developing a three dimensional display facility on a conventional medical imaging computer, available in many hospitals in this country. Such systems were equivalent to laboratory minicomputers of the same date, with approximately 128 Kbytes of memory, one or two workstations and magnetic disks of a few Mbytes capacity. The display screens were conventional two dimensional raster graphics displays, with typical image memories of 64 to 256 Kbytes. During the period of the project the performance of such systems has gradually increased, providing more memory, larger disks and additional workstations. However, there have been no radical or qualitative changes in the design or operation of medical image processors.

Several three dimensional display techniques are suitable for use on such a system, and the 'shaded surface' method was chosen for further study. This can be thought of as an attempt to display the object of interest as if it could be seen using visible radiation, or as if it were extracted from the body and then photographed. The 'object' is a subset extracted from the three dimensional distribution of the imaging parameter, which is of interest to the observer. Depending on the clinical situation, this object may be one or several of the actual body organs in the volume imaged. In the case of ECAT data, it is best thought of as a region defined by a common degree of function, rather than just an anatomical structure. Several object representation techniques have been devised for use with conventional medical imaging systems, and these are assessed and further developed in Section 3.

Once the object has been modelled in the computer, it is necessary

to specify the aspect to be displayed, determine the visible surfaces, and render those surfaces on the display screen using an illumination model. Although the final image is two dimensional, it simulates the appearance of a solid opaque object, and enables the observer to perceive three dimensional shape. Display techniques for solving the hidden surface problem are described in Section 4, and shading algorithms are described in Section 5. In both cases novel techniques have been developed, and are compared with existing methods, using data derived from CT, ECAT and NMR images.

The methods described in this thesis have been applied to a variety of clinical conditions, and the results are described in Section 6. The value of shaded surface display techniques will have to be established using clinical trials, comparing the utility of a three dimensional display with conventional 'slice' mode displays. However, there are some circumstances in which a full set of adjacent slices can be obtained at no extra cost, e.g. when performing ECAT studies using a rotating gamma camera. In such cases, a three dimensional display mode which makes use of existing hardware is essentially free, and would be justified even if the gain in comprehension was only slight. In practice the results from this work, and those published by other workers, suggest that shaded surface images do provide considerable additional information which is not apparent using conventional tomographic display methods.

2. Review of Three Dimensional Display Techniques

2.1 Slice and Projection Displays

The raw data for three dimensional display techniques consists of a set of transverse sections. Each slice can be displayed as a two dimensional digital image on a conventional raster graphics system, using the value of the imaging parameter to control the brightness of the corresponding pixel in the display.

Multiple slices can be displayed simultaneously if the system has sufficient screen memory (Figure 2.1). It has been suggested (Williams 1983) that a rapid movie sequence showing successive transverse slices can provide additional three dimensional information, since the observer integrates the perception of structures seen in one slice with those in adjacent slices. An alternative technique displays selected widely spaced slices on the same screen as if seen from an oblique angle (Jackson et al 1983). Unfortunately, in order to convey three dimensional information, the slices must be displayed close together, and parts of underlying slices are therefore obscured (Figure 2.1).

Tomographic sections are usually obtained in the transverse plane. The pixel size and slice spacing depend on the imaging modality and on the organ of interest. A typical CT section consists of 256 x 256 pixels each approximately 2 mm square, but the slice spacing is generally coarser, and may be 5-10 mm or more. However, the spacing can be adjusted to be equal to the pixel size by linear interpolation between slices. The resulting cubic array represents a three dimensional sampling of the volume of interest, in which each element corresponds to a small volume of space. The term 'voxel' has been used to describe such an element. Such an array of voxels can easily be reformatted to obtain orthogonal slices along either coronal or sagittal

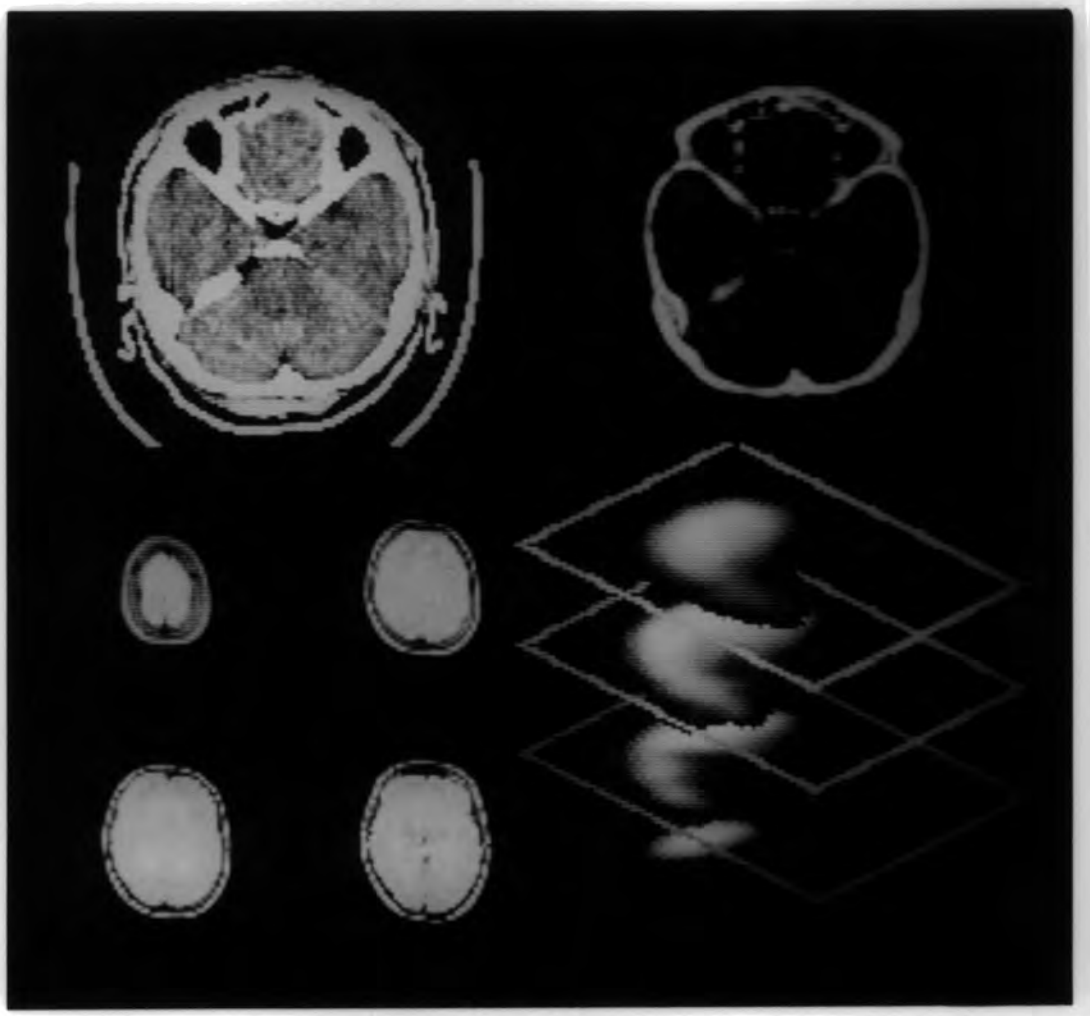


Figure 2.1 A representative selection of tomographic data displayed as slices :-

Upper left - CT slice through the skull.
Upper right - The same slice showing only bony tissues.
Lower left - NMR slices through the brain.
Lower right - ECAT data, showing the liver from the left anterior oblique aspect.

planes (Maravilla 1978). Using more complex techniques, oblique slices may also be obtained from the same set of original data (Borrello et al 1981). Oblique slices have been reported to be particularly useful in cardiac imaging (Akins et al 1985), where the principal axes of the left ventricle are not aligned with transverse sections through the chest (Olson et al 1981).

Orthogonal and oblique slices are both planar, but many body organs are curved. One example is the spine, which curves markedly in the sagittal plane. Conventional coronal sections are therefore difficult to interpret, since the spine curves in and out of each plane. Modified 'coronal' sections have been constructed (Rothman et al 1984), which follow the natural curvature of the spine, hence ensuring that all the vertebral bodies appear on each slice.

All these displays show only a few slices at a time, and rely on the observer's ability to integrate the appearance of an object in several adjacent slices to determine three dimensional shape. The difficulty of this task forms the justification for attempts at three dimensional display.

The set of cubic voxels can also be used to create re-projection images, formed by additive projection of the imaging parameter onto a two dimensional plane (Harris 1982). This seems at first sight to be a retrograde step. Projection images are widely used in medicine (e.g. radiographs, gamma camera images etc), and often form the raw data from which the tomographic sections are reconstructed. However, projection images are not normally obtained using imaging modalities such as Computed Tomography and Nuclear Magnetic Resonance, and they can be useful for conveying an immediate impression of the size and shape of an organ. The density at a point in the re-projection image is obtained by

summing the value of the imaging parameter along a line through the original sections perpendicular to the plane of projection (Figure 2.2). Such images can be obtained at any angle around the patient, including angles which could never be obtained using conventional imaging techniques, e.g. showing the superior or inferior aspects of an internal organ. However, the principal advantage of the re-projection technique is the ability to selectively enhance features of interest during re-projection (Lenz et al 1986). Such enhancement may be based on two criteria:-

a) Spatial - the summation is weighted according to a function of position, so that tissues at selected distances or in selected regions contribute more to the re-projection image

b) Intensity - the summation is weighted according to a function of the imaging parameter, so that tissues corresponding to selected organs contribute more to the re-projection image

Using either or both of these criteria the re-projected images show features which may not be apparent on conventional projection images.

Because of the relative simplicity and ease with which conventional projection images can be obtained, attempts have been made to improve depth perception without the need for tomographic reconstruction. A pair of projection images obtained with a small angular difference can be used as a stereo pair (Charkes 1971). If many projection images are obtained at small angular increments around the patient, a rapid movie display of the set gives good depth perception from the motion parallax effect (Williams 1981, Keyes and Rogers 1981). However, a study comparing this form of display with static imaging in the assessment of

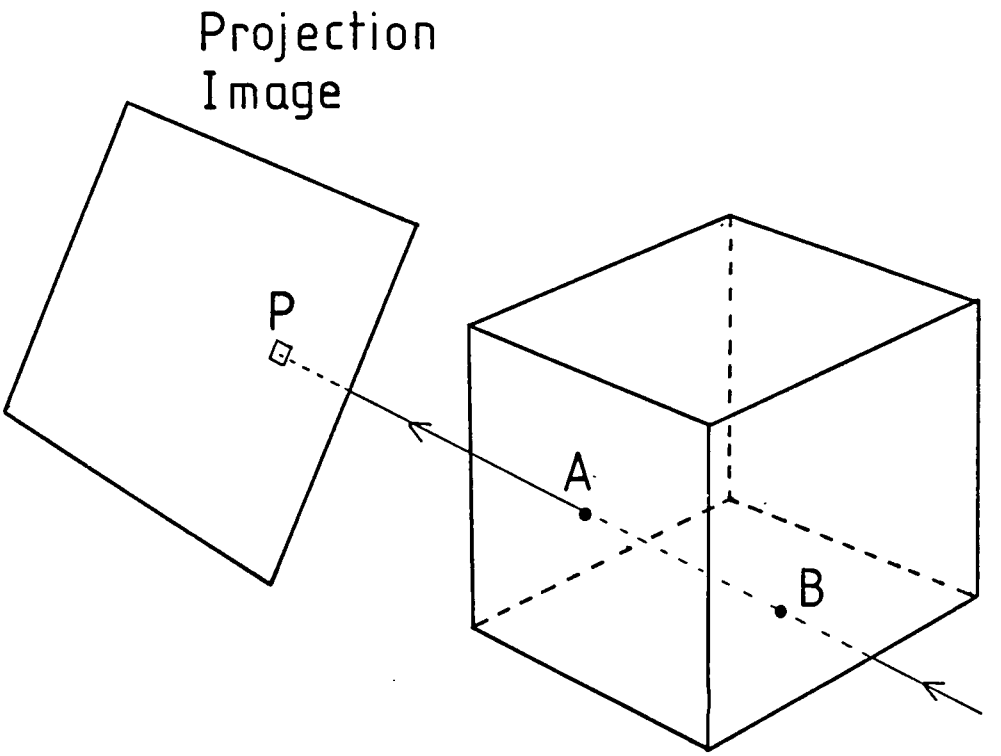


Figure 2.2 Pixel P contains a value derived from the linear integral of voxel values along the line AB, which is perpendicular to the plane of projection.

liver disease found no significant improvement (Kalff et al 1984), although it was considered a useful aid to the interpretation of liver tomograms (Keyes et al 1984). The motion parallax effect has also been applied with simultaneous stereoscopy (Barber and Skellas 1982), and using digitally re-projected images (Keyes 1982).

Re-projection techniques have been used to create polar maps of the uptake of Thallium-201 in the left ventricular myocardium (Garcia et al 1985). Circular or 'bullseye' images are created showing the uptake at the ventricular apex in the centre of the image, with successive rings indicating the summed uptake in the ventricular wall, moving from the apex to the valve plane. These images are able to show the variation in uptake around the walls of the ventricle, but have to sacrifice anatomical realism. A similar approach has been used to display position emission tomograms (Kehtarnavaz et al 1984), using the re-projected values to colour code a 'cylindrical' representation of the ventricular surface.

The major use of re-projection techniques has been to display CT data in a readily comprehensible form. This has been of particular value for the display of angiographic data (Block et al 1984), where the complex shape of an arterial tree is very difficult to interpret using cross-sectional images (Liu et al 1986). There has also been an attempt to display re-projection images from CT data acquired using only a limited number of views (Jaman et al 1985). The results were only satisfactory for re-projection close to the angles at which the original data was obtained.

In a more interesting variation of the re-projection idea, an image was created by summation not of the original tomographic data, but of 'colour gel' values derived from them (Fishman et al 1987). The

reconstructed data in each voxel was used to identify the tissue type (e.g. bone, muscle, fat etc.), and the corresponding value added to the re-projection image had a hue and transparency defined by that type. The result was an image with a translucent appearance in which thicker volumes of tissue were more opaque than thin volumes, and in which colour indicated the relative amounts of each tissue type. The method has been developed by Lucasfilm (San Rafael, California) for commercial applications, and exact details of the technique have not been published.

Re-projection images have the ability to display information from many slices simultaneously. When used with radionuclide tomograms they are able to include functional information, by summation of the total uptake in parts of an organ, but it is difficult to present this data and depth information simultaneously. Because they make use of the reconstructed data directly, an intermediate surface or solid model is not required. Re-projection is therefore most likely to be useful in cases where the object cannot be readily isolated from the surrounding tissues. However, a re-projected image does not convey an impression of solidity. The image grey scale value is dedicated to the display of integrated voxel values, and so is unavailable for the provision of depth cues. Attempts can be made to combine hue and intensity to overcome this limitation, but there are limits to the number of colour combinations which can be either displayed or comprehended. Techniques which render only the surfaces will be better able to convey object morphology, since they can concentrate on the display of surface location and orientation, enabling the observer to perceive the shape directly.

2.2 Space Filling Displays

A cubic array of voxels forms a three dimensional scalar field, and there are no natural analogies for the perception of structures or objects in such a field. Nevertheless, a number of attempts have been made to display all the tomographic data simultaneously. These 'space filling' displays incorporate visual cues such as stereopsis and head movement parallax to form truly three dimensional images. The most obvious technique to use is holography, preferably using a white light viewing system. Initial results have been reported using multiple CT slices (Greguss 1977), but the time consuming photographic procedures required to produce the hologram preclude interactive display.

Other optical techniques reported make use of mirrors and a two dimensional display system. Successive slices are displayed at a rapid refresh rate, either using a closed loop of film (DeMontebello 1977) or a conventional raster graphics screen (Pizer et al 1982). Images of each slice are made to appear at slightly different spatial locations, and if the refresh rate is sufficiently high the observer integrates these images into a three dimensional whole. In the most common implementation, a vari-focal mirror is used to reflect images of each slice which are displaced in space according to the curvature of the mirror (Baxter and Hitchner 1982, Anderson et al 1982). The mirror is a silvered mylar membrane which oscillates in front of a loudspeaker face, and by synchronising the vibration of the mirror with the appearance of slices on the display screen, each reflected slice appears at a unique position in space. Because the image is truly three dimensional, it can be inspected from several aspects by moving around the display screen. The viewing angle is restricted, since images can only be seen from one

side of the mirror, and because of the size of the ring in which the membrane is held. In one implementation the three dimensional image forms a cube with volume 18 x 18 x 18 cm (Harris et al 1986), and is produced using a mirror oscillating at 30 Hz. Systems using varifocal mirrors have been produced commercially by the Genisco Corporation, and one such has been applied to the display of NMR images (Kennedy and Nelson 1987).

Two fundamental problems affect these display systems. The first is purely technical, to display the slices at a sufficiently high display rate. If the mirror oscillates at 30 Hz, and an array of N x N x N voxels is to be displayed, each represented by a single byte, then data transfer rates of 60 N^3 bytes per second are required. Note that transfer can only take place during half the cycle, as the mirror moves either backwards or forwards, since the positions of the reflected images are slightly different for the two directions of movement. If the object is sampled at 128 x 128 x 128 resolution, transfer rates of over 120 Mbytes per second are required. This problem has been circumvented by reducing the amount of data displayed. Either a smaller subset of voxels are selected from the raw data in a random pattern (Fuchs et al 1982), or the data are compressed by addition of adjacent slices along the direction perpendicular to the surface of the mirror (Harris et al 1986). In the latter case up to 27 slices at 128 x 128 resolution could be displayed, using a special purpose high speed image memory.

The second fundamental problem relates to the attempt to produce an image of a three dimensional field, and is therefore unavoidable. Because all the data is displayed, objects of interest can be obscured behind 'clouds' of intervening voxels. This can be obviated by removing unwanted voxels from the scene, and criteria based on both spatial

location and imaging parameter have been used for this purpose. A look up table technique enables real time modification based on the value of the imaging parameter, allowing the operator to rapidly highlight structures of interest. Nevertheless the system has been described by one user as being excellent for line drawings and good for surfaces, but having weaknesses for display of tomographic sections (Pizer et al 1982).

Other methods for producing an image of the entire three dimensional array have also been described (Hesselink 1985). A vertical two dimensional array of light emitting diodes can be rapidly rotated about a vertical axis through the centre of the array. As the array sweeps out a cylindrical volume, the diodes are illuminated in proportion to the intensity of the imaging parameter at the corresponding location (Jansson and Kosowsky 1984). There are some problems in converting the Cartesian array of tomographic data into a cylindrical coordinate representation, but this method does use a solid state display with a relatively simple mechanical system. It is probable, however, that the perceptual difficulties associated with space filling displays would still remain.

A very sophisticated system incorporating voice input and twin display screens has been proposed for the display of cardiovascular data (Wixson 1983). The space filling display is to be produced using pairs of stereoscopic reprojected images, created off-line, and subsequently viewed as a rapidly rotating sequence. Alternatively, a set of 32 pairs of stereo images can be produced with different planes of sharp focus. Infrared eye trackers locate the observer's current area of interest, and calculate the distance to that point. The corresponding stereo pair is then displayed, giving the effect of a

binocular microscope in which eye movements alter the plane of focus.

Space filling displays are genuinely three dimensional images, and as such would have many advantages. Perception of depth would be immediate and intuitive, several observers could discuss a case simultaneously (since special glasses are generally not required), and alteration of the angle of view would frequently be as simple as moving to the other side of the image. The problem of object obscuration has already been referred to, but there are also other disadvantages. Hard copy of the image can be obtained by photographing the display, but this will include few three dimensional depth cues. Special purpose hardware is required, often with rapidly moving mechanical parts, which may prove unsuitable for the clinical environment. Since one aim of this work was to develop a three dimensional display facility using a conventional medical image processing system, space filling displays were not considered further.

2.3 Outline and Tiling Displays

Techniques described in this section and those below have as their objective the detection of the surfaces of the object(s) of interest, and the display of those surfaces on a two dimensional graphics screen. This simulates the way in which objects are perceived in the natural world, as opaque, translucent or transparent surfaces from which their shape is determined. The simplest technique for representing surfaces extracted from a stack of cross-sections is in the form of contours. The object is described by a set of two dimensional closed outlines, obtained either by interactive drawing or by a more objective segmentation procedure. Conventional computer graphics techniques for three dimensional transformation and display may then be used to display line images of the set of outlines as seen from any angle. These

techniques have been applied to serial histological sections (Chawla et al 1982, Perkins and Green 1982) and to organ outlines obtained from CT scans (Fram et al 1982, Siddon and Kijewski 1982). The appearance of such wire frame images is improved by eliminating hidden lines (Chamayou et al 1982), but the representation of a complex object by a stack of contours can still be confusing. Nevertheless this method has been applied to the display of the vertebrae (Verbout et al 1983).

Given a set of outlines representing the borders of an object, it is easy to imagine a surface formed by stretching a cover over the wire framework of the contours (Figure 2.3). This surface can be most conveniently created as a set of polygonal patches, with vertices derived from points in adjacent contours. Several algorithms have been developed using this 'tiling' scheme, differing in the procedure used to find the best surface.

If a set of m_p points represents a closed object outline in one slice, and a second set of n_p points represents the outline in an adjacent slice, the problem is to find a set of polygonal patches which produce an acceptable surface. One technique is to re-sample the contour which has fewer points to obtain two contours with equal numbers of points. Quadrilateral patches are then defined by linking together corresponding points 1, 2 ... n_p in each outline (Mazziotta and Huang 1976). Another approach uses B-splines to approximate the contours within each slice, and then the surface between contours is interpolated using cardinal splines (Sunguroff and Greenberg 1978). This technique produces smooth surfaces from sparse data (e.g. coarsely spaced slices) but is computationally very demanding.

Most tiling techniques use triangular surface patches, with vertices formed from two points in one contour and one in the adjacent

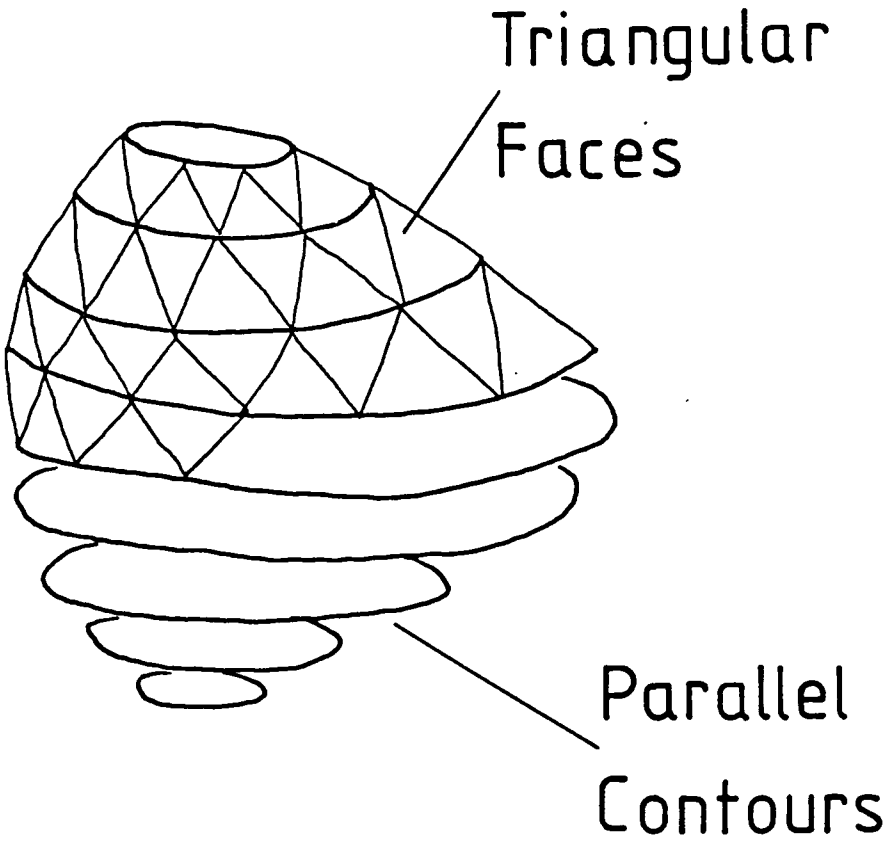


Figure 2.3 Surface representation by triangular tiles obtained from points on a set of parallel closed contours.

slice. Because of the large number of possible triangles, it is important to keep n_p and m_p as small as possible. Hence these techniques used re-sampled outlines with a limited number of points (Udupa 1983). The triangulation problem (i.e. the choice of one of the many possible sets of triangles) can be formulated in terms of graph theory (Keppel 1975). Finding the 'best' surface is equivalent to traversing a minimum cost path through a directed graph. Costs associated with each path (corresponding to a set of triangular patches forming a closed surface) can be either the volume enclosed (Keppel 1975) or the area of the surface (Fuchs et al 1977). Computationally the method requires $O(m_p.n_p)$ operations, and has the merit of being an entirely objective algorithm.

However, the surface produced may be visually unacceptable, containing sharp discontinuities which the observer finds distracting. Alternative schemes have therefore been proposed which are designed to produce visually acceptable surfaces. Given an initial edge formed by selection of a point in each outline, successive edges are selected using the criterion that the orientation of the next edge should be as similar as possible to the line joining the centroids of the two outlines (Cook et al 1981, Cook et al 1983). This technique is of complexity $O(m_p+n_p)$, and has been applied to the display of cardiac muscle cells (Marino et al 1980), and organs imaged using both CT (Preston et al 1982) and ECAT techniques (Minato et al 1982). A similar algorithm has also been described based on selecting the longer of two candidates for the next edge (Christiansen and Sederberg 1978). If the two contours differ greatly in shape, then operator interaction is required to produce more acceptable tiling of certain segments.

All these tiling algorithms have as output a set of surface

patches. These may be displayed using conventional computer graphics techniques for hidden surface removal and shading (Newman and Sproull 1973). If 'flat' shading of surface patches is acceptable, and a terminal with local intelligence for polygon filling is available, then near real time display of simple medical objects is possible (Fuchs et al 1983). More sophisticated shading algorithms can produce smooth images which disguise the discrete nature of the surface representation (Phong 1975).

Tiling algorithms are best suited to the display of topologically simple organs such as the stomach and liver (Wind et al 1986). For more complex topologies extensive operator interaction may be necessary to guide the algorithm. Nevertheless this remains a popular technique, with applications in electron microscopy (Jimenez et al 1986), and medical imaging using both xray CT (Pate et al 1986) and radionuclide tomograms (Webb et al 1987). A specific use for tiling algorithms has been found in the area of pre-operative surgical planning, using a relatively coarse representation of the facial bones. This facilitates both interfacing to a commercial CAD/CAM system (Cutting et al 1986) and real time manipulation of bony structures (Marsh et al 1986). The technique has recently been applied to the display of isodose surfaces in three dimensional radiotherapy planning, using horizontal 'ribbons' for the isodose surfaces, to show the relationship of the irradiated volume to the underlying target organs (Bauer-Kirpes et al 1987).

The idea of using triangular or other polygonal patches to represent a surface has much to recommend it. The algorithms developed over many man-years for the display of artificial objects such as machine parts and space shuttles can be immediately applied to such a representation, and a correspondingly high degree of realism can be

achieved. However, for natural objects, and particularly for medical objects with complex topologies, the creation of the surface representation can be difficult. There is the additional problem of defining the surface of interest. It may be feasible to specify a fixed threshold for attenuation coefficient when extracting a bony object from a set of CT slices, and to create the corresponding surface model. However, if the observer wishes to examine the object after removal of some overlying layers, then much of the model must be reconstructed. In the case of ECAT data, the choice of threshold is almost always an iterative procedure, since the 'objects' isolated at each stage are iso-surfaces in the radionuclide distribution, whose appearance cannot be readily predicted. There is therefore a requirement for a form of object or surface representation which is capable of interactive alteration, and which can be re-created from the original data without significant delay.

2.4 Voxel based displays

A complete set of transverse sections forms a three dimensional array of voxels, each containing the average value of the imaging parameter in the corresponding volume. Display methods using representations based on such three dimensional arrays have been described as 'cuberille' techniques (Herman and Liu 1979). An obvious attraction of these methods is that even topologically complex objects can be represented by a cuberille array (Herman and Udupa 1981). Several authors have described techniques for the identification and display of surfaces in such an array. These may be summarised as follows:-

Facet lists: Once the object of interest has been segmented from the background, the surface is formed by the set of all surface facets (Herman and Udupa 1981). Each surface facet is the square face of an 'object' voxel which is adjacent to an empty or 'background' voxel. If the surface is constrained to be 'connected', then it can be represented by a directed graph (Artzy et al 1981). Each node in the graph represents a face, with two incoming and two outgoing arcs representing adjacency to four other faces. Given an initial face it is possible to traverse the graph and so to enumerate all connected faces of the object. This algorithm can be generalised to multidimensional data (Udupa et al 1982).

The facets can subsequently be displayed from any angle in the same way as the triangular patches described above. However, the hidden surface problem is much easier to solve for a facet list. Each face falls into one of six groups, depending on its alignment with the major axes of the cubic array. Hence for any viewing position, approximately half the faces may be discarded, since they fall into the three categories of 'back-faces' which point away from the observer. Moreover it can be shown that for cubic voxels, priorities for determining visibility can be resolved by a simple depth sort, based on the distance of the centre of each face from the observer (Herman and Udupa 1983). These techniques have been applied to the display of NMR data (Axel et al 1983, Trivedi et al 1986), and radiotherapy planning (Bloch and Udupa 1983), but their principal application has been to the display of CT data. Specific areas of interest include the temperomandibular joint (Roberts et al 1984, Kursunoglu et al 1986), acetabular fractures (Burk et al 1985), measurement of shape and volume (Udupa 1981, DuFresne et al 1987), and congenital and traumatic facial abnormalities (Altman et al

1986, DeMarino et al 1986, Gillespie et al 1987).

Solid contours: Each slice in the three dimensional array can be thresholded to produce a binary image, in which bits set to one correspond to voxels which are part of the object. This image in turn can be represented by a 'directed contour', enumerating the voxels in that slice which lie on the outline of the object (Tuy and Udupa 1982). The contour can be represented by the coordinates of the start point, with a chain code to indicate the location of succeeding points. Because each contour has a direction associated with it, all points in the section can be unambiguously assigned to the interior or exterior of the contour in that plane. The contour can be described as 'solid' because it specifies a series of voxels lying in a slice of finite thickness, unlike the line outlines described in Section 2.3. A stack of adjacent directed contours therefore represents the surface without the need for tiling. This form of object representation facilitates operator interaction and segmentation (Chen et al 1985a), whilst still enabling a full cuberille representation to be recreated for display purposes (Udupa 1982).

A similar technique has been described for displaying the surface of the heart (Heffernan and Robb 1985). In this case the chain code was further represented by a Fourier series to obtain a smoother outline. Although this produces smooth contours, the technique restricts object rotation to the single axis perpendicular to the plane of the sections, and surface shading is similarly determined by the orientation within each slice, rather than between slices. Nevertheless, a considerable increase in display speed can be achieved using these representations rather than the full cuberille model.

Octrees: If a three dimensional volume contains a set of objects which are reasonably homogeneous, then the recursive structure known as an octree may be used to represent the volume (Jackins and Tanimato 1980). The original cube is divided into eight sub-cubes of half the linear dimension. If a sub-cube is homogeneous (i.e. contains either no object or entirely the same object value) then that portion of the tree terminates. Non-homogeneous sub-cubes are recursively sub-divided into eight smaller cubes, and so on. The process continues until all branches of the tree have terminated, or until the volume of the remaining sub-cubes has fallen below some pre-determined threshold. Once complete the octree forms a compact representation of three dimensional shape, and because of its recursive structure it can be rapidly traversed to produce hidden surface images without sorting (Doctor and Torborg 1981).

Application to medical data is restricted by the fact that the original volume is usually highly inhomogeneous, containing many different object values. Once it has been segmented into object and background, then the octree can be constructed (Meagher 1984). Using special purpose hardware, objects represented by octrees can be rotated and displayed in real time. However, the time to create the octree representation is a significant problem, requiring 6 to 8 hours for a complex skull model (Salyer et al 1986).

This representation differs from those described above in that the object is represented as a solid, rather than as a set of surfaces. The recursive form of representation facilitates display, but is less suited to operator interaction, e.g. to remove overlying objects.

Raw Data: The idea of representing the solid object rather than just its surface can be taken to a logical extreme by using the original set

of tomographic sections. Such schemes have the advantage that no object manipulation is required other than segmentation, i.e. the identification of the object and its separation from the background. Although the entire object is represented, only the surfaces are displayed. This is a particularly useful technique when the observer is likely to require several attempts at object segmentation.

For use with the Raw Data representation two broad categories of display algorithms can be identified; those which are based in the object space and 'project' voxels onto the display screen, and those which are based in the image space and 'interrogate' the object along ray paths. The former group exploit the intrinsic depth ordering of a voxel array to solve the hidden surface problem. Successive planes of the cubic array can be offset to produce images which are reasonable approximations to oblique views at small angles of rotation (Farrell 1983). The planes are displayed in sequence from back to front, with successive frames offset left or right, up or down, to form an isometric image. This technique has been applied to the display of cranial CT scans (Farrell et al 1984), and has several advantages. Oblique views are generated simply by orthogonal shifting of successive planes, without the need for trigonometrical calculations. By combining horizontal and vertical shifts, and by reversing the order in which planes are processed, up to 18 different isometric views may be produced. However, isometric displays are only a reasonable approximation to true three dimensional transformations for small angles of obliquity. In practice angles of less than 20^{o} must be used to avoid noticeable distortion. This algorithm exploits the inherent priority order of a set of coronal sections, and no sorting stage is required. It can be implemented using the raw data, in which case each slice must

be segmented as it is encountered, or it can be implemented using a set of sections which have already been segmented. The choice between these two schemes will depend on whether the observer needs to try several segmentation thresholds or prefers to examine a single object from several directions.

More generally, the raw data array may be traversed in a back to front sequence to display the object from any aspect (Frieder et al 1985). As each voxel is encountered it is transformed into observer coordinates and displayed in the image array. Nearer voxels are encountered later in the sequence and hence overwrite voxels which are hidden. Dedicated hardware can be used to achieve rapid display rates using this technique (Goldwasser et al 1985), and there are no restrictions on the angles of view which can be obtained.

In the second group of display algorithms the pixels in the output image act as sources of interrogating 'rays' which pass into the object array. When the first voxel is encountered which is considered to be part of the object, the distance travelled along the ray is stored in the corresponding pixel of the image array. The output is therefore a depth coded image, containing the distance to the voxel visible at that point. This technique has been described for the display of range-echo data obtained using ultrasonography (Nakamura 1984), and electron microscopy data in the form of successive sections (Radermacher and Frank 1984).

When rays are travelling at angles which are oblique to the major axes of the voxel array, care must be taken to ensure that the increment by which the rays are extended is sufficiently fine to intersect with all voxels along the path (Tuy and Tuy 1984). If too large a step is

used then small objects may be missed, while too small a step results in excessive delay. The largest number of applications of techniques of this form have been in the field of CT scanning (Marsh and Vannier 1983a). Specific areas of interest include complex musculo-skeletal anatomy (Totty and Vannier 1984), craniofacial disorders (Marsh and Vannier 1983b) and surgical planning (Vannier and Marsh 1983).

2.5 Conclusions

Techniques which display an object using shaded surfaces can be implemented on a conventional medical imaging system without requiring additional hardware. Of the available methods, cuberille schemes offer the best approach. This object representation has a natural relationship with the input data, enables both simple and complex objects to be represented, and can be displayed using simple, rapid hidden surface algorithms. Cuberille schemes are relatively poor at representing smooth objects, and require special consideration when shading the visible surfaces. However, given the convoluted structure of many body organs, and the fact that three dimensional displays are of most value when the morphology is complex, the advantages of the cuberille schemes were considered to outweigh their disadvantages, and they were selected as the basis for further investigation.

For routine use in the clinical environment a three dimensional display technique should have the following properties:-

a) It must be capable of using tomographic data acquired without significant changes to existing procedures, and be able to convert that to a three dimensional representation in a reasonably short time without the need for large amounts of operator interaction.

b) The hidden surface algorithm should execute rapidly, allowing the observer to display object surfaces from any aspect using both simple and sophisticated shading algorithms.

c) The form of object representation used should facilitate interactive manipulation, either to investigate alternative segmentation schemes or to remove parts of the object.

d) The technique should be able to use data from all imaging modalities and be capable of displaying multiple objects simultaneously.

The techniques developed for this work are an attempt to satisfy these criteria using readily available computer systems.

3. Object Representation

3.1 Input Data

Tomographic imaging techniques produce two dimensional digital images representing transverse slices through the body. Each slice is of finite thickness, and a set of parallel contiguous slices can be considered as a three dimensional digital array, sampling the whole of the volume of interest. This array forms the input data for three dimensional display. Surface display algorithms are greatly facilitated if the original array has equal sampling intervals along all three directions. It is common to pre-process the data, interpolating additional slices where necessary, to obtain a cubic array. Cubic arrays can be readily obtained from sets of radionuclide tomograms obtained using a rotating gamma camera, because the slice spacing will always be an integral multiple of the pixel size. For other modalities the slice spacing may have no simple relation with the pixel size, but linear or polynomial interpolation can still be used to re-sample the data at equal intervals.

If several sets of tomographic sections are reconstructed through the same object at different times, then the input data is a four dimensional array showing the time variation of the imaging parameter throughout the sampled volume. It will still be desirable to have equal spacing along each of the three spatial axes, but sampling along the time axis will usually be relatively coarse. This may consist of two points only, representing the object before and after medical intervention, or cardiac end-diastole and end systole.

3.2 Segmentation

A three dimensional array of voxels contains information about the imaging parameter throughout the reconstructed volume, including both the object(s) of interest and the surrounding tissues. In the natural world objects are perceived by means of light reflected from their surfaces, and three dimensional surface display techniques simulate this process. To do so it is first necessary to detect and locate the surfaces of the objects of interest in the reconstructed volume. This problem is analogous to the segmentation problem in two dimensional image processing, namely the division of an array of pixels into those which are part of the object, and those which are not (Gonzalez and Wintz 1977). Even if fully automated segmentation schemes were available, this process must retain a subjective element, since only the observer can specify which objects will be of interest in a given clinical situation. In practice segmentation schemes are usually highly interactive.

The simplest algorithm uses grey scale thresholding of individual voxels (Marsh and Vannier 1983a) with no spatial restrictions and no consideration of adjacent voxels. A voxel is considered to be part of the object if the value of the imaging parameter (the 'grey scale value') lies above, below or between selected thresholds. This approach is only suitable for isolating objects which have grey scale values markedly different from all other voxels in the scene, and where the random variation in grey scale value for the same tissue is small. In practice, especially for radionuclide tomograms, neither of these requirements are satisfied. If the object of interest has grey scale values similar to those of another tissue or object, then interactive

spatial bounding must be used to isolate only the organ required. Such limits may be drawn as a bounding region of interest on each tomogram making up the reconstructed volume. Alternatively, if the object of interest forms a spatially isolated group then a region growing technique may be used. The observer specifies a threshold range, and indicates a seed voxel which is in this range, and which is part of the object of interest. The algorithm selects all voxels which have grey scale values in the correct range, and which are connected via a chain of such voxels to the original seed (Udupa 1982). This algorithm has the merit of requiring little intervention on the part of the operator, and the boundaries of the objects generated are entirely objective. However, determining connectivity for all voxels in the array can be very time consuming, since it must be performed using a three dimensional search algorithm. Let OBJECT be an array in which the point (x,y,z) is a seed voxel. The following recursive algorithm can be used to list all voxels which are connected by face to face adjacency :-

Algorithm 3.1

```
procedure CONNECT(x,y,z)
  if OBJECT(x,y,x) is a background voxel then EXIT(CONNECT)
  record (x,y,z) as an object voxel
  set OBJECT(x,y,z) to a background value
  CONNECT(x-1,y,z)
  CONNECT(x+1,y,z)
  CONNECT(x,y-1,z)
  CONNECT(x,y+1,z)
  CONNECT(x,y,z-1)
  CONNECT(x,y,z+1)
end of procedure
```

Algorithms which use two dimensional searches to determine connectivity within each slice separately are considerably faster, since they make fewer recursive calls and only need to access a single slice at any time. However, complex medical objects may be continuous in three dimensions but discontinuous when considered as a stack of sections. A two dimensional recursive search may therefore be insufficent to locate all parts of the object.

Random variations in grey scale value give rise to irregular organ boundaries, with isolated voxels appearing outside the organ, and holes or cavities appearing within it. This effect can be avoided by segmenting the tomographic data after low pass filtering. Averaging grey scale values within and between slices reduces the random variation, and subsequent thresholding yields smoother surfaces. There is considerable experimental evidence to suggest that this procedure can give accurate organ boundaries, since the volumes of objects estimated using this technique correlate closely with the true volume for a wide range of organ sizes (Tauxe et al 1982). Providing that the object is large relative to the pixel size, then low pass filtering is unlikely to affect the apparent size of the object. In this work, low pass filtering was accomplished by a convolution of the original data with a 9 point smoothing kernel, using the following pattern of weights :-

0.0625	0.125	0.0625
0.125	0.25	0.125
0.0625	0.125	0.0625

Smoothing using this pattern can be implemented as a rapid procedure by successive addition of adjacent rows and columns. For particularly noisy

data, several passes of the convolution kernel were used. This kernel can be approximated by a Gaussian function with a variance of 0.5 pixels^2. When two functions with variances V_1 and V_2 are convolved together, the resulting function has variance V_1+V_2, and the 'width' of the function can be considered as proportional to the square root of the variance. Hence a single pass of the smoothing function described above increases the 'width' of an object by a factor given approximately by :-

$$F_W = [\ 1 + (\ 0.5 / V_O\ )\ ]^{\frac{1}{2}} \qquad 3.1$$

where V_O is the original object variance. The actual variance depends on the shape of the object, but for a 'top hat' function with a width of 12 pixels, the variance is also 12 pixels^2, and the corresponding factor is only 1.02. Even several passes of the convolution kernel will only increase the object width by a few percent, and this can probably be compensated for by an increased threshold value. For smaller objects the factor F_W increases, but is still only 1.11 for a top hat function with a width of 5 pixels. Most objects of interest will be larger than this, with correspondingly smaller distortion factors.

A more serious consequence of low pass filtering is the loss of contrast between adjacent tissues or abnormalities which have different grey scale values. Only when the object to background (or normal to pathological) ratio is reasonably high can extensive low pass filtering be used. In all cases presented in this work the smoothed sections were examined individually for evidence that the filtering had obscured genuine differences.

Figure 3.1 shows a transverse section through the cardiac blood pool, obtained using the technique of radionuclide tomography. It is

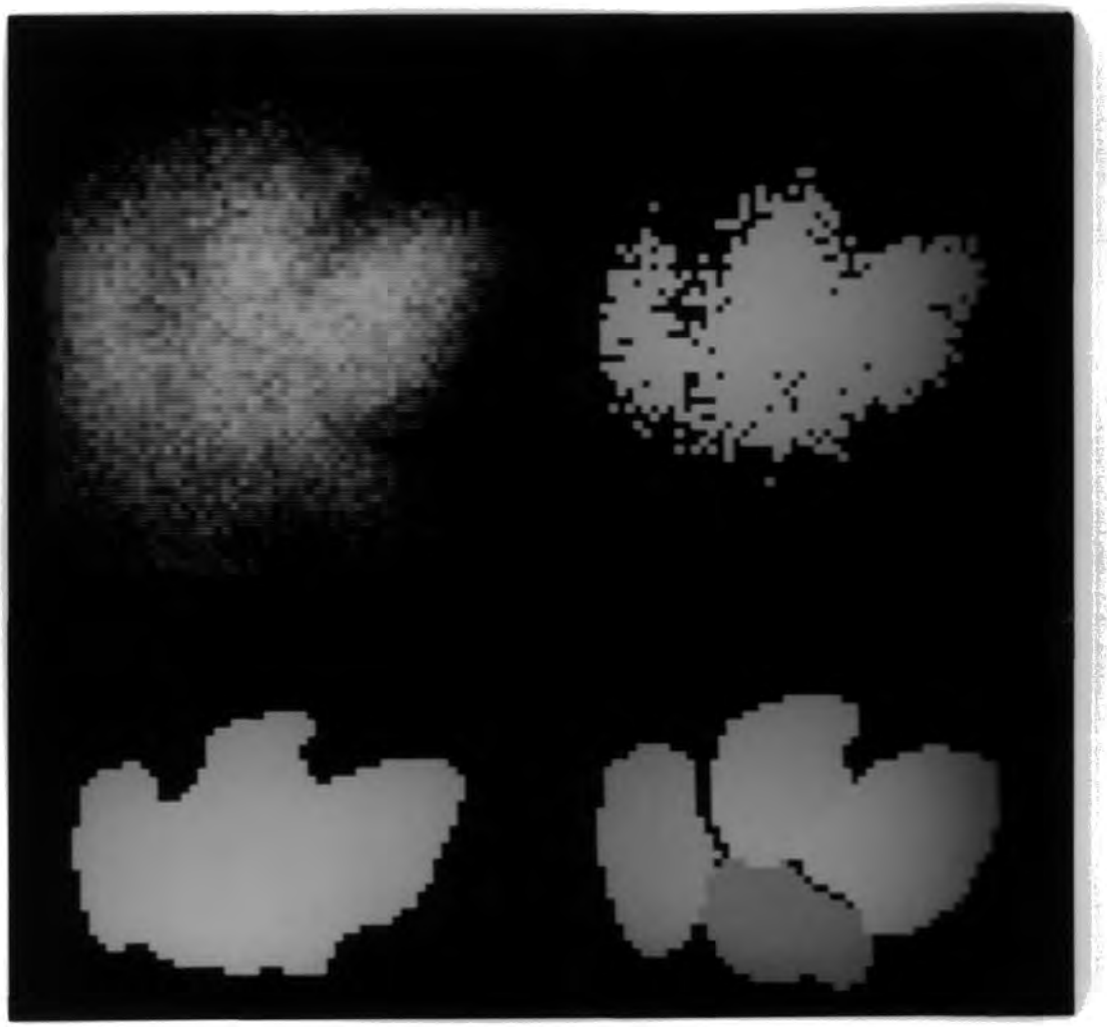


Figure 3.1 Segmentation of ECAT data to isolate the cardiac blood pool :-

Upper left - The raw data.
Upper right - Simple thresholding, showing irregular outline.
Lower left - Thresholding after low pass filtering, giving a much smoother outline.
Lower right - Further interactive segmentation to identify individual chambers.

thresholded without low pass filtering, showing the effect of random noise on the object surface. The object obtained after filtering corresponds more precisely to the true cardiac outline, and may be used as input to a further interactive segmentation algorithm. Here the four cardiac chambers have been identified using regions of interest drawn by the observer. Multidimensional classification procedures can also be used to improve the specificity with which the object is isolated. Regions satisfying criteria based on both grey scale and local gradient have been used to isolate areas of infarcted myocardium from both normal tissue and surrounding air (Trivedi et al 1986). Such sophisticated segmentation procedures are time consuming, and they must be performed as a pre-processing stage with subsequent rapid display of the selected object. For all the results presented in this work, segmentation was performed using low pass filtering and thresholding. Masking was used only infrequently, usually to exclude unwanted areas outside the area of interest, and was performed using outlines drawn manually on individual sections. The image processing system was provided with a trackerball for interactive image manipulation. Outlines were drawn rather larger than the cross-section of the object of interest. The object was then segmented using thresholding, but only pixels inside the marked outlines were included. This approach combined the flexibility of manual segmentation with the objectivity of a thresholding technique.

The process of segmentation results in a loss of information, since the original multi-valued data has been converted into a two-valued or binary form, indicating only that the object is or is not present at each point. However, unambiguous definition of the object is an essential pre-requisite of shaded surface display. In the case of CT and NMR data, where the difference between grey scale values is mainly due

to the anatomy of the corresponding tissues, a set of segmented data can be considered as a representation of an anatomical structure. For ECAT data, however, the difference between grey scale values indicates the different levels of uptake of radionuclide in adjacent tissues, and is therefore indicative of function rather than structure. The segmented data must now be thought of as indicating the spatial extent of tissues whose function, as measured by uptake of tracer, is above the chosen threshold. Selection of different thresholds will often be required, to allow the observer to study several isocount surfaces.

3.3 Test Objects

After segmentation, the object can be represented in a form which may be more compact, or which facilitates three dimensional display. Data storage requirements for several object representation schemes were evaluated using eight test objects. All the objects were sampled at 64 x 64 x 64 resolution, and they were segmented using low pass filtering followed by thresholding. The same segmentation procedure was followed for each object representation. All the objects except the Cube were obtained by ECAT techniques, using a rotating gamma camera. The objects were (Figure 3.2) :-

Spiral : A test object formed by bending a tube containing Tc-99m around a cylinder.

Tumour : Uptake of Ga-67 in three areas of the chest, indicating bronchogenic carcinoma.

Myocardium : Perfused myocardial tissue imaged using Tl-201.

Liver : Uptake of Tc-99m tin colloid in the reticuloendothelial tissues of the liver and spleen.

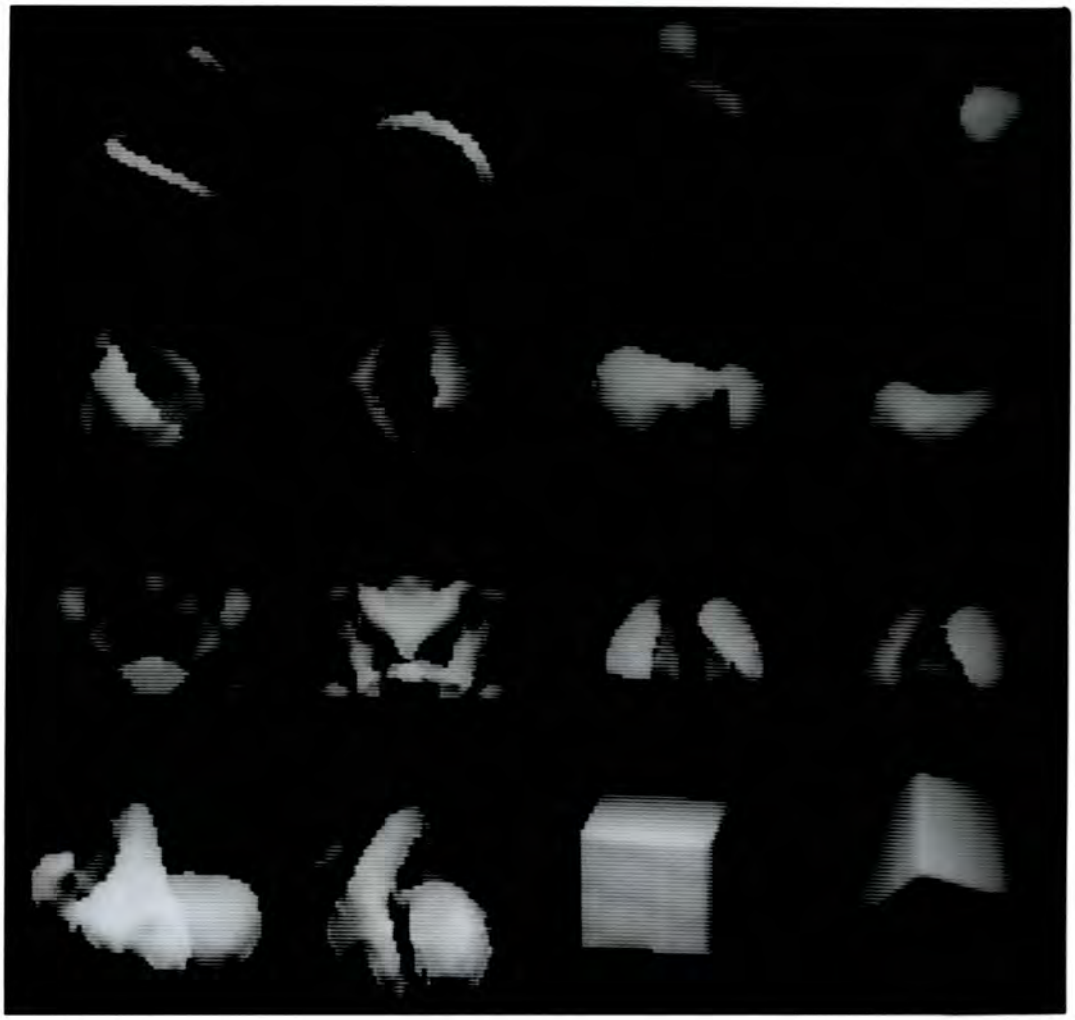


Figure 3.2 Eight test objects (Spiral, Tumour, Myocardium, Liver, Pelvis, Lungs, Heart and Cube) arranged in increasing order of size. Each object is shown from two aspects.

Pelvis : Uptake of Tc-99m methylene diphosphonate in the bones of the pelvis, and in the bladder.

Lung : Perfused lung tissue imaged using Tc-99m macro-aggregated human albumin.

Heart : The cardiac blood pool, imaged after in vivo labelling of red blood cells using Tc-99m.

Cube : A mathematical test object.

These test objects are arranged in increasing order of size (after thresholding), which ranged from 1100 voxels for the Spiral to 29791 voxels for the Cube. They included both simple (Cube) and complex (Pelvis) topologies, as well as both rounded (Liver) and irregular (Heart) shapes.

The amount of data storage required by different object representations has important implications for subsequent display. The amount of data to be processed will affect the display rate which can be achieved, as will the necessity for disk access if the data storage requirement is large. The variability of storage requirements is also important, since algorithms must be capable of displaying objects with a wide range of sizes and shapes.

3.4 Surface Patches

As described in Section 2.3, object representation by means of polygonal surface patches is intuitively reasonable, and corresponds closely to conventional display programs used for computer aided design. Although this form of representation was not used in this work, it

represents a baseline for comparison with other methods. The data storage requirements for a surface patch representation depend on both the size and shape of the object, and on the size of the polygons used. Small tiles will give a more accurate representation, but more of them will be required to cover a given object.

Each of the eight test objects was segmented, and the tiling requirements were predicted using a perimeter finding algorithm. This operated on each slice separately to determine the number of voxels in that slice which formed closed outlines. It was assumed that tiles would be created using every other point in each outline, i.e. that an outline containing n_p points would generate $n_p/2$ tiles between it and the adjacent superior contour, and a further $n_p/2$ tiles between it and the adjacent inferior contour. Each vertex of a triangular patch requires 3 bytes to store the (xyz) coordinates (assuming that the sampling resolution does not exceed 256 x 256 x 256 voxels), hence 9 bytes are required per face.

Figure 3.3 shows the total storage requirements for the test objects, which ranged from 5.7 to 38.3 Kbytes, compared with 256 Kbytes for the raw data. Considerable data compression is achieved by the conversion from multi-valued data representing a solid object to purely geometric data representing only the surface. As expected, larger objects require more storage, as do the more complex objects (e.g. the Myocardium, Pelvis and Heart). Also shown in Figure 3.3 is the storage requirement expressed as bytes per voxel after segmentation, which shows a clear trend downwards as the object size increases. This is also to be expected, since the object volume increases as the cube of the linear dimension, while the surface area increases only as the square.

The results obtained also indicate the storage requirements for

SURFACE PATCHES

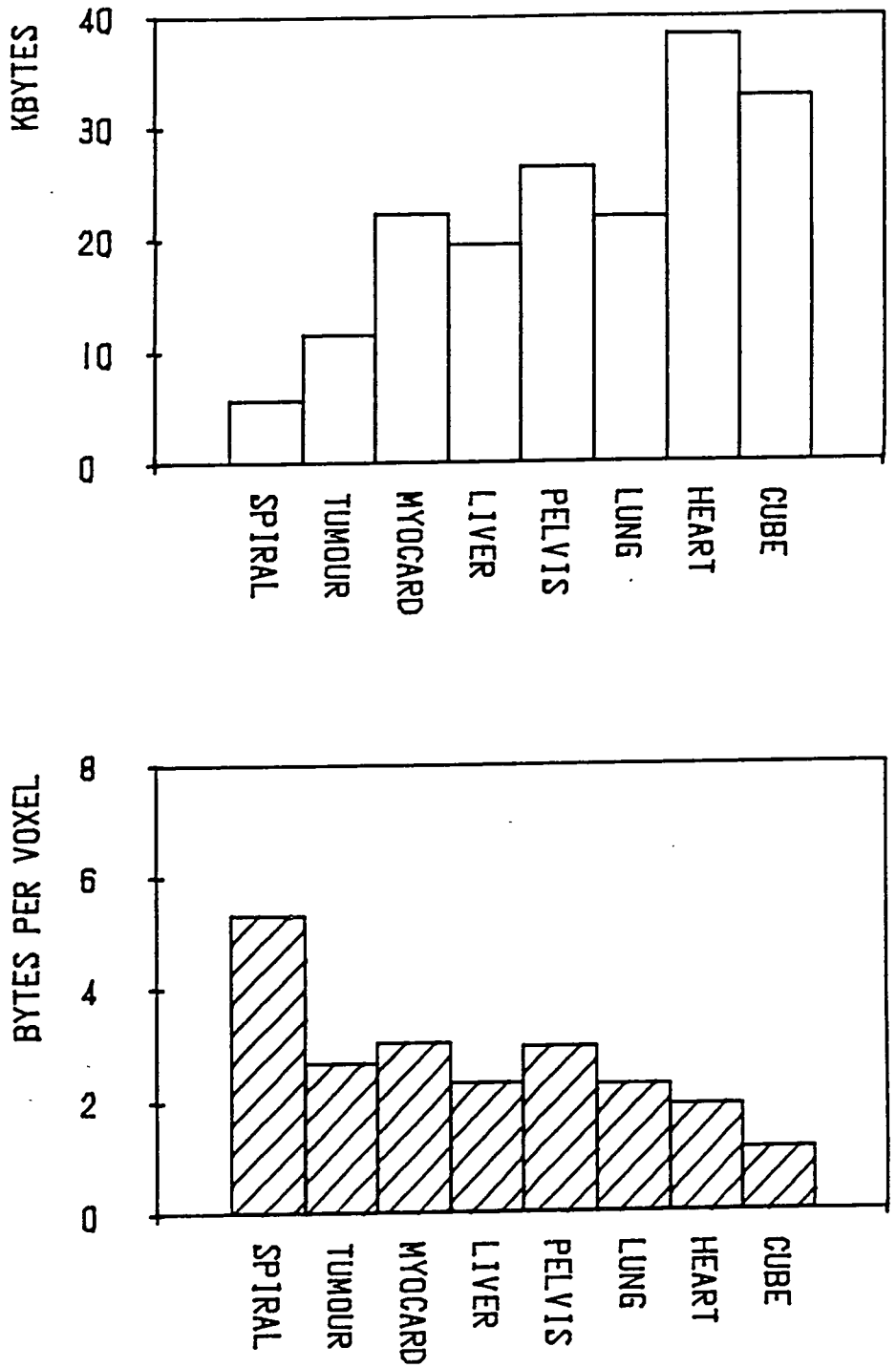


Figure 3.3 Storage requirements for the eight test objects using the Surface Patch representation scheme :-

Upper - Total storage.
Lower - Storage per voxel.

objects sampled at higher resolution. The raw data array (size N x N x N) voxels) requires data storage to increase $O(N^3)$ for finer sampling, whereas surface patch requirements for objects of similar shape should only increase $O(N^2)$, and the data compression obtained using tiling representations will therefore increase as N increases.

Tiling representations are best suited to creating surfaces between two closed contours with approximately equal numbers of voxels, and have difficulty in representing objects with complex topology. Even for the simple case of a bifurcation, where an object appears as a single outline in one section and two outlines in the next section, empirical or interactive methods are required. Similar difficulties arise when objects contain interior voids, and when thin objects are aligned almost parallel to the plane of the sections. An algorithm similar to that used to determine connectivity was used to count the number of closed contours in successive slices, using a two dimensional object array :-

Algorithm 3.2

```
COUNT=0
for z=0 to N-1
  read slice(z) into array OBJECT
  for y=0 to N-1
    for x=0 to N-1
      if OBJECT(x,y) is part of the object then
        COUNT=COUNT+1
        ERASE(x,y)
    next x
  next y
next z
```

This algorithm makes use of a recursive procedure for erasing closed contours which is similar in form to Algorithm 3.1, but which operates within a single slice only :-

Algorithm 3.3

```
procedure ERASE(x,y)
  if OBJECT(x,y) is a background voxel then EXIT(ERASE)
  set OBJECT(x,y) to a background value
  ERASE(x-1,y)
  ERASE(x+1,y)
  ERASE(x,y-1)
  ERASE(x,y+1)
end of procedure
```

Algorithm 3.2 searches each slice for an object voxel, then erases all connected voxels in that plane. It can be used either to count the total number of closed contours, or to predict the presence of tiling discontinuities. The number of such discontinuities can be defined as the sum of the absolute value of any differences between the number of closed contours in adjacent slices. Table 3.1 shows the number of discontinuities found in the test objects. Such topological complexities occur frequently in irregular organs (e.g. the bones of the skull), but are also present at the boundaries of simpler organs (e.g. the lobulated inferior surface of the liver). The tiling approach is unlikely to be capable of representing all medical objects, and requires significant operator interaction even for relatively smooth organs.

3.5 Cuberille Representations

Modelling a three dimensional object as the union of many small cubes is an extremely flexible approach, and is capable of representing

Table 3.1 Number of tiling discontinuities in the test objects

Object	No. of Voxels	No. of Slices containing at least one voxel	No. of Tiling Discontinuities
Spiral	1100	32	2
Tumour	4424	30	12
Myocardium	7515	35	12
Liver	8685	27	6
Pelvis	9191	29	24
Lung	9931	29	6
Heart	20664	36	14
Cube	29791	31	2

objects having simple or complex topology with equal facility. Moreover the model has a direct relationship with the input data, since that also forms a three dimensional array of cubic (or at least cuboid) voxels. Different approaches using cuberille representations differ in respect of the display algorithms used, whether to threshold the object before or during the display stage, and whether to represent the entire organ or just the surface voxels. The following techniques were compared :-

Facet List: If only the surface of a segmented object is of interest then a list of all faces of surface voxels may be created prior to display (Herman and Liu 1979). The boundary surface of the object is defined as a list of connected faces separating object voxels from background voxels. However, there is no intrinsic ordering to the list, and a depth buffer or similar hidden surface algorithm must be used to display the object. The length of the list depends on both the size and complexity of the object, since a highly convoluted shape will have many surface elements. The test objects shown in Figure 3.2 have widely differing numbers of surface voxels, each of which may contribute from one to six faces to the list. Each face is defined by four bytes, three specifying the (xyz) coordinates of the centre and the fourth to indicate the orientation of the face, of which there are only six possibilities. Total data storage required for this representation is shown in Figure 3.4, and ranges from 7.6 to 36.7 Kbytes. As for the Surface Patch representation, the three irregular objects (Myocardium, Pelvis and Heart) require relatively more storage space, reflecting their larger surface area. Since this is a surface representation the data storage requirement should only increase $O(N^2)$ for larger arrays. This is confirmed by the results in Figure 3.4, which show a fall in the

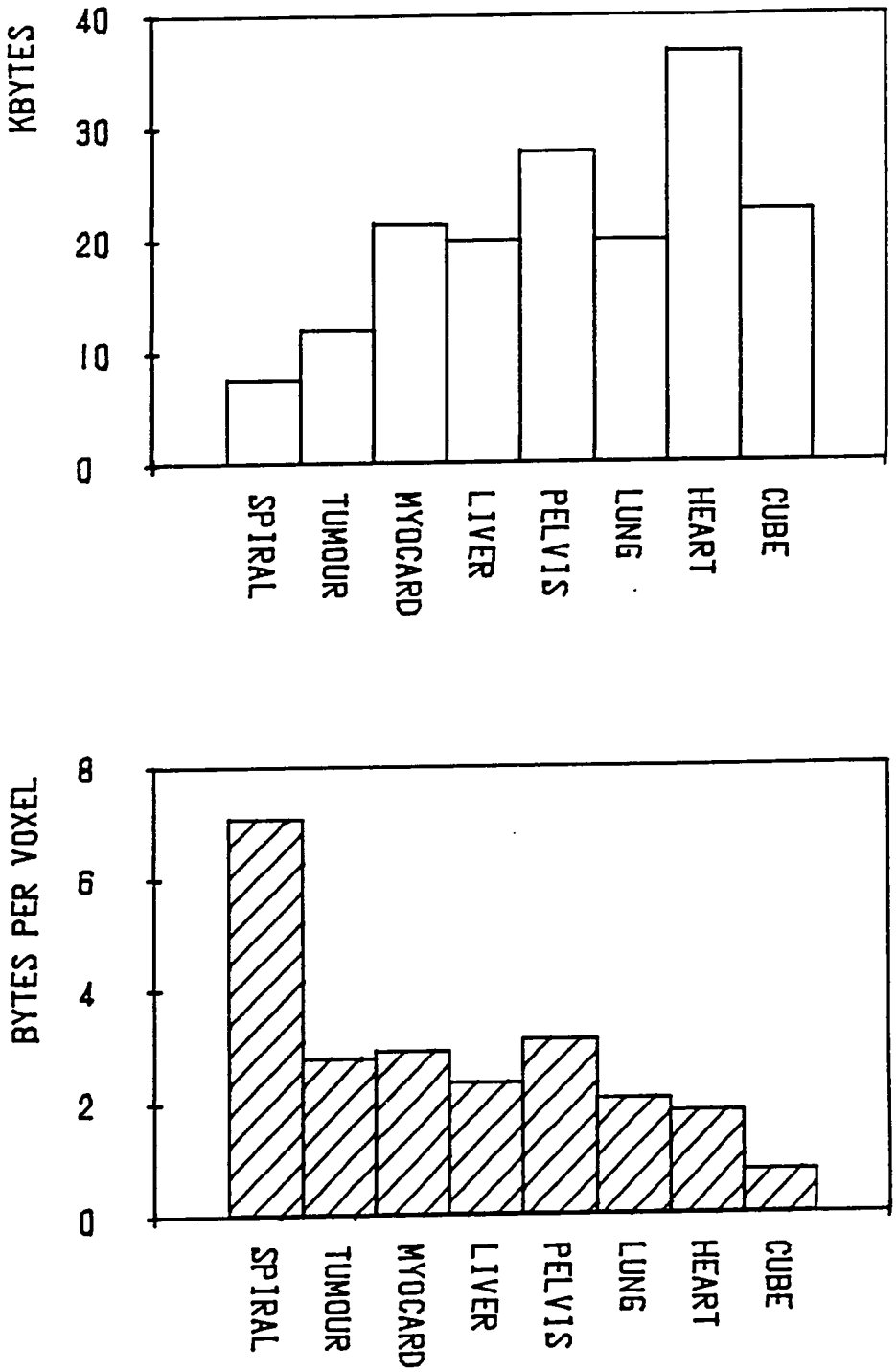


Figure 3.4 Storage requirements for the eight test objects using the Facet List representation scheme :-

Upper - Total storage.
Lower - Storage per voxel.

storage requirement per object voxel as the object increases in size.

The list is created after segmentation, and so object manipulation is difficult using this representation. To cut away part of the object requires the creation of an entirely new Facet List before the object can be redisplayed.

Binary Object: The criteria identified in Section 2.5 for an ideal display technique included the ability to manipulate the object after segmentation to show interior structures. For this work a cuberille representation was devised which has this property. Each voxel is either part of the object or part of the background, and so may be represented by a single bit in a binary array. A segmented array of N x N x N voxels can therefore always be represented by $N^3/8$ bytes, irrespective of the size or complexity of the object to be represented. If N is reasonably small (up to 64 in this study) then the entire array may be stored in main memory with subsequent rapid display (Gibson 1983). This scheme does not allow the observer to vary the segmentation threshold during display, but compensates for this restriction by enabling the observer to interactively explore the interior of the chosen object. Representation using a relatively coarse array is particularly suitable for the display of ECAT data, where a sample size of 64 x 64 x 64 voxels is standard. Perhaps surprisingly, however, acceptable results were also obtained using CT images sampled at higher resolution.

Once a binary object array has been created, simple and extremely rapid 'slicing' techniques may be used to reveal interior details. An array of 64 x 64 x 64 bits requires only 32 Kbytes of storage, and so the whole of the segmented object was represented in main memory. Logical operators (AND, OR and NOT) may also be used to modify or

combine two binary objects. Figure 3.5 shows a liver and spleen with increasing amounts of tissue removed to reveal a cavity in the right lobe of the liver. This was accomplished by setting all bits to zero which were closer to the observer than some chosen value. The binary object array was arranged in memory in a form which facilitated this mode of interaction. The 64 voxels forming a line through the cube parallel to the viewing direction (i.e. directly away from the observer) were always represented by four 16-bit words of memory. Proximal voxels in that line were erased by setting 0,1,2 or 3 words to zero (corresponding to 16 voxels at once), and performing a logical AND of the next word with a simple mask to zero the remaining bits. Using assembly language coding the time required for such object manipulation was typically only 0.1 s (Gibson 1986a).

The representation of CT data sampled at 256 x 256 resolution within each slice required additional pre-processing. Either subsections were extracted showing only the area of interest, or the slices were compressed by 'averaging' adjacent pixels. To avoid blurring the edges of objects, the value chosen to represent the averaged pixels in the compressed data could be either the arithmetic mean of the pixel values, or the maximum value. The latter was particularly useful for the subsequent segmentation of bony objects, since the contrast betwen bone and soft tissue was retained. Figure 3.6 shows shaded surface images of a thorax, derived from CT sections thresholded to select out all tissues with density greater than the lungs. The object was rotated to show the patient supine and viewed from the feet, then modified as described above before being rotated back to show an oblique aspect. As increasing amounts of the binary object are sliced away the lung cavity and the mediastinum are revealed.

Figure 3.5 A liver and spleen represented by a Binary Object obtained from ECAT data :-

Upper left - Anterior view of original object.
Upper right - Part of the object removed to show a cavity in the right lobe of the liver.
Lower left - The same defect after more of the object has been removed.
Lower right - The same object seen from the right anterior oblique aspect.

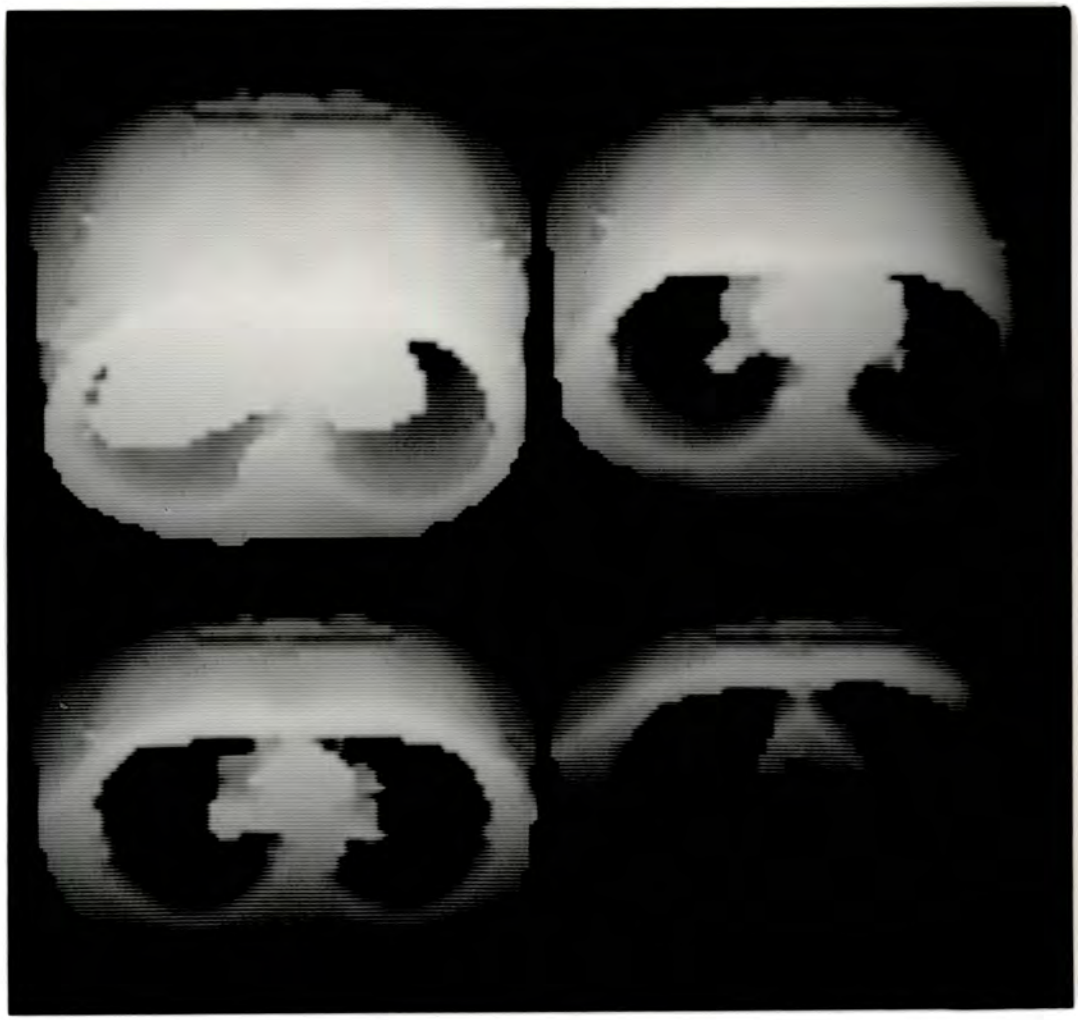


Figure 3.6 A thorax represented by a Binary Object obtained from CT data. All the images show the anterior and inferior aspects, with increasing amounts removed to reveal the heart and mediastinum.

Figure 3.7 shows the same object after processing with the logical NOT operator, transforming the solid thorax with lung cavities into solid lungs. The new object can then be rotated and tilted (see Section 4.4) to display the lungs from any aspect, showing the cardiac cavity, the concave inferior surfaces and the rib impressions on the lateral surfaces. Despite the fact that the data was compressed from 256 x 256 voxels in each slice to only 64 x 64 voxels, the detail in the image is excellent.

Raw Data: The simplest cuberille scheme uses the raw data itself, which forms (after interpolation) a cubic array. The entire object is represented, hence data storage for an array of N x N x N voxels will always be N^3 bytes, assuming that one byte is sufficient to represent the grey scale content of each voxel. Segmentation must be performed during the display stage, and so only simple algorithms will be sufficiently rapid. However, the ordered enumeration of voxels in the three dimensional object array means that a rapid display algorithm can be used, which exploits the intrinsic depth priority of this representation scheme. Display algorithms for all object representations are discussed in more detail in Chapter 4.

Ordered Surface List: The raw data forms a set which is too large to fit into main memory using standard image processing systems, and display is correspondingly slow because of the need for disk access. A new surface representation was devised for this work which uses eight lists of all the voxels (rather than their faces) which lie on the boundary between the object and the background (Gibson 1986b). Each list is created by traversing the original object array in a unique manner, so that subsequent display can be performed using the simple and rapid hidden

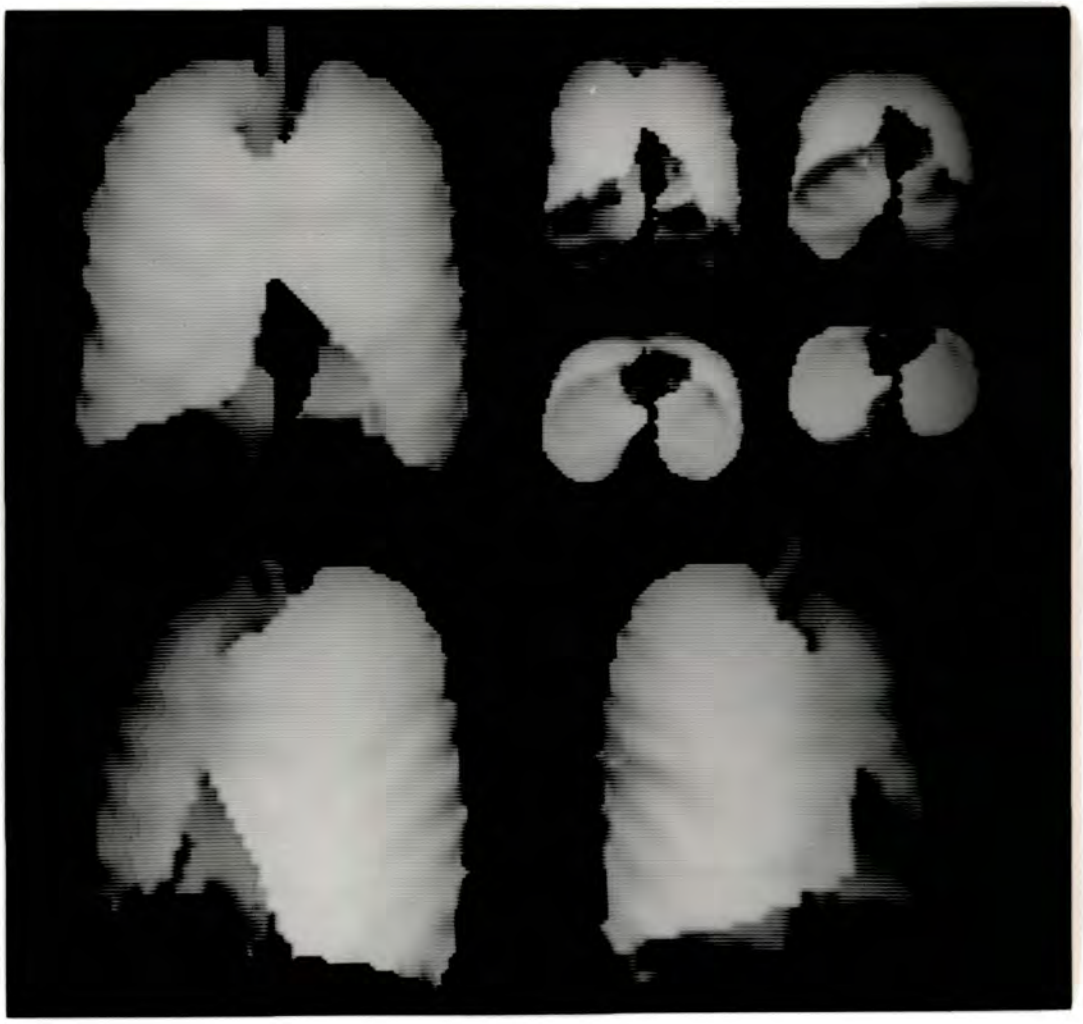


Figure 3.7 Lungs extracted from the thorax shown in Figure 3.6 by using a logical NOT operator :-

Upper left - Anterior view, showing the cardiac impression.
Upper right - Anterior and Inferior aspects, showing the concave inferior surfaces of the lungs.
Lower left - Left anterior oblique, showing rib impressions.
Lower right - Right anterior oblique.

surface algorithm described in Section 4.5. Each voxel can be represented by three bytes specifying the (xyz) coordinates of the centre of the voxel, and the storage requirements for the test objects are shown in Figure 3.8, at the same scale as Figures 3.3 and 3.4. Note that for any given observer position only one of the eight lists needs to be considered, since that list will contain all the voxels which can ever be seen from that position. The values plotted in Figure 3.8 are therefore the average list lengths for each object, rather than the total lengths of the eight lists.

This technique achieves a greater degree of data compression than other cuberille representations, whilst retaining all the flexibility of the cuberille model. Because the list length is reasonably small, the current list can be stored in main memory with correspondingly rapid display. List lengths ranged from 617 voxels to 3668 voxels, with a maximum coefficient of variation for any individual object of 6.8 %. The small coefficient of variation for list sizes implies that objects are of similar complexity on all sides, and hence that the list length will be reasonably constant for all viewing positions. Because they are a surface representation Ordered Surface Lists are expected to increase $O(N^2)$ for larger matrix sizes, and this is confirmed by the results in Figure 3.8, which show a fall in the storage requirement per voxel as the object size increases.

Although this is a 'surface only' form of representation, the generation of Ordered Surface Lists from the raw data is a simple procedure. This form of object representation is therefore a good compromise between the flexibility of a solid object scheme for interactive manipulation, and the simplicity of a surface scheme for display.

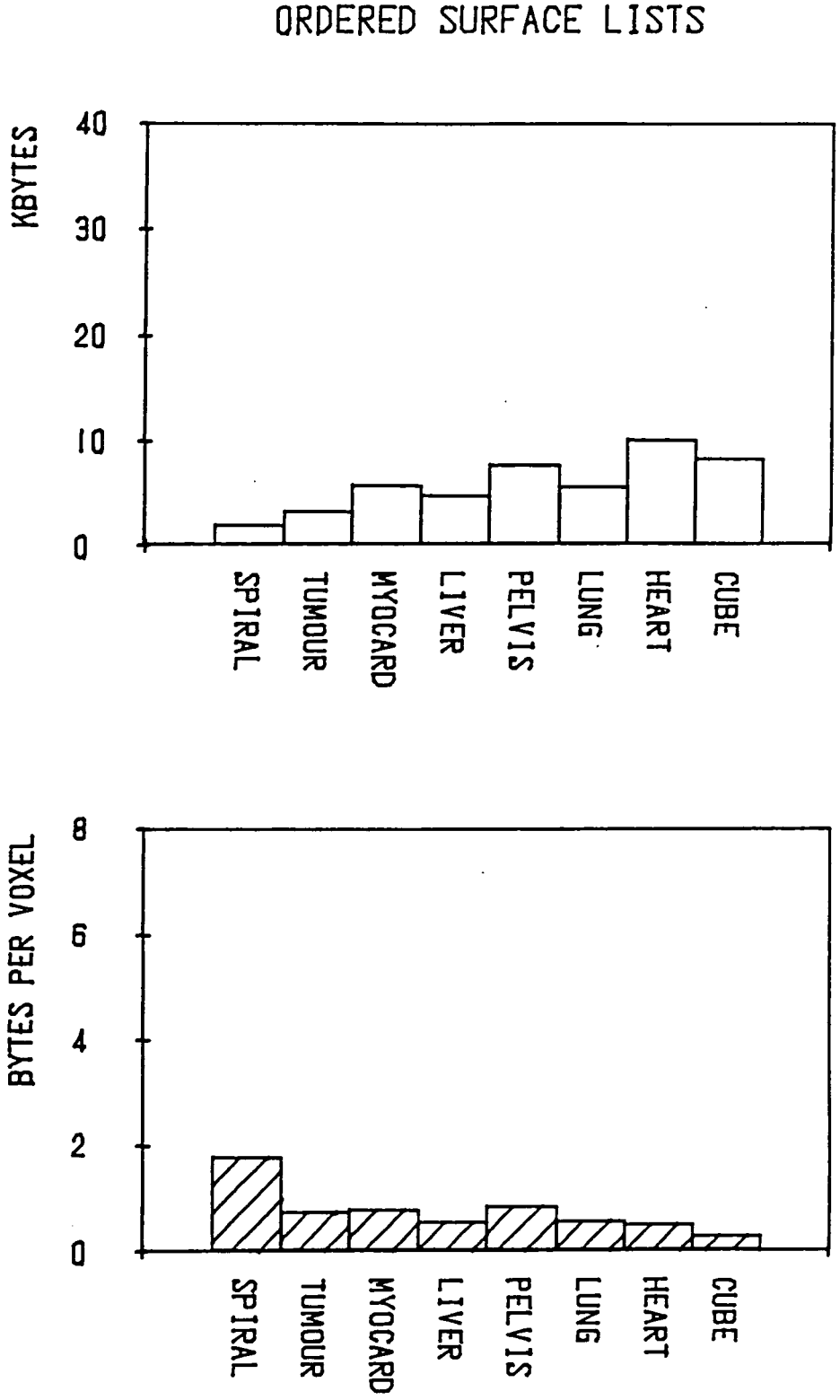


Figure 3.8 Storage requirements for the eight test objects using the Ordered Surface List representation :-

Upper - Total storage.
Lower - Storage per voxel.

3.6 Conclusions

The storage space required by each of the five representations assessed is shown in Table 3.2, normalised to 100 units for the raw data. The Surface Patch representation is not markedly more compact than the cuberille representations, and has far less flexibility for both object representation and object manipulation. Amongst the cuberille methods, the Binary Object scheme has the merit of requiring the same space for all objects sampled at the same resolution. This property simplifies programming, since all objects can be represented in a pre-defined area of memory.

The requirements for the Ordered Surface List representation are expressed as both total storage, and average storage per list, and the latter figures confirm that this is the most compact form of cuberille representation assessed. Although it is a surface representation, and therefore does not permit the operator to explore the interior of the object, it retains all the topological flexibilty of the cuberille approach.

The extrapolation of these results to other objects and to other sample sizes is justifiable. The eight test objects were carefully selected to include a range of sizes, shapes and topologies, and are likely to be representative of most medical objects. The results presented in Figures 3.3, 3.4 and 3.8 show the pattern expected for each algorithm as the object size increases, and these results are therefore a reasonable guide to the behaviour of each scheme at larger sample sizes. Storage requirements for three of the schemes are plotted against the object size, expressed as the number of voxels, in Figure 3.9. All three schemes are surface representations, and should require storage space to increase in proportion to the volume raised to the power 2/3.

Table 3.2 Summary of storage space required by all object representation techniques

Object	Data Storage (Relative units)				
	Surface Patch	Facet List	Binary Object	Raw Data	Ordered Surface List
Spiral	2.2	3.0	12.5	100	5.8/0.7
Tumour	4.5	4.7	12.5	100	10.0/1.3
Myocardium	8.7	8.3	12.5	100	17.9/2.2
Liver	7.6	7.8	12.5	100	14.6/1.8
Pelvis	10.3	10.9	12.5	100	23.7/3.0
Lung	8.5	7.8	12.5	100	17.1/2.1
Heart	15.0	14.3	12.5	100	31.3/3.9
Cube	12.8	8.8	12.5	100	25.5/3.2
Average	8.7	8.2	12.5	100	18.2/2.3
Average (Kbytes)	22.3	21.0	32.0	256	46.6/5.9
Average (bytes/voxel)	2.7	2.8	6.9	55	6.0/0.75

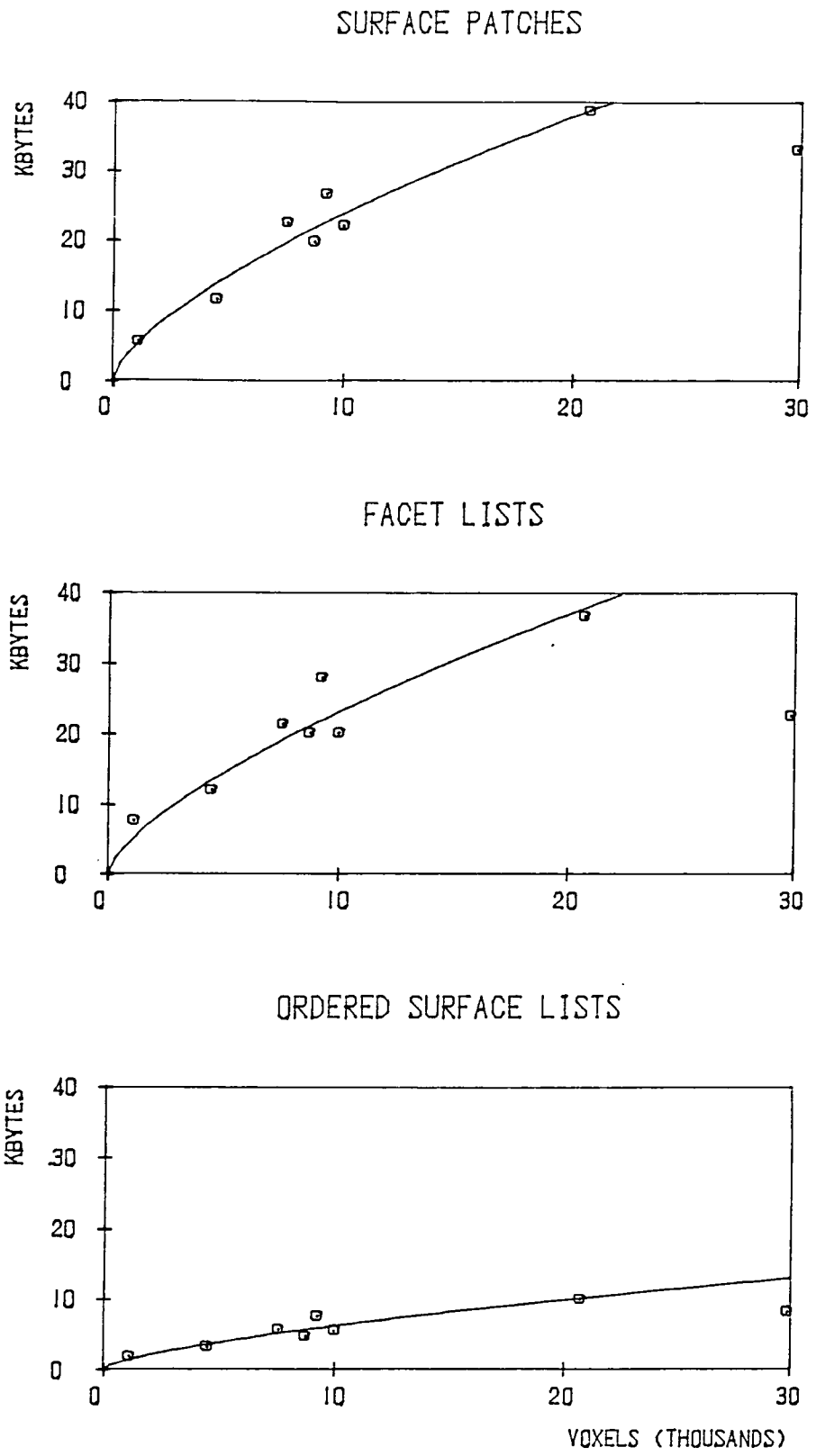


Figure 3.9 Storage requirements for the eight test objects using surface representations, as functions of object size. Power law functions were fitted to the data using a least squares procedure.

Upper - Bytes required = 52.6 [No. of voxels] $^{2/3}$
Middle - Bytes required = 51.6 [No. of voxels] $^{2/3}$
Lower - Bytes required = 13.8 [No. of voxels] $^{2/3}$

This is shown to be approximately true, with very similar constants of proportionality obtained for Surface Patches and Facet Lists, for the seven natural test objects. The Cube was excluded from the least squares fitting procedure because it was a wholly artificial object with a much smaller ratio of surface area to volume. The Ordered Surface List scheme shows a similar functional relationship, with a smaller constant of proportionality. As N increases, the Facet List and Ordered Surface List schemes become increasingly attractive, since the storage requirement for either of these techniques increases only $O(N^2)$. The Ordered Surface List scheme has the additional merit of being suitable for rapid display using a specially devised algorithm.

4. Display Algorithms

4.1 The Viewing Transformation

The essential feature of any three dimensional display technique is the ability to show the object as it appears from any aspect. Each point can be represented by a four element row vector:-

$$\mathbf{r} = (x\ y\ z\ 1) \qquad 4.1$$

where (x y z) specify the location of the point relative to the object coordinate system. The use of four elements to represent a three dimensional point is known as the homogeneous coordinate technique (Newman and Sproull 1973), and it simplifies all transformations in which the object moves relative to the coordinate system. To present an image of the object as seen from a different aspect, each point must be transformed into the observer coordinate system (Figure 4.1). Generally the observer may be located anywhere in the object space, and may be looking in any direction. The transformation which converts object coordinates to observer coordinates is known as the 'viewing transformation' (Newman and Sproull 1973). Any complex transformation such as this can be broken down into a sequence of elemental transformations. If the shape of the object must not change then the elemental transformations are restricted to translation, rotation and scaling (isometric expansion or contraction). Each elemental transformation can be represented by a (4 x 4) matrix, and the coordinates in the new coordinate system are obtained by matrix multiplication:-

$$\mathbf{r}' = \mathbf{r} \cdot \mathbf{T} \qquad 4.2$$

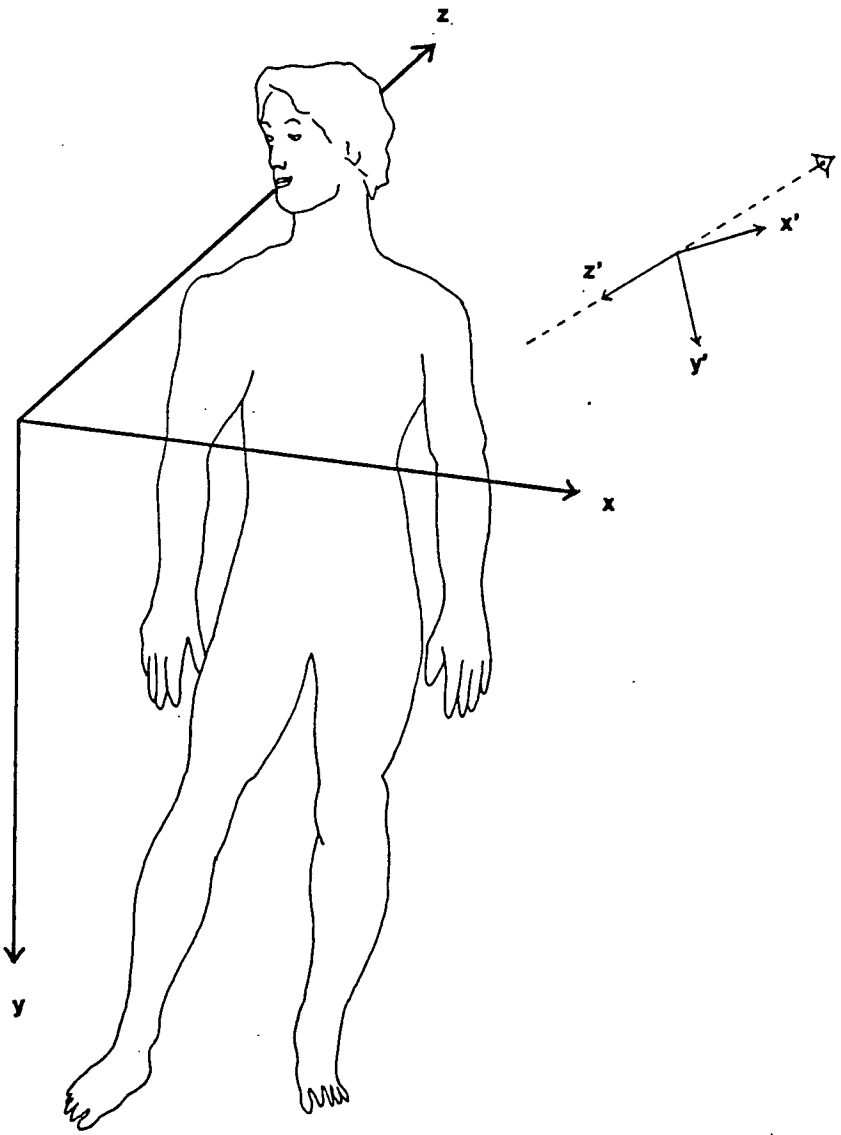


Figure 4.1 The object (xyz) and observer (x'y'z') coordinate systems, and their orientation relative to the patient.

where **r'** is a four element row vector containing the new coordinates and **T** is the (4 x 4) transformation matrix. If the new coordinate system has axes parallel to the old, but the new origin lies at the point (x_o, y_o, z_o), then the elemental translational transformation matrix is :-

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x_o & -y_o & -z_o & 1 \end{pmatrix} \qquad 4.3$$

Scaling the object coordinates by a factor s can be accomplished using the matrix:-

$$T = \begin{pmatrix} s & 0 & 0 & 0 \\ 0 & s & 0 & 0 \\ 0 & 0 & s & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad 4.4$$

There are three rotational transformations, corresponding to rotation about each coordinate axis. The convention is to regard a positive angle as representing anti-clockwise rotation as seen looking in the positive direction along that axis. Using that convention, rotation through angle A corresponds to:-

$$T_x = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos A & \sin A & 0 \\ 0 & -\sin A & \cos A & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad 4.5$$

$$T_y = \begin{pmatrix} \cos A & 0 & -\sin A & 0 \\ 0 & 1 & 0 & 0 \\ \sin A & 0 & \cos A & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad 4.6$$

$$T_z = \begin{pmatrix} \cos A & \sin A & 0 & 0 \\ -\sin A & \cos A & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad 4.7$$

where the subscripts indicate the axis of rotation. If several transformations are applied sequentially the final coordinates are:-

$$r' = ((r \cdot T_1) \cdot T_2) \cdot T_3 \qquad 4.8$$

or $$r' = r \cdot T \qquad 4.9$$

where $$T = T_1 \cdot T_2 \cdot T_3 \qquad 4.10$$

Any complex transform (such as the viewing transformation) may therefore be represented by a single matrix formed by multiplying several elemental matrices. This overall matrix needs to be computed only once for any viewing position and direction, and all object points may then be transformed into observer coordinates using equation 4.2.

The unusual orientations of the coordinate systems in Figure 4.1 have been chosen to correspond with the structure of the input data and the output display device. Transverse section tomograms are planes perpendicular to the y axis, and are normally stored as two dimensional arrays with axes aligned with the x and z axes. Similarly, medical imaging computers usually have display memories with the origin at the top left of the screen, with column and row addresses corresponding to the x' and y' axes of the observer coordinate system. The z' axis indicates the depth of all points relative to the observer position. As well as natural correspondences with the input data and the output

device, these coordinate systems contain only positive values. Any negative value must correspond to a point which is invisible, since it transforms outside the valid range of observer coordinates.

To derive the viewing transformation both the location of the observer and the direction of view must be specified. Let the origin of the observer coordinate system lie at the point (x y z) in object coordinates. The viewing direction can be specified by the unnormalised direction cosines of the viewing direction (l,m,n). A further constraint is required to produce a unique transformation, and it is convenient to constrain the observer x' axis parallel to the object (xz) plane. This allows the observer to view the object from all aspects, but ensures that the y axis will remain vertical in the new coordinate system. Given these parameters and the chosen constraint a generalised viewing transformation is (Appendix 1) :-

$$T = \begin{bmatrix} \frac{n}{u} & \frac{-lm}{uv} & \frac{l}{v} & 0 \\ 0 & \frac{u}{v} & \frac{m}{v} & 0 \\ \frac{-l}{u} & \frac{-mn}{uv} & \frac{n}{v} & 0 \\ \frac{1}{u}(lz-nx) & \frac{m}{uv}(lx+nz)-\frac{yu}{v} & \frac{-1}{v}(lx+my+nz) & 1 \end{bmatrix} \qquad 4.11$$

where $u^2 = l^2 + n^2$ and $v^2 = l^2 + m^2 + n^2$.

This general transformation may be used to obtain views of the object from any position. However, it requires six parameters to specify the observer position and viewing direction. This makes it difficult to incorporate interactive devices to alter the viewing transformation, since there are six degrees of freedom. A slightly more

restricted transformation with only two degrees of freedom is shown in Figure 4.2. The viewing direction is constrained to lie parallel to the line joining the centres of the object array and the image array, and the x' axis is constrained to lie parallel to the (xz) plane. If the distance of the observer is also fixed then the only variable parameters are the angles θ and Ø, which specify the observer's position. This technique allows the observer to move freely around the surface of a sphere, looking towards the centre of the object array at all times. The parameters θ and Ø correspond to the longitude and latitude of the observer's position on the sphere, and can be specified by any interactive device with two degrees of freedom (e.g. a trackerball, joystick or mouse). This restricted viewing transformation is given by (Appendix 1) :-

$$T = \begin{bmatrix} \cos\theta & \sin\theta.\sin\phi & -\sin\theta.\cos\phi & 0 \\ 0 & \cos\phi & \sin\phi & 0 \\ \sin\theta & -\cos\theta.\sin\phi & \cos\theta.\cos\phi & 0 \\ C(1-\sin\theta-\cos\theta) & C(1-\cos\phi) & C(1-\sin\phi) & 1 \\ & + C\cos\theta.\sin\phi & + C\sin\theta.\cos\phi & \\ & - C\sin\theta.\sin\phi & - C\cos\theta.\cos\phi & \end{bmatrix} \quad 4.12$$

where (C,C,C) are the coordinates of the centre of the object array. This transformation is not as general as that described by equation 4.11, since not all views can be generated. However, it is sufficiently flexible for use in examining almost all medical objects.

4.2 Rapid Transformations

Although the viewing transformation needs to be calculated only once for a given observer position and viewing direction, transformation

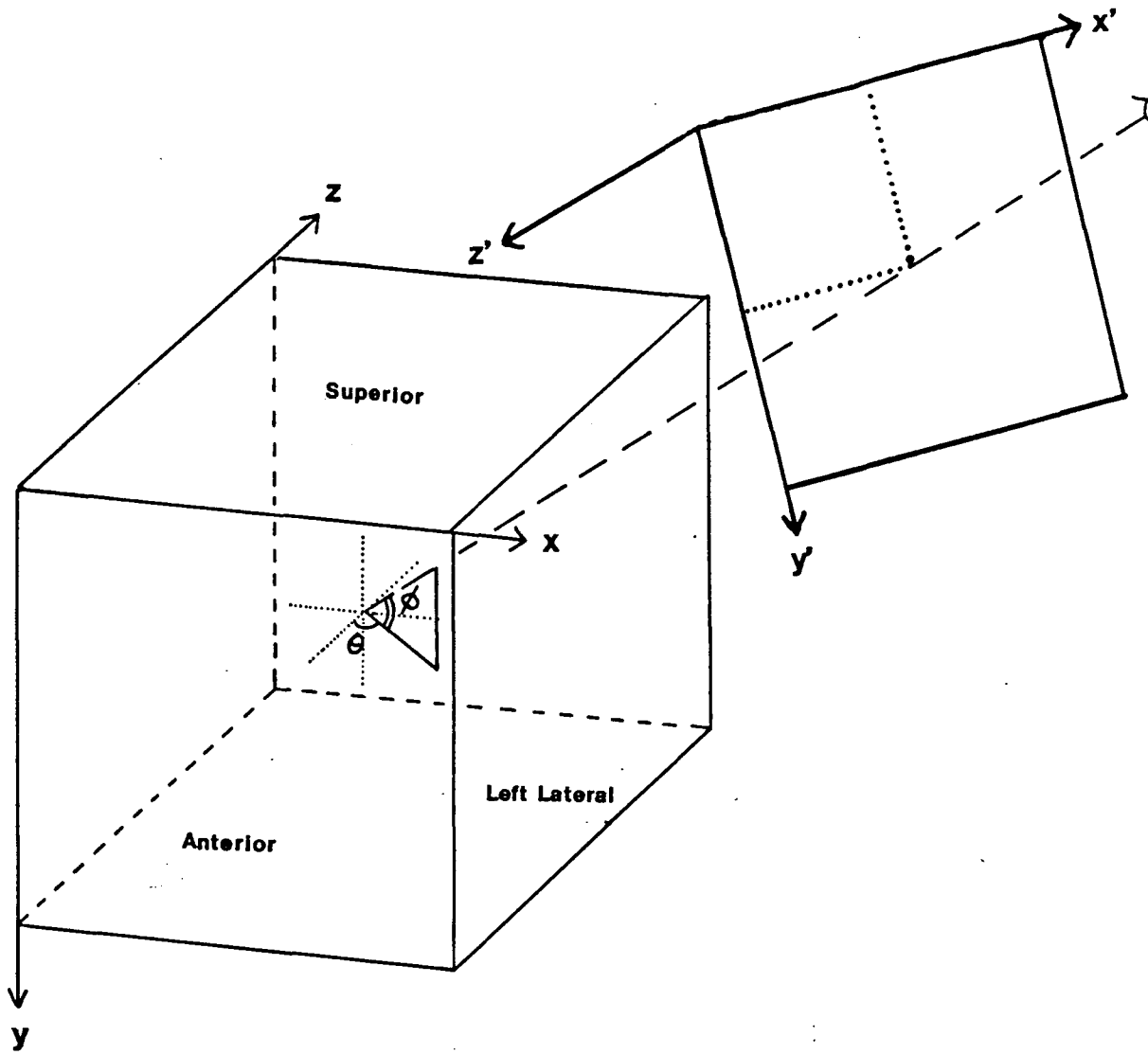


Figure 4.2 Definition of the parameters θ and ø for the restricted viewing transformation.

of the object into observer coordinates implies evaluation of equation 4.2 for all points in the scene. Direct evaluation will require three multiplications and three additions for each coordinate, or a total of nine multiplications and additions for each point to be transformed. However, object points have been obtained from a three dimensional array of N x N x N voxels. Using the coordinate scheme of Figure 4.1 object points can have only integer coordinates in the range 0 to N-1. Even if interpolation schemes are used to try and locate object points with sub-voxel accuracy (Hohne and Bernstein 1986), it is likely that points could still be represented by a limited range of integer values. It will be assumed here that points need only be located to the nearest voxel. Look-up tables containing pre-computed trigonometric values may be used to speed up the process of transformation (Frieder et al 1985). The observer coordinates are obtained from equations of the form:-

$$x' = x.T_{11} + y.T_{21} + z.T_{31} + T_{41} \qquad 4.13$$

with similar expressions for y' and z'. Since each of x, y and z can have only N values, then each term $(x.T_{11})$, $(y.T_{21})$ and $(z.T_{31} + T_{41})$ can also have only N values. Once the viewing transformation is known, nine look-up tables can be constructed containing separate terms $(x.T_{11})$, $(y.T_{21})$ etc. for all possible values of x, y, and z. Each look-up table contains N elements and computation of all the tables requires 9N multiplications and 3N additions. Subsequent calculation of observer coordinates for each object point requires nine look-up table accesses and six additions. Since table access will always be much faster than multiplication, this represents a considerable increase in speed over direct evaluation. The overhead required to compute the look-up tables

increases as O(N), whereas the saving in computation time increases as $O(N^3)$ for larger arrays.

Some hidden surface algorithms traverse the object array in sequence along each of the x, y and z axes (Gibson 1986b). Successive object points therefore differ by one voxel in either their x, y or z coordinate. For the case of traversal along the x axis two such points are,

$$\mathbf{r}_n = (x, y, z, 1) \qquad 4.14$$

$$\mathbf{r}_{n+1} = (x + 1, y, z, 1) \qquad 4.15$$

Assume that the transform of the first point is already known using equation 4.2,

$$\mathbf{r}_n' = \mathbf{r}_n \cdot \mathbf{T} \qquad 4.16$$

The transform of the adjacent point can be written as,

$$\mathbf{r}_{n+1}' = \mathbf{r}_{n+1} \cdot \mathbf{T} = (\mathbf{r}_n + \mathbf{x}) \cdot \mathbf{T} \qquad 4.17$$

where $\mathbf{x} = (1, 0, 0, 0)$ is a unit vector along the x axis. Expanding equation 4.17 we have,

$$\mathbf{r}_{n+1}' = (\mathbf{r}_n \cdot \mathbf{T}) + (\mathbf{x} \cdot \mathbf{T}) \qquad 4.18$$

or $$\mathbf{r}_{n+1}' = \mathbf{r}_n' + (T_{11}, T_{12}, T_{13}, T_{14}) \qquad 4.19$$

Equation 4.19 implies that the transformed coordinates of the second point can be obtained by addition of a row vector to the transformed coordinates of the first point. The row vector is formed from the elements of the transformation matrix, and is therefore the same for all object points. Similar results can be obtained for voxels adjacent along the y and z axes. For this special case, therefore, points may be transformed in sequence using only three additions, once the first point in a row has been transformed.

Some display algorithms also use a limited range of observer coordinate values. It is possible to display an N x N x N object array using an intermediate N x N image array containing the z' coordinates of the visible voxels (Chen et al 1985b). If this is the case then it may be possible to use integer representation of x' and y', and possibly also of z' values, enabling faster processing, since only integer functions need to be invoked. The look-up table approach is ideally suited for use with integer values, since only additions are performed (multiplications are slower and carry more risk of integer overflow). Even if sub-voxel precision is required (e.g. for more accurate z' values), it is still possible to use integral values in the look-up tables. By storing the integer value in the more significant byte (msb) of a 16 bit word, the less significant byte (lsb) is available to represent any fractional component, with a precision of 1 part in 256. All the look-up tables can be computed using this technique, and transformed coordinates are then obtained by 16 bit addition of these scaled integer values. The output word specifies the observer coordinate with integer and fractional components in the msb and lsb respectively. This approach utilises the look-up table technique to maximum effect, since integer addition is the most rapid arithmetic

procedure, without significantly compromising the precision with which observer coordinates are calculated.

4.3 Conventional Hidden Surface Algorithms

The observer coordinate system shown in Figure 4.1 was chosen to facilitate the display of a transformed point (x' y' z') using a raster display memory. However, if the object points were simply transformed and displayed in an arbitrary sequence then either all parts of the object would be visible, or some hidden surfaces would obscure parts which should be visible. Simulation of light shining on an opaque solid requires that only the visible parts of the object should be displayed, with elimination of the hidden surfaces. Many hidden surface algorithms have been described (Sutherland et al 1974) and there is no single algorithm which will always be the best. Algorithms operate on different object representations, of varying degrees of complexity, with different time constraints. It is not surprising that several approaches have been made to this problem.

Most general purpose hidden surface algorithms assume that the object is represented by a list of surface patches, (as described in Section 3.4), which has been constructed in an arbitrary order. The vertices of each face can be transformed into observer coordinates using the techniques of Section 4.1. Simple traversal of the transformed list with shading and display of each patch in sequence will give an incorrect image, since hidden surfaces will overwrite and hence obscure visible surfaces if they are encountered later in the list. The Depth-Buffer algorithm (Blinn and Newell 1976) avoids this by using a depth array equal in size to the image array. As each polygonal face is encountered the image array pixels which lie within it are identified. For all these pixels the depth of the polygon at that pixel location is

calculated, and compared to the existing value in the depth array. If the calculated depth is greater than the value in the depth array then that portion of the polygon is hidden by another part of the object, and can be ignored. If the polygon depth is less than the depth buffer value, then the depth buffer is updated, and the corresponding pixel in the image is set to display the shade associated with the polygon. After all the polygons have been processed the image array contains shading values associated with the visible faces only. This simple algorithm is easily implemented, and has been applied to the display of medical images (Herman and Liu 1979). Unfortunately it requires a large depth buffer with the same number of elements as the display. If this is impracticable the display may be split into a number of smaller segments, each of which can be processed in turn. However, the polygon list must then be traversed several times to create the complete image.

The depth buffer algorithm treats each pixel in the output image individually. In general, pixels along the same line or in the same area show considerable coherence, in that they frequently display the same polygon. This property has led to the development of algorithms which exploit either Scan-Line Coherence or Area Coherence. The order in which polygons are processed, however, remains arbitrary. An alternative approach uses the concept of depth priority amongst the faces of the object to sort them into a chosen order before display. One such technique has been described as the Painter's Algorithm, (Newman and Sproull 1973), since it resembles the way in which a landscape painter might choose to complete the background before overpainting with foreground details. After transforming into observer coordinates the polygons are sorted into priority order based on their depth, with more distant faces listed first. The tests required to

determine priority order may require careful calculation to determine whether a given polygon obscures another. Moreover, the list is only valid for a given viewpoint, and the priorities will change if the observer moves or alters the viewing direction. Once completed, the list may be traversed to display each polygon in sequence, overwriting earlier polygons without consideration of individual pixels.

This concept of priority ordering is extremely fruitful, and has been extended in several ways :-

Back faces: Triangular faces are represented by an ordered triplet of vertices, e.g. ABC rather than ACB. The conventional order for enumeration of vertices is clockwise as seen from within the object looking out. This ordering can also be used to define a surface normal vector, which points outwards from the object. If the object is composed of a closed mesh of connected facets, then faces with surface normals pointing away from the observer can never be seen. These back faces can be discarded immediately after transforming into observer coordinates, thereby reducing the length of the list, and giving a significant decrease in the time required to perform a priority sort.

Faces of cubes: The cuberille models described in Section 3.5 represent the object using many small cubes. To display the object, however, the square faces of cubes lying on the surface have been used. For any viewing transformation these faces can have only six orientations, three of which represent back faces which can be discarded. The priority list for the remainder can be created using depth sorting only, since it has been shown that this resolves all obscuration priorities for cubic faces (Herman and Udupa 1982). Hence the time required to solve the hidden

surface problem is less than would be expected for the relatively large number of faces in the model.

Space partitioning trees: In many cases the position of the observer changes more frequently than the shapes or positions of the objects being displayed. This is frequently true for medical objects, where a complex and probably interactive segmentation procedure may be followed by visual inspection of the object from many angles. If the object is represented by a list of polygonal surface facets, then these can be arranged into a binary space-partitioning tree. For any observer position and viewing direction this tree can always be traversed in such a way that polygons are enumerated in priority sequence (Fuchs et al 1983). This reduces the time required to display the object considerably, since determining the priority order can be achieved without any sorting at all. However, should the object change (e.g. because the observer alters the segmentation threshold) then both the list of surface facets and their arrangement in a space-partitioning tree must be recalculated before the new object can be displayed.

Inherent priority ordering: A cubic array has an inherent priority order which can be exploited to determine the display sequence. Using the object coordinate system of Figure 4.1, voxels with the same z coordinate form a coronal section through the body. Voxels in adjacent coronal planes have either higher or lower priority for display, depending on whether the current viewing direction shows an anterior or a posterior aspect. Similar arguments apply to planes of constant x (sagittal sections) and constant y (transverse sections). Hence for any viewing direction the array can always be traversed in priority order, without the need for sorting or even transformation into observer

coordinates. In this work, only cuberille representations were used, together with hidden surface algorithms based on the inherent priority ordering of a cubic array.

4.4 Direct Display of Binary Objects

The Binary Object representation described in Section 3.5 uses a cubic array with a single bit to represent each voxel. Bits are set to one if the voxel is part of the object, and to zero if the voxel is part of the background. If the viewing transform of equation 4.12 is used with $\theta=\emptyset=0$, then the transformation matrix reduces to the identity matrix. This corresponds to the observer looking along the z axis, and implies that object and observer coordinates are identical for every voxel. For this special case an extremely rapid hidden surface algorithm was developed (Gibson 1983).

A pixel (x', y') in the output image is aligned with the set of voxels in the object which have $x = x'$ and $y = y'$. The visible voxel at that point in the image is therefore the first object voxel encountered as the set is traversed along the z axis in the positive direction. For an N x N x N object array this corresponds to searching N adjacent voxels to find the first non-zero bit, for each of N x N pixels. This is a ray-tracing algorithm, but differs from those described by other workers (Radermacher and Frank 1984, Tuy and Tuy 1984), because the rays are constrained to travel along a direction parallel to a major axis of the cubic array. This means that the algorithm can be readily implemented in assembly language, and executes rapidly. The set of test objects shown in Figure 3.2 and used to assess object representation schemes was also used to assess display techniques. Figure 4.3 shows the size of each test object after segmentation, together with the time

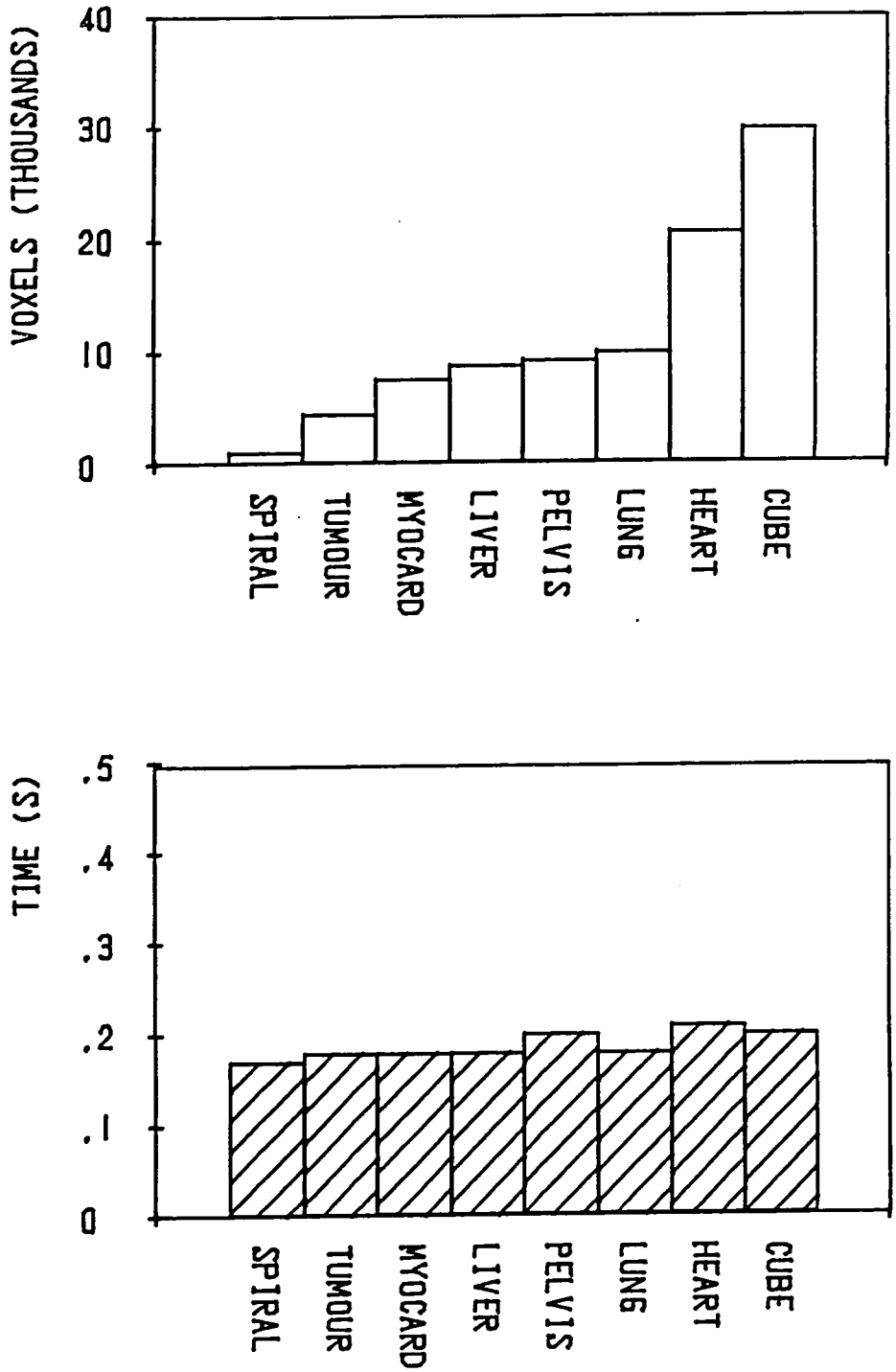


Figure 4.3 The time required to determine the visible surfaces of the test objects using Direct Display of the Binary Object representation :-

Upper - Object size.
Lower - Execution time.

required to determine the visible voxels using Direct Display of a Binary Object. This time remains essentially constant despite the wide range of object sizes.

It might have been anticipated that a ray tracing algorithm such as this would execute more rapidly for larger objects, since more of the rays would intersect the object and so terminate earlier. In the extreme case of an entirely empty object array, all N^3 voxels must be examined. However, the detailed structure of the binary object array described in Section 3.5 enabled empty areas of the array to be interrogated efficiently. Searching 64 voxels along a direction parallel to the z axis of the binary array corresponds to examining four 16-bit words of memory. Only a single instruction is required to determine whether an entire word is zero, thereby assessing 16 voxels simultaneously. However, when a word is non-zero, several steps are necessary to determine the first non-zero bit within the word. The time required to process a ray which intersects the object is therefore similar to that required to process one which passes wholly through the background, and the algorithm executes at approximately the same rate for both large and small objects.

This hidden surface algorithm can only be used for a very limited range of viewing positions. Searching along either x, y or z axes in either positive or negative directions allows six orthogonal images to be produced, but oblique aspects cannot be obtained. Moreover, only searches parallel to the z axis have the ability to assess 16 adjacent voxels simultaneously. In order to show the object from other angles it must be rotated in the binary object array prior to executing the hidden surface algorithm. The transforms described in equations 4.3 to 4.7 show how the object coordinates change when the coordinate system moves.

The inverse transformations may also be used to show how object coordinates change when the object moves and the coordinate system remains fixed (Newman and Sproull 1973).

By analogy with equation 4.2 we have,

$$\mathbf{r}' = \mathbf{r}.\ \mathbf{T} \qquad 4.20$$

where $\mathbf{r}'$ specifies the new coordinates of the point $\mathbf{r}$, and $\mathbf{T}$ is the transformation matrix. Using the coordinate system of Figure 4.1, let the object be rotated through an angle θ about an axis parallel to the y axis which passes through the object centre (C,C,C). The transformation matrix is given by:-

$$T = \begin{bmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ C(1-\cos\theta+\sin\theta) & 0 & C(1-\cos\theta-\sin\theta) & 1 \end{bmatrix} \qquad 4.21$$

A similar rotation of the object through angle $\emptyset$ about an axis parallel to x passing through the centre of the object corresponds to the transformation:-

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\emptyset & -\sin\emptyset & 0 \\ 0 & \sin\emptyset & \cos\emptyset & 0 \\ 0 & C(1-\cos\emptyset-\sin\emptyset) & C(1-\cos\emptyset+\sin\emptyset) & 1 \end{bmatrix} \qquad 4.22$$

Rotations about axes parallel to the z axis do not reveal any further details of the object and will generally not be required. In the implementation of this display technique the observer was presented with 'rotate' and 'tilt' options corresponding to equations 4.21 and 4.22 respectively. This separation of the two possible modes of object manipulation facilitated transformation of large object arrays in the limited memory space of a typical medical imaging computer. In both cases one coordinate always remains unchanged after transformation, either y (in the case of equation 4.21) or x (in the case of equation 4.22). Transformation of an N x N x N object array may therefore be performed using a buffer array of only N x N bits, rather than requiring an additional array of N x N x N bits to hold the new object during transformation. The method can be illustrated by considering rotation of the object about a vertical axis. To implement equation 4.21 a plane with constant y coordinate was copied from the object array into the N x N buffer. The voxels in that plane of the object array were set to zero, and then filled from the buffer using transformed x and z coordinates obtained by applying equation 4.21. The process was repeated for all planes of constant y coordinate in the object array, rotating the object one horizontal slice at a time. An exactly similar technique was used to implement equation 4.22, using planes of constant x. The transformation of an N x N x N array can therefore be accomplished using only $N^3 + N^2$ bits rather than $2N^3$ bits.

The time required to rotate an object to obtain a view from another aspect is almost independent of both the size of the object and of its complexity. Figure 4.4 shows the time required to rotate (about the y axis) and tilt (about the x axis) the test objects. Times range from 5.0 to 8.7 s, hence this display technique can be used

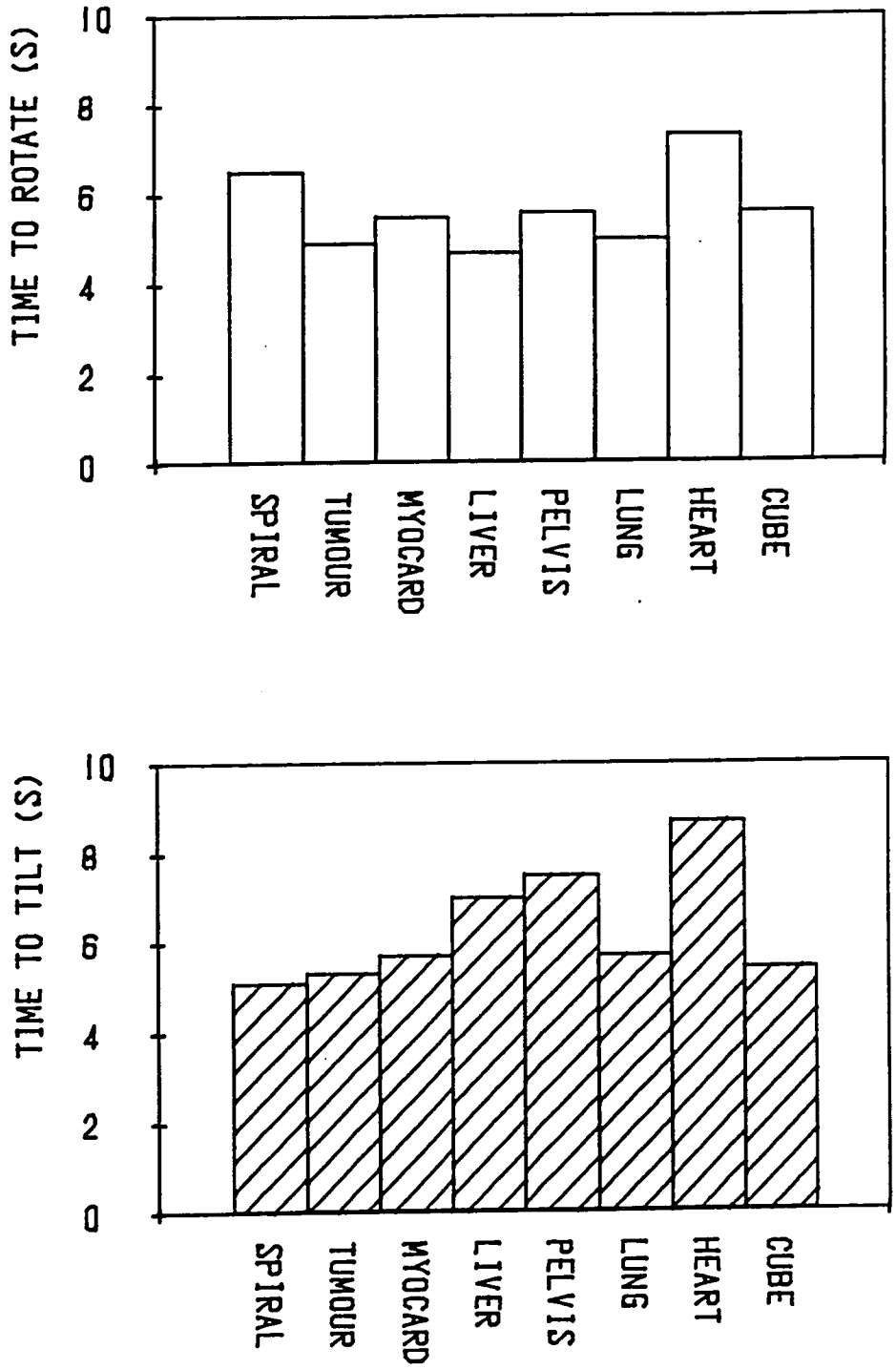


Figure 4.4 The time required to rotate and tilt the test objects using the Binary Object representation :-

Upper - Time to rotate through 30^{o}.
Lower - Time to tilt through 60^{o}.

interactively for objects sampled at 64 x 64 x 64 resolution. Larger object arrays will require disk storage, with consequent increase in the time required to manipulate the object. Even if extremely large arrays could be stored in main memory, the execution time will increase $O(N^3)$, since all voxels are processed whenever the object is to be displayed from a different aspect.

At first sight the results in Figure 4.4 seem to imply that tilting objects through large angles takes longer than rotation through small angles. In fact the time taken to perform rotation about either axis is independent of the angle, and is principally affected by the linear size of the object parallel to the axis of rotation. Longer objects have at least one voxel present in more planes perpendicular to the axis of rotation, and therefore require more processing. This can be seen by comparing the results obtained for the Spiral and the Liver. The Spiral is relatively tall and narrow, and takes longer to rotate (6.5 s) than to tilt (5.1 s). The Liver, however, is relatively short and wide, and hence takes longer to tilt (7.0 s) than to rotate (4.7 s).

Once the object has been rotated the visible voxels can be determined using the simple algorithm described above, and an image produced based on either depth or surface shading. One feature of the orthogonal ray-tracing used by this algorithm is that voxels can never be 'missed' by the use of too large a ray increment. Figure 4.5 shows two binary objects obtained from a set of CT slices through the skull and from a set of radionuclide tomograms of the cardiac blood pool. They are displayed with individual voxels shaded to show the cuberille nature of this representation, together with surface shading.

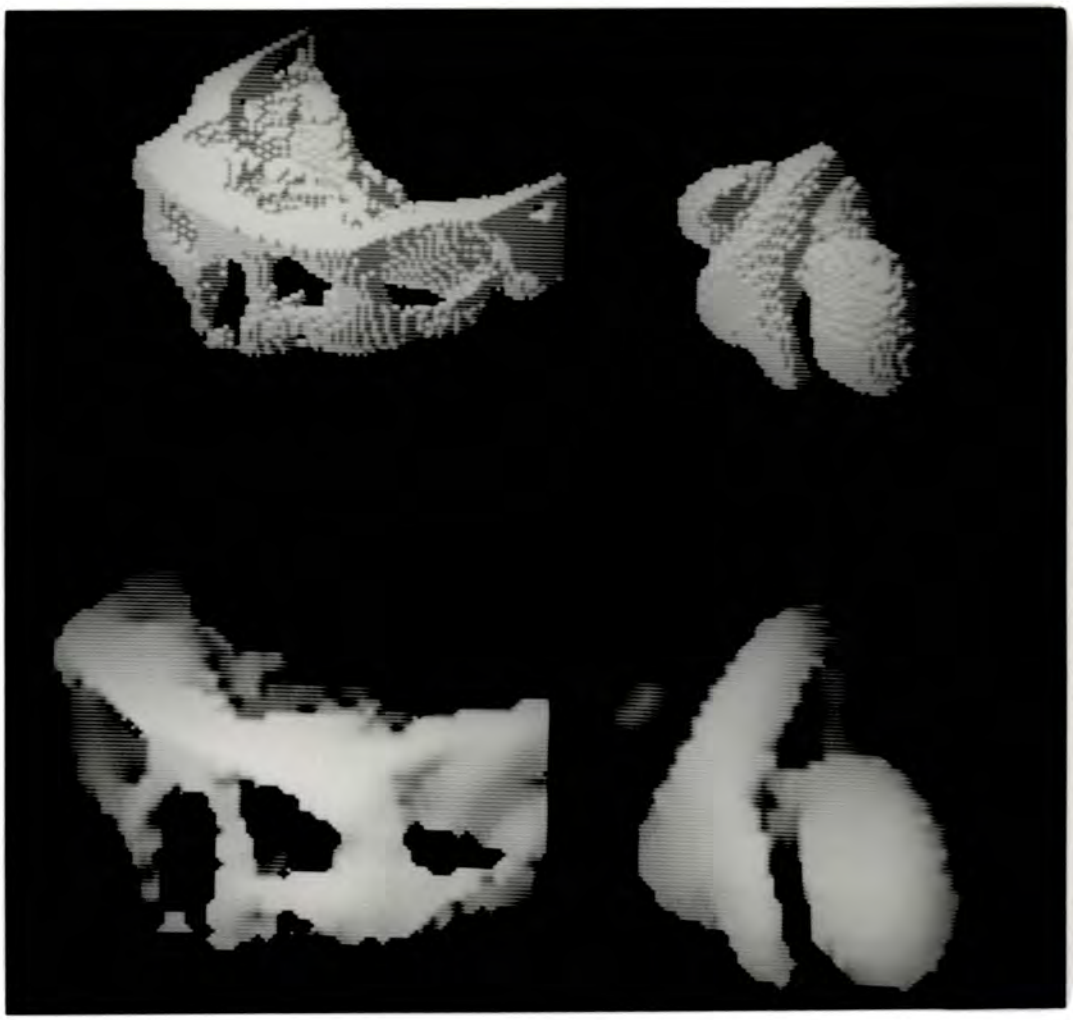


Figure 4.5 Part of the skull, and a heart, represented by Binary Objects :-

Upper left - Skull derived from CT data.
Upper right - Heart derived from ECAT data.
Lower left - Depth shading of the skull disguises the discrete cuberille nature of this representation.
Lower right - Depth shaded image of the heart.

4.5 Back-to-Front Display of Cuberille Arrays

The inherent priority ordering of a cubic array was discussed in Section 4.3, and forms the basis for a display algorithm which requires no sorting of object voxels to determine the order for display, and yet which allows complete freedom of choice for the observer's position and viewing direction. Let an N x N x N object array be indexed along each of the x, y and z axes using integer values in the range 0 to N-1. A triple nested loop can be used to visit all voxels in sequence:-

Algorithm 4.1

```
for z = 0 to N-1 step 1
  for y = 0 to N-1 step 1
    for x = 0 to N-1 step 1
          .
          .
      process voxel at location (xyz)
          .
          .
    next x
  next y
next z
```

Any of the coordinates may be traversed in the opposite direction using a loop of the form:-

```
for  y = N-1 to 0 step -1
```

By combining positive and negative traversals along each of the three axes eight possible traversal patterns can be created. It has been pointed out by several authors that one of these patterns will always generate a correct hidden surface image (by visiting the voxels in

priority order) for any viewing direction (Frieder et al 1985). Let the viewing direction have unnormalised direction cosines (l,m,n) relative to the (x,y,z) axes of the object coordinate system. If the sign of a direction cosine is negative, then the corresponding axis should be traversed in a positive direction, and vice versa. This ensures that the object array is always traversed from the back (as seen from the current viewing direction) to the front, hence this algorithm is known as the back-to-front or BTF algorithm. Figure 4.6 shows a cuberille model of lungs displayed using this method. As the array is traversed, voxels are transformed into observer coordinates and displayed in the output image.

The traversal pattern ensures that nearer voxels are encountered later, and hence obscure more distant voxels. This process can be seen at four stages in Figure 4.6. The three axes of the object array correspond to successive slices, rows and columns. The outermost loop is around the slices, and these are being processed from the bottom of the lungs to the top. If the aspect to be displayed was inferior rather than superior, the slices would be processed from the top to the bottom. The middle loop is around the rows, and these are being processed from the back of the patient to the front. If the aspect to be displayed was posterior rather than anterior this would be reversed. Finally, the inner loop is around the columns, and these are being processed from the right side of the lungs to the left. If the aspect to be displayed was right lateral rather than left lateral this in turn would also be reversed.

The BTF algorithm can be applied to cuberille models using the raw data (Frieder et al 1985), lists of object voxels obtained after segmentation, binary object arrays, or ordered surface lists (Gibson

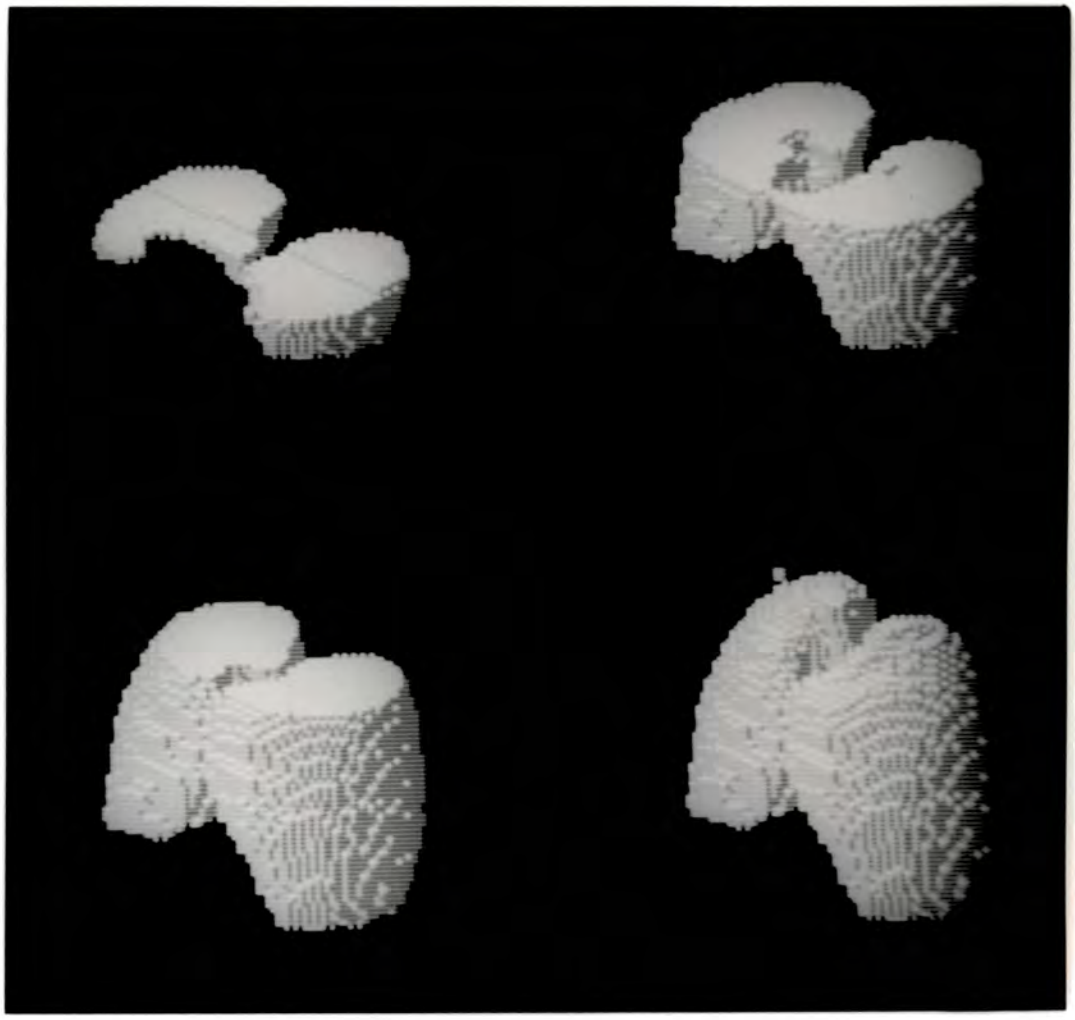


Figure 4.6 Stages in the back-to-front (BTF) display of a lung model, showing the build up of the image from (with respect to the patient), bottom to top, posterior to anterior, and right to left.

1986b). If it is applied to the raw data then segmentation is implemented during the display stage. In this case the data set to be processed is too large to fit into main memory (see Section 3.5), and successive frames have to be read from disk. However, the BTF algorithm is well suited to this approach. Providing that each coordinate axis is traversed in the correct direction then it does not matter which coordinate forms the outer, middle or inner loop. For object data stored as transverse sections the y coordinate of the object coordinate system corresponds to successive slices, which form (xz) planes. Then the BTF algorithm can be summarised as:-

Algorithm 4.2

```
for y = yfirst to ylast step ydirection
  read slice (y) into array OBJECT
  for z = zfirst to zlast step zdirection
    for x = xfirst to xlast step xdirection
      if OBJECT (x,z) is part of the object then
        transform (xyz) to (x'y'z')
        set IMAGE (x',y') to a value derived from z'
    next x
  next z
next y
```

where yfirst, ylast, ydirection and similar variables are deduced from the viewing direction, OBJECT is an N x N array holding successive transverse sections, and IMAGE is an N x N output array which will contain the depth values of the visible voxels when the algorithm has finished. This array is computed with minimal disk access overhead, since sections are read only once, in simple succession from one end of

the file to the other. Note that no restrictions are placed on the viewing transformation, which can therefore show the object from any aspect. All the rapid transformation techniques of Section 4.2 may be used, since the algorithm uses integer object and observer coordinates, and traverses the object in sequence along each axis. Figure 4.7 shows the time required to display the test objects using Algorithm 4.2 applied to the raw data representation. The display time is similar to that required to rotate or tilt the binary object representation.

Larger objects take longer to process, as would be expected from the form of Algorithm 4.2. This can be demonstrated more clearly by appplying the BTF algorithm to a list of object voxels created after segmentation. These were stored on disk using 3 bytes per voxel, to represent the x, y and z coordinates. The time required to process the lists is also shown in Figure 4.7, together with the time per voxel. For the smaller objects the list of voxels is relatively short, and display is consequently quicker than for the raw data representation. For larger objects the list may be so long that the additional time required to read successive voxel coordinates from the disk almost outweighs the advantage of not having to segment the object during the display stage. The time taken to display the smallest object (the Spiral) decreases by a factor of more than six, whereas the time taken to display the largest object (the Cube) remains essentially unchanged. The time per voxel shows little variation with size for medium to large objects, once disk access time becomes nearly proportional to the list length, confirming that the overall execution time of the BTF algorithm is approximately proportional to the object size.

The speed of execution of this technique can be further improved by rejecting voxels which can never be seen. For any viewing direction

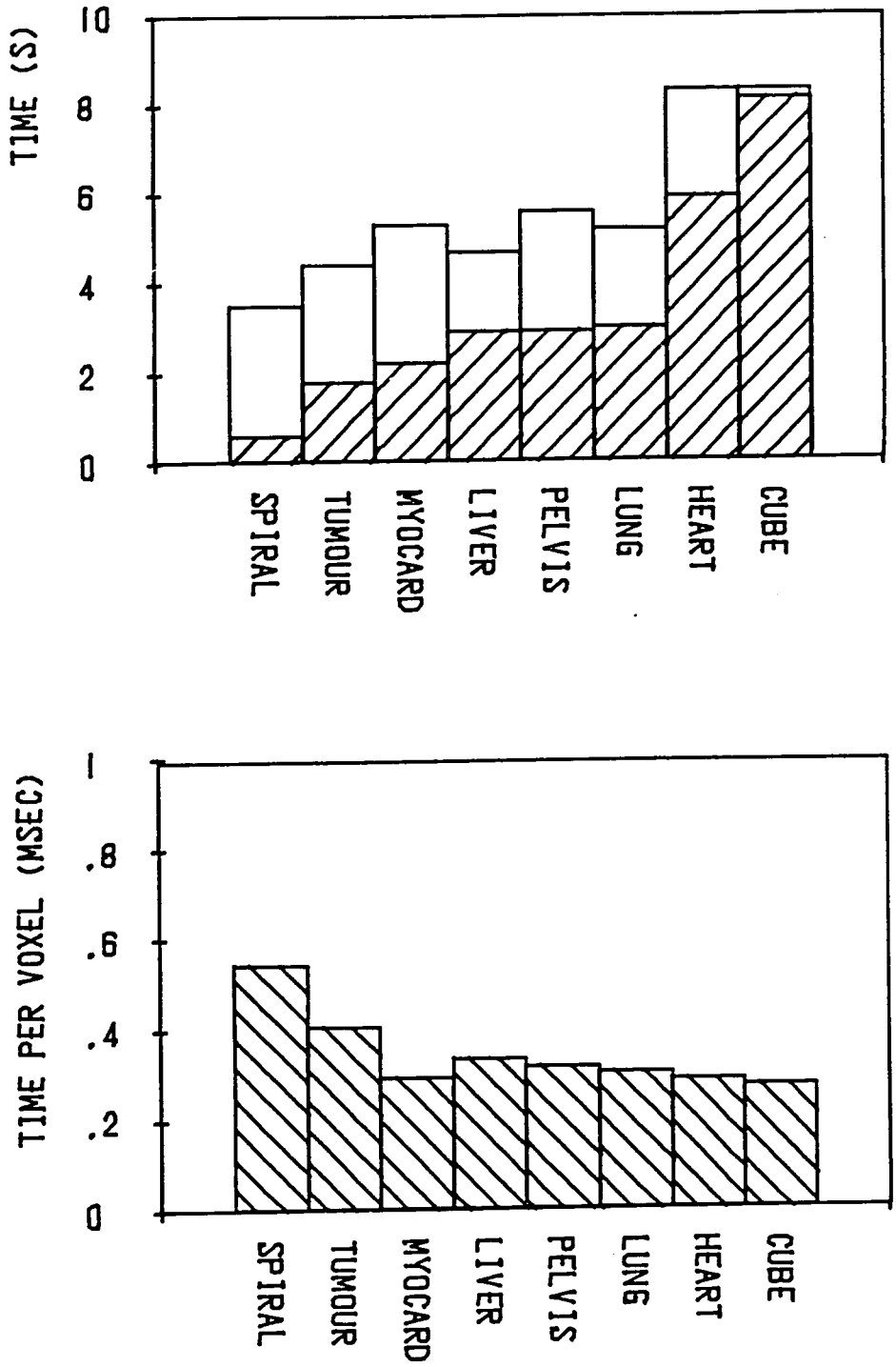


Figure 4.7 The time required to determine the visible surfaces of the test objects using the original BTF algorithm :-

Upper (open) - Using the raw data.
Upper (hatched) - Using a list of segmented voxels.
Lower - Time per voxel for the segmented list.

only a subset of the N^3 voxels in the array can ever contribute to the final image. Assuming that a segmentation threshold has been selected, then a list of potentially visible voxels may be created. Voxels will be discarded for three reasons :-

a) They are background voxels, i.e. they are not considered to be part of the object using the current segmentation criteria.

b) They are object voxels which are wholly enclosed by other object voxels. If all six orthogonally nearest neighbours are also part of the object then the central voxel can never be seen.

c) They are object voxels forming part of the surface of the object, but their only surface facets are on the opposite side of the voxel to the observer, and so cannot be seen using the current viewing direction.

Application of these criteria to the object array results in considerable data reduction. At first sight the subset of voxels selected using criterion c might be expected to alter continuously as the viewing direction altered, requiring many different lists. However, cubic voxels always have three facets visible from any direction, and these must be adjacent. If we consider the number of combinations of facet triplets, there are six possible choices for the first facet, four for the second (to be adjacent to the first) and two for the third (to be adjacent to both the other facets). Ignoring permutations of the same facets (of which there are six for every triplet) there are therefore only eight unique combinations of visible facet triplets, which correspond to the eight sets of adjacent facets which meet at each of the vertices. Hence there are only eight possible subsets of voxels which will be selected using criterion c. This degree of data repetition, requiring eight lists of object voxels rather than one,

might be considered excessive, were it not for the fact that each of the eight sets of potentially visible voxels corresponds to one of the eight traversal patterns for the BTF algorithm. This correspondence has been exploited to develop a modified form of the BTF algorithm using the Ordered Surface List representation (Gibson 1986b, Gibson 1987).

The algorithm operates in two stages, firstly using the raw data to segment the object and to create eight lists of potentially visible voxels, and secondly using the Ordered Surface Lists to allow rapid interactive display from any angle. Each list is created by traversing the cuberille array in the same order as for the corresponding BTF traversal. As each object voxel is encountered it is added to the list only if it is potentially visible using a viewing direction corresponding to the order of traversal. Once complete each list is stored on disk as successive (x,y,z) coordinates, requiring 3 bytes per voxel (see Section 3.5). Subsequent passage through the list from top to bottom corresponds to a full BTF traversal of the original object, but visiting only potentially visible voxels. Since these represent a subset of all the voxels in the original array, transformation and calculation of the output image is correspondingly more rapid. Although the algorithm generates a list of all voxels which may be seen using a particular viewing direction, it does not follow that all those in the list will be seen. The criteria for visibility consider only a voxel and its six nearest neighbours. It will commonly be the case that a potentially visible voxel is actually obscured by an entirely separate part of the object. Because the Ordered Surface List is created using a BTF traversal, a correct hidden surface image will still be generated. The algorithm for generating one of the lists is :-

Algorithm 4.3

```
for y = yfirst to ylast step ydirection
  read slice (y) into array OBJECT
  read slice (y + ydirection) into array NEIGHBOUR
  for z = zfirst to zlast step zdirection
    for x = xfirst to xlast step xdirection
      if OBJECT (x,z) is part of the object and is visible then
        add (x,y,z) to the list
    next x
  next z
next y
output list to disk
```

Note the similarity between this and Algorithm 4.2; providing that the time required to determine whether OBJECT (x,z) is visible is short, then the time required to create one list is approximately the same as that required to create one visible surface image using the original BTF algorithm. The array NEIGHBOUR is an N x N array which may be required to determine whether voxels in array OBJECT are visible. Subsequent display of the voxels stored in the list can be accomplished using an algorithm of the form:-

Algorithm 4.4

```
repeat
  obtain next (x,y,z) coordinates from list
   transform (x,y,z) to (x',y',z')
   set IMAGE (x',y') to a value derived from z'
until end of list
```

This very simple algorithm is relatively easy to code in assembly language using the rapid transformation techniques of Section 4.2.

4.6 Evaluation of the modified BTF algorithm

The two stages of this algorithm were evaluated separately. The pre-processing stage consisted of threshold selection, object segmentation and the creation of eight surface lists (Algorithm 4.3). Figure 4.8 shows that the mean number of visible voxels falls rapidly with increasing threshold for four typical medical objects. Each object was derived from a set of standard radionuclide tomograms sampled at 64 x 64 resolution. Figures 4.9 to 4.12 show how the object appearance alters as the count threshold is raised. In Figure 4.9 the liver and spleen are isolated from the surrounding tissues at all threshold levels above 300 counts per voxel. The cardiac blood pool shown in Figure 4.10 is optimally segmented at levels 500 or 550, showing both ventricles separated by the intraventricular septum. Figure 4.11 shows the distribution of perfused lung as a function of isocount threshold. In this case no single image can be regarded as having unique status, and the observer must select the isocount level (which corresponds to the degree of perfusion) which is of interest. For the myocardium displayed in Figure 4.12 there is again an optimum threshold value in the range 450 to 550, showing the area of infarction.

Although an optimum or preferred object appearance can usually be identified, it may require several attempts before the operator is satisfied that the object has been correctly segmented. Hence it is desirable that this stage should be reasonably rapid, to facilitate the interactive selection of the correct threshold. Table 4.1 shows how the time required to create all eight lists of visible voxels varied as a

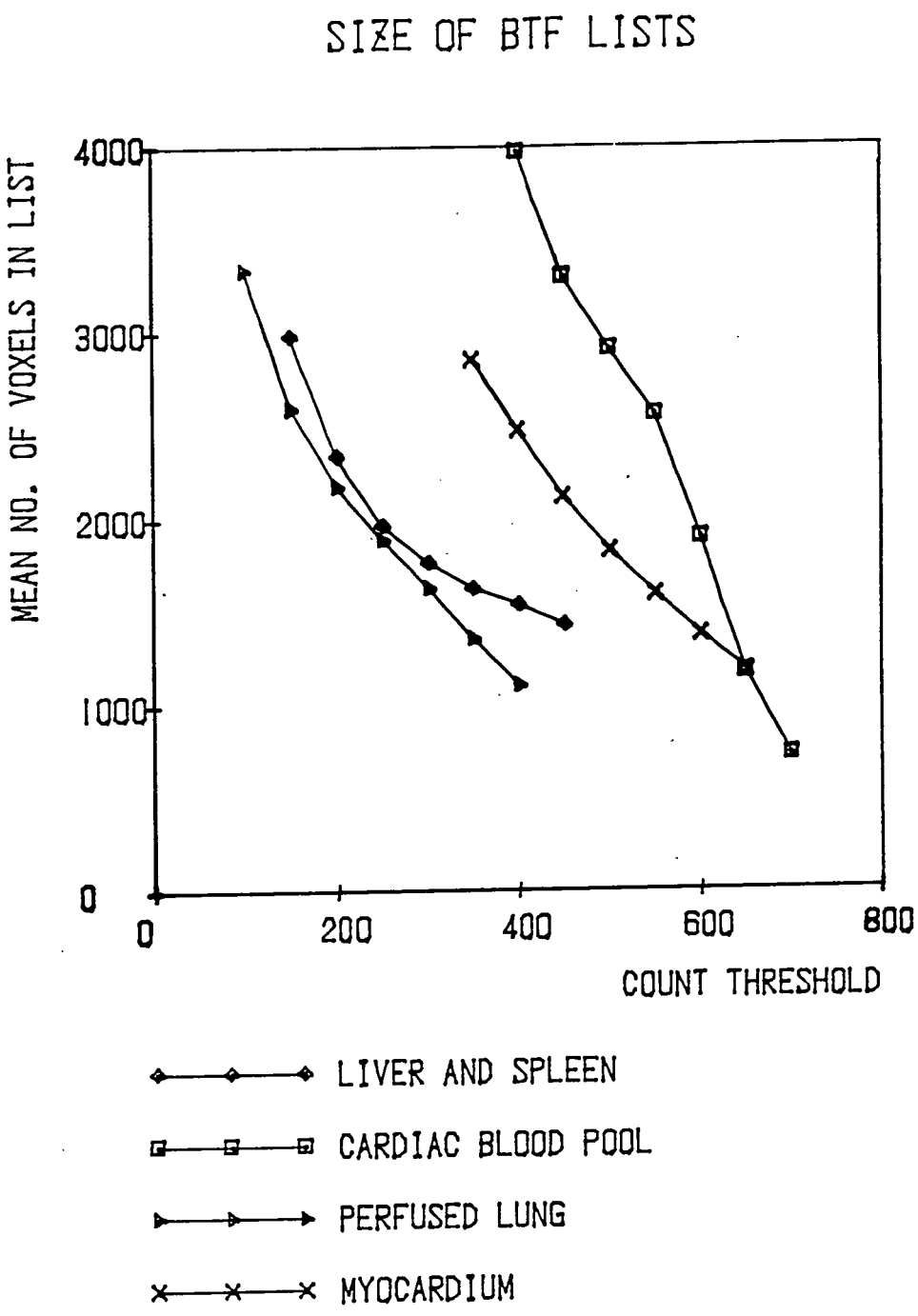


Figure 4.8 The mean size of Ordered Surface Lists as a function of the segmentation threshold.

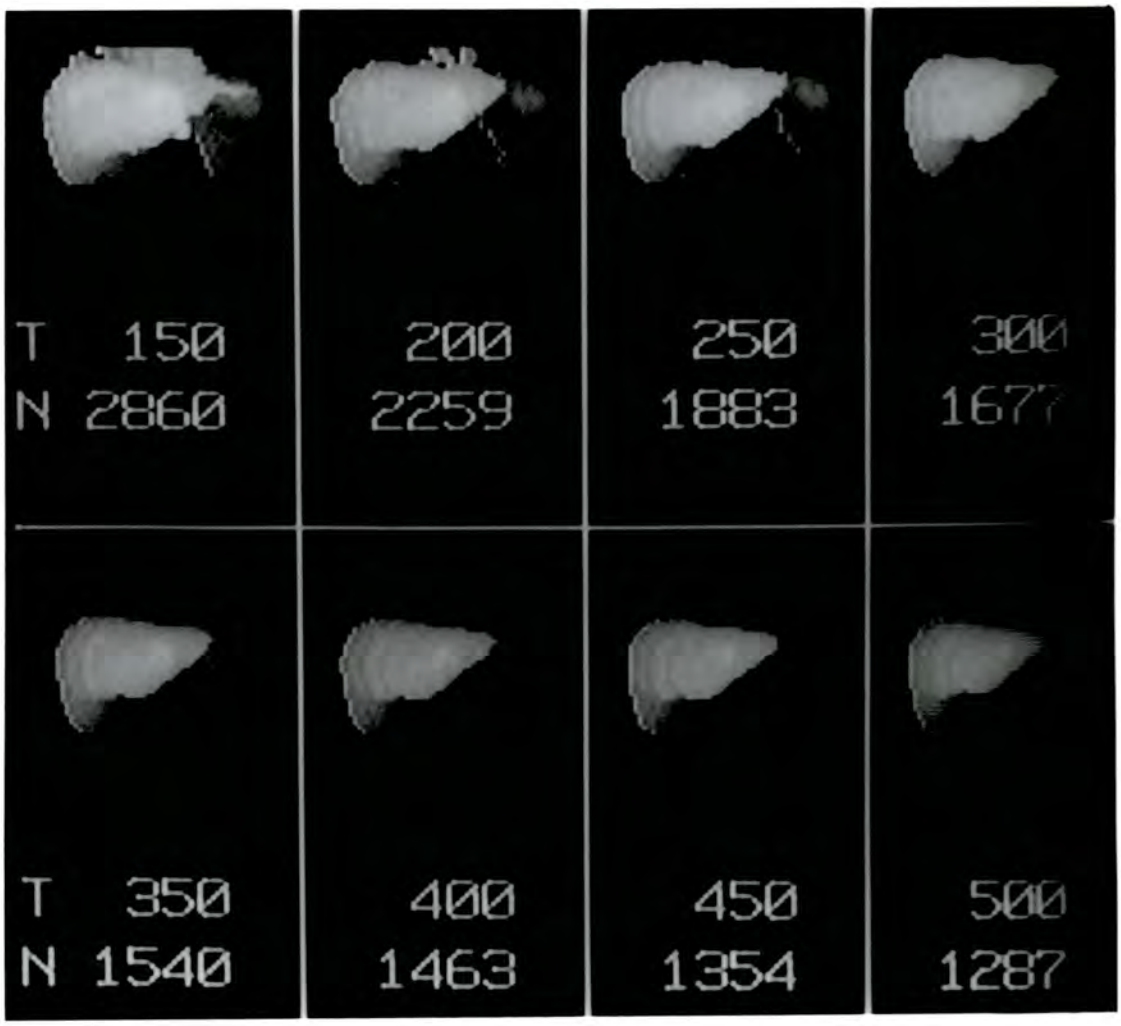


Figure 4.9 The appearance of the Liver and Spleen as the segmentation threshold was raised :-

T - Threshold level in counts per voxel.
N - Mean number of voxels in the corresponding Ordered Surface Lists.

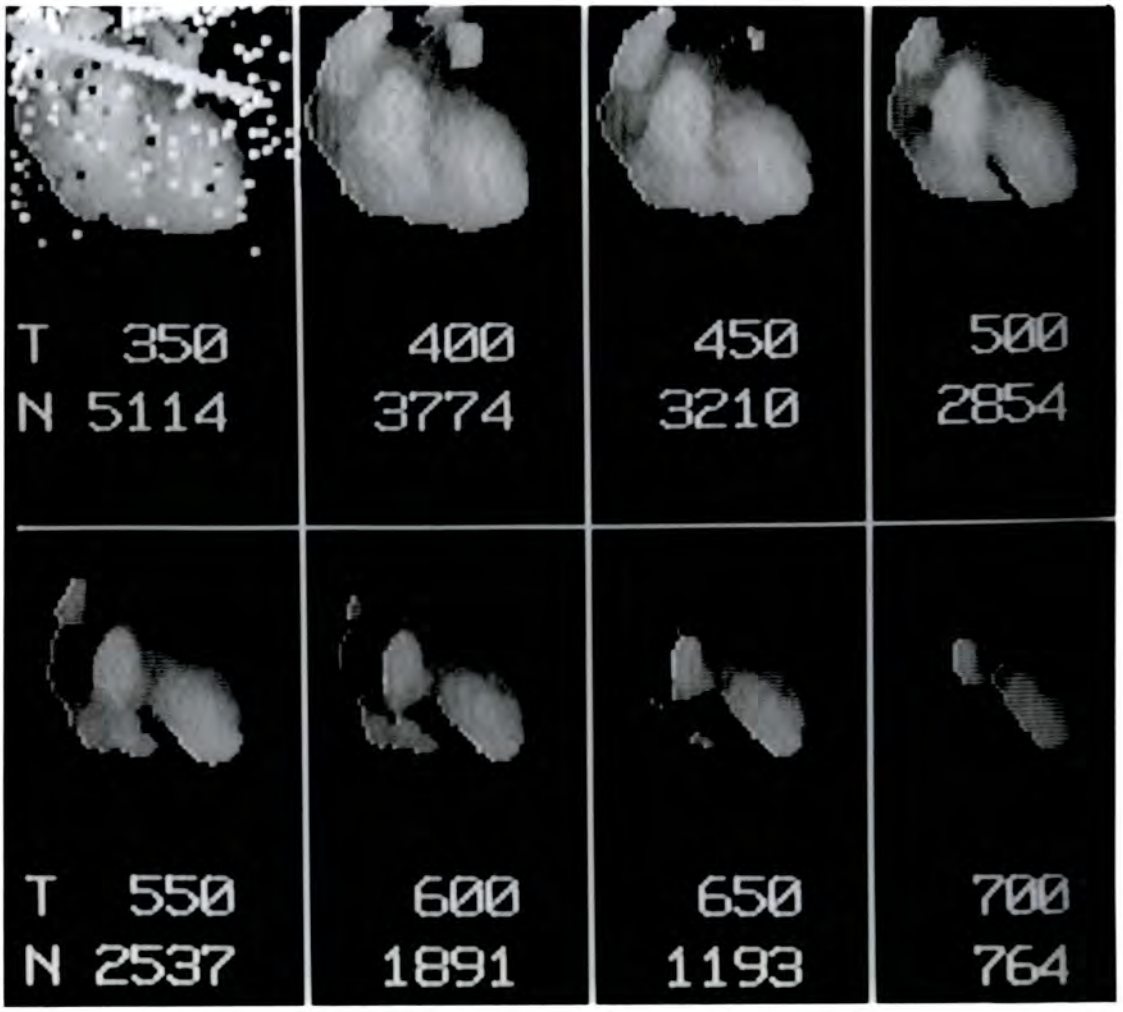


Figure 4.10 The appearance of the Cardiac Blood Pool as the segmentation threshold was increased :-

T - Threshold level in counts per voxel.
N - Mean number of voxels in the corresponding Ordered Surface Lists.

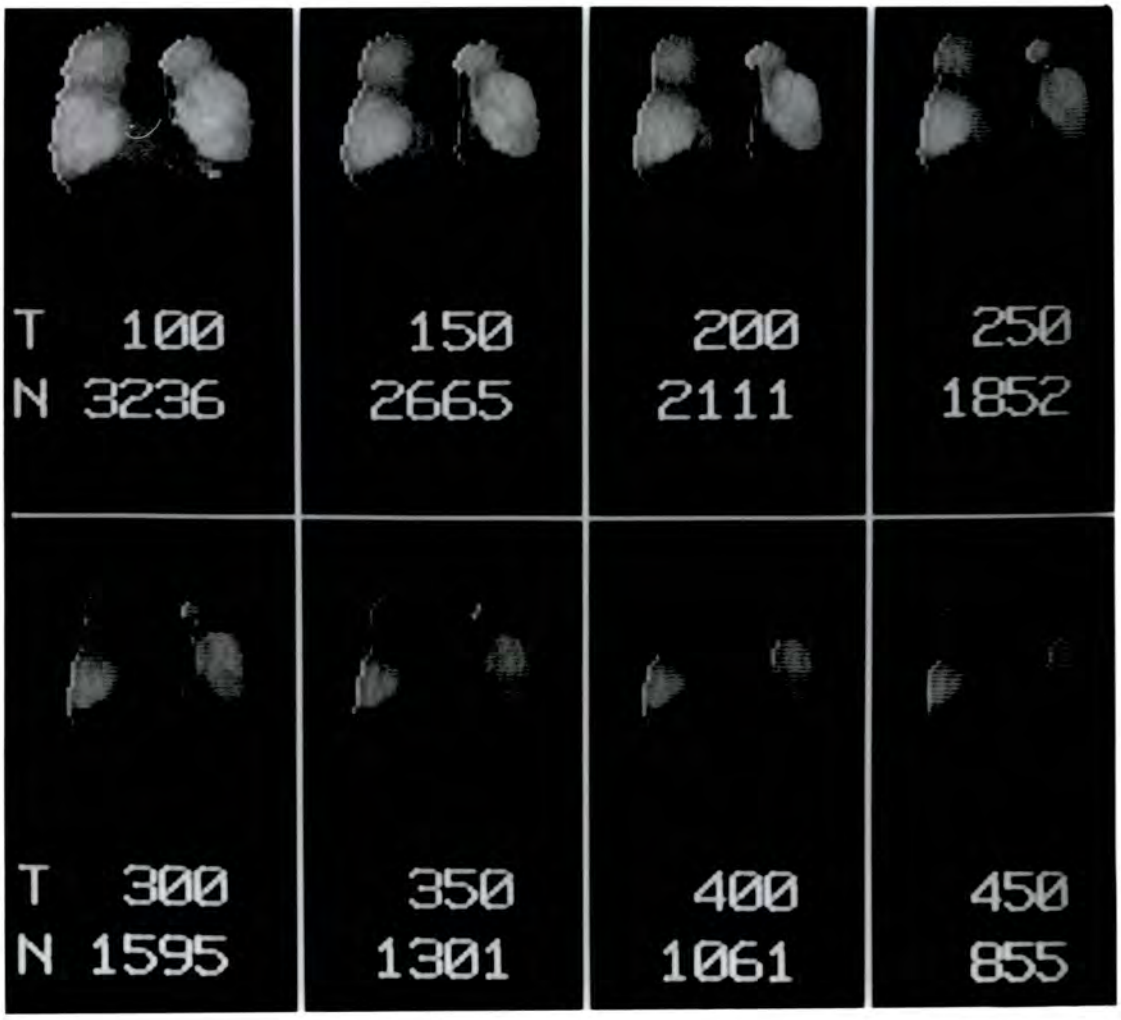


Figure 4.11 The appearance of the Perfused Lung as the segmentation threshold was increased :-

T - Threshold level in counts per voxel.
N - Mean number of voxels in the corresponding Ordered Surface Lists.

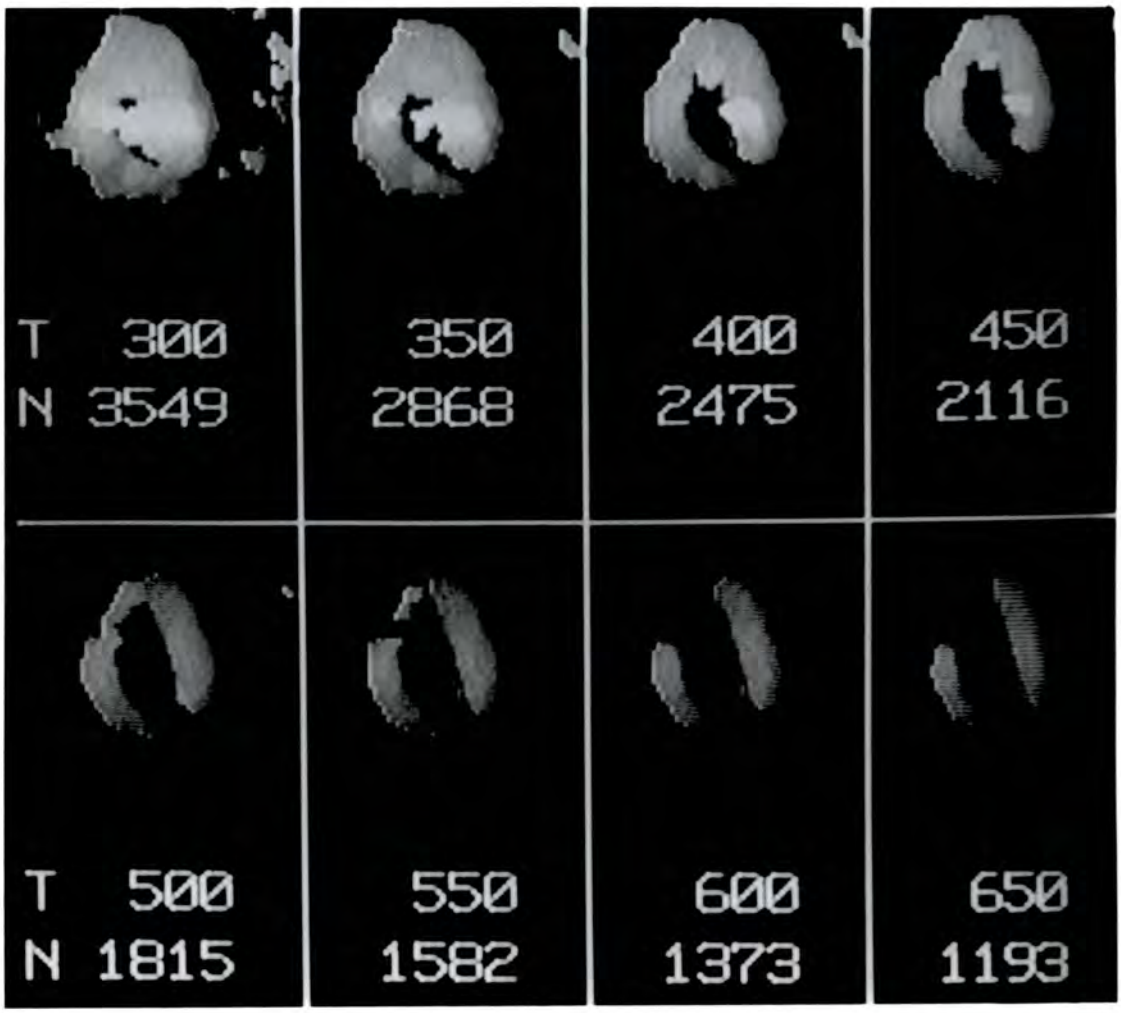


Figure 4.12 The appearance of the Myocardium as the segmentation threshold was increased :-

T - Threshold level in counts per voxel.
N - Mean number of voxels in the corresponding Ordered Surface Lists.

Table 4.1 Time required to segment several objects and create eight surface lists using the modified BTF algorithm

Object	No. of Slices	Time (s) using relative threshold				
		-100	-50	0	50	100
Myocardium	28	36.4	36.3	36.2	36.1	36.2
Liver	35	45.4	45.3	45.1	45.0	45.0
Lung	42	54.3	54.2	54.1	54.1	54.0
Blood Pool	50	67.1	66.5	66.1	65.8	65.4

function of threshold, expressed as counts per pixel relative to the optimum value. Although the lengths of the output lists are strongly dependent on the threshold, the time required to create them is not, decreasing only slightly as the threshold increases. Test files containing different numbers of slices were extracted from these clinical studies, and used to investigate the relationship between the number of slices in the input study and the pre-processing time required. Figure 4.13 shows a strong correlation between the number of slices processed and the mean time required to create all eight lists. Linear regression analysis was performed, and the estimated mean time per slice was 1.29 s. This implies a total pre-processing time of approximately 40 to 60 s for a typical object, which is certainly acceptable as a pre-processing stage.

The display stage (Algorithm 4.4) was evaluated using the eight test objects which were already segmented, and for which lists of visible voxels had been prepared. Figure 4.14 shows the average time taken to display each object from a nearly constant aspect (using a single list stored in main memory) and from all aspects (using one of eight lists read from disk as required). The latter case represents an accurate simulation of the way in which clinical objects would be examined. The average display time for the eight objects increased by only 2.2 % when multiple views were obtained, showing that disk access does not interfere significantly with the execution of the modified BTF algorithm. The results in Figure 4.14 also confirm that execution times for the modified BTF algorithm are significantly less than for the original algorithm.

There are, however, considerable differences in execution time for different objects. The form of Algorithm 4.4 suggests a linear

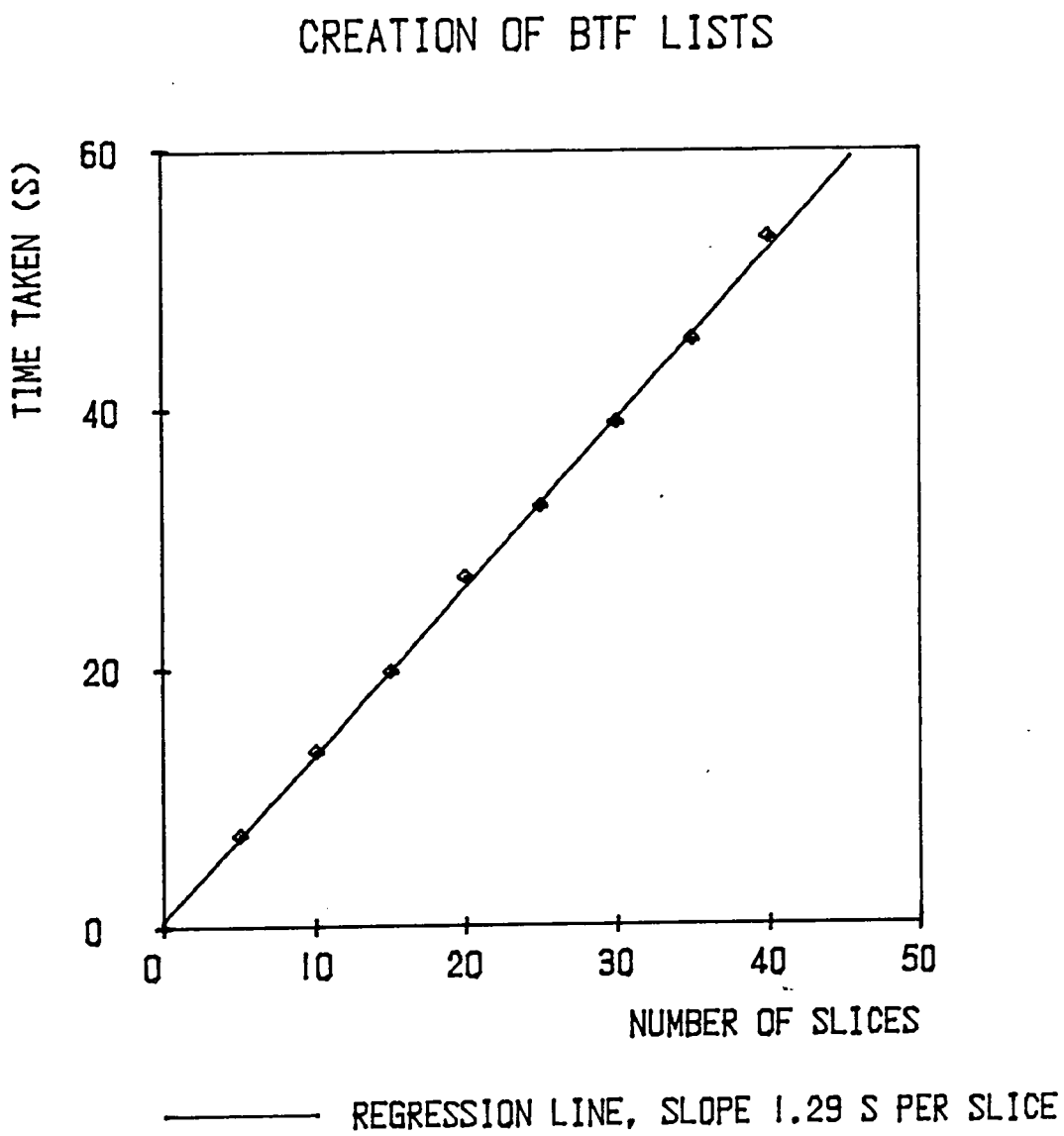


Figure 4.13 The linear relationship between the time taken to create the Ordered Surface Lists and the number of tomographic slices processed.

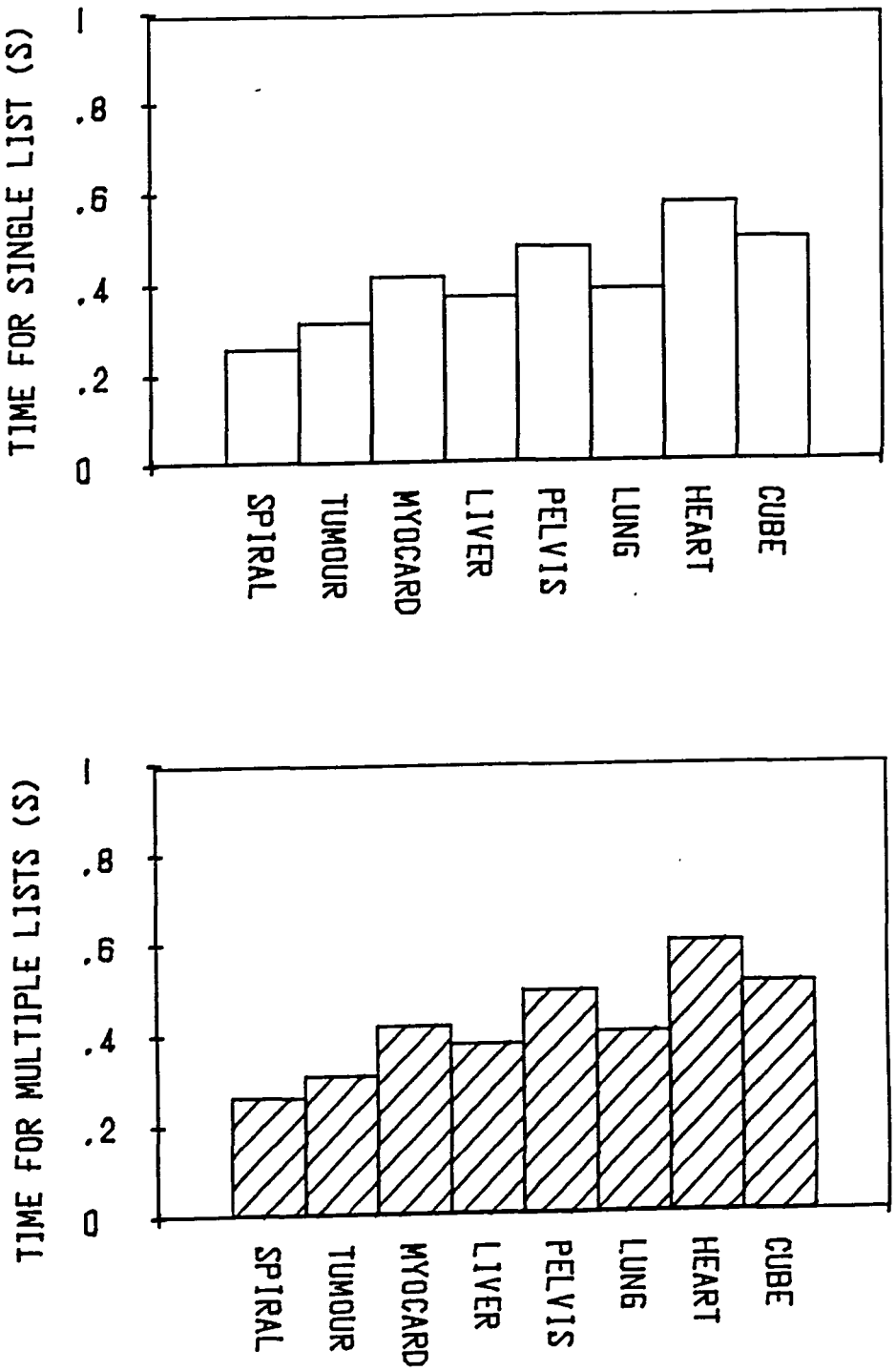


Figure 4.14 The time required to determine the visible surfaces of the test objects using the modified form of the BTF algorithm :-

Upper - Using a single Ordered Surface List.
Lower - Using multiple lists.

relationship between list length and execution time, and this is confirmed when the timing data in Figure 4.14 are plotted against the mean number of voxels in each list (Figure 4.15). Linear regression analysis was performed, and the estimated mean time per voxel processed was 0.12 ms, with an additional 0.2 s for transfer of the image to the display memory. This implies a total display time of approximately 0.4 s for a typical object, corresponding to a refresh rate of 2.5 Hz. For small objects the time taken to transfer the image to the display memory is a significant fraction of the total display time. An improved algorithm was devised to speed up this transfer, and the results are also shown in Figure 4.15. The slope of the regression line was identical to three significant figures, but the intercept has been reduced by approximately 0.04 s. For these objects hidden surface images can be displayed with refresh rates in the range 2.0 to 4.6 Hz.

The program which implemented the modified BTF algorithm incorporated two further techniques to assist the observer. The data in Figure 4.13 shows that the eight lists of potentially visible voxels are created from tomographic slices at a rate of 1.29 s per slice, or a total of 40 to 60 s for typical objects. However, a skilled observer can usually assess the optimum threshold from a single aspect. This means that only one out of eight lists needs to be created, typically requiring 5 to 7 s, for each choice of threshold. The program starts by displaying the surfaces which are visible from the anterior aspect, and the observer is allowed to vary the threshold until a satisfactory result has been obtained. All eight lists are then created using the chosen threshold, after which the observer is free to inspect the object from any angle. However, the choice of threshold is not irrevocable. By pressing a key the observer returns to the pre-processing stage, and

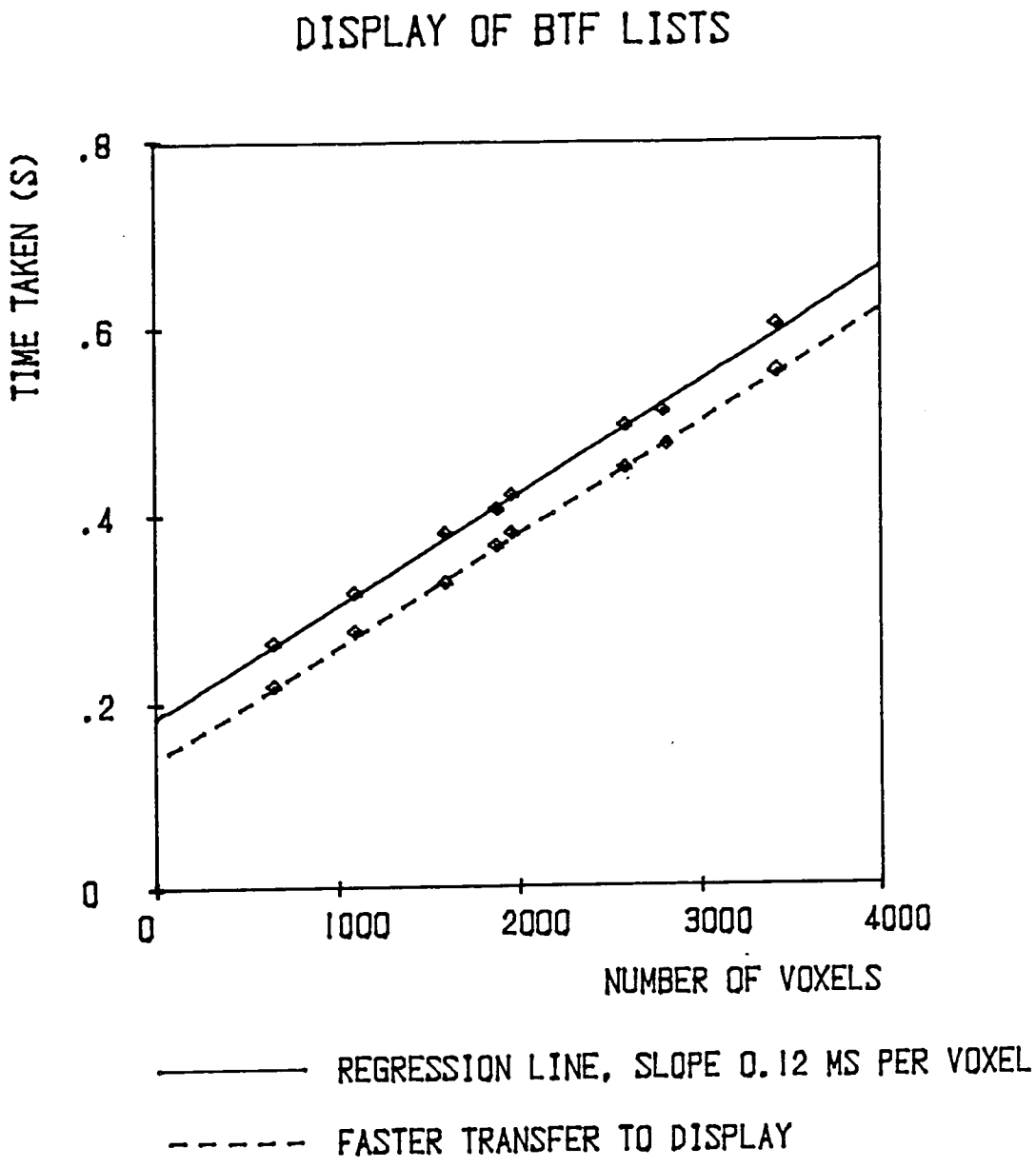


Figure 4.15 The linear relationship between the time taken to display the Ordered Surface Lists and the number of voxels processed.

can select a new threshold. Moreover, the single aspect which is presented for approval during the iterative selection of the new threshold is that which was being displayed when the key was pressed. Hence the observer can choose the optimum threshold by inspecting a different aspect, e.g. the left anterior oblique view of the cardiac blood pool.

The second technique concerned the interactive selection of viewing direction. Both the original and the modified BTF algorithms impose no restriction on the viewing transformations which may be used. However, the speed of the latter suggests that an interactive device should be used to control the viewing direction. A trackerball was used, together with the viewing transformation described in equation 4.12. The trackerball is normally used to determine the position of a cursor on the 256 x 256 display screen, and the current columnn and row coordinates were available to the program at any time as integers in the range 0 to 255. These values were used to control the parameters θ and ϕ respectively, so that horizontal motion of the trackerball varied θ, and vertical motion varied ϕ.

Because of the correspondence between trackerball motion and the rapid screen refresh rate, the observer can use the trackerball intuitively to alter the angle of view. The choice of which BTF list to use depends on the signs of the direction cosines of the current viewing direction. Using this viewing transformation the list to use can be determined from the angles θ and ϕ. Since these are in turn derived from the row and column coordinates of the trackerball, it follows that the choice of BTF list can also be determined by inspecting the trackerball coordinates. Figure 4.16 shows the correspondence between θ, ϕ, and the trackerball location, together with the zones which correspond to

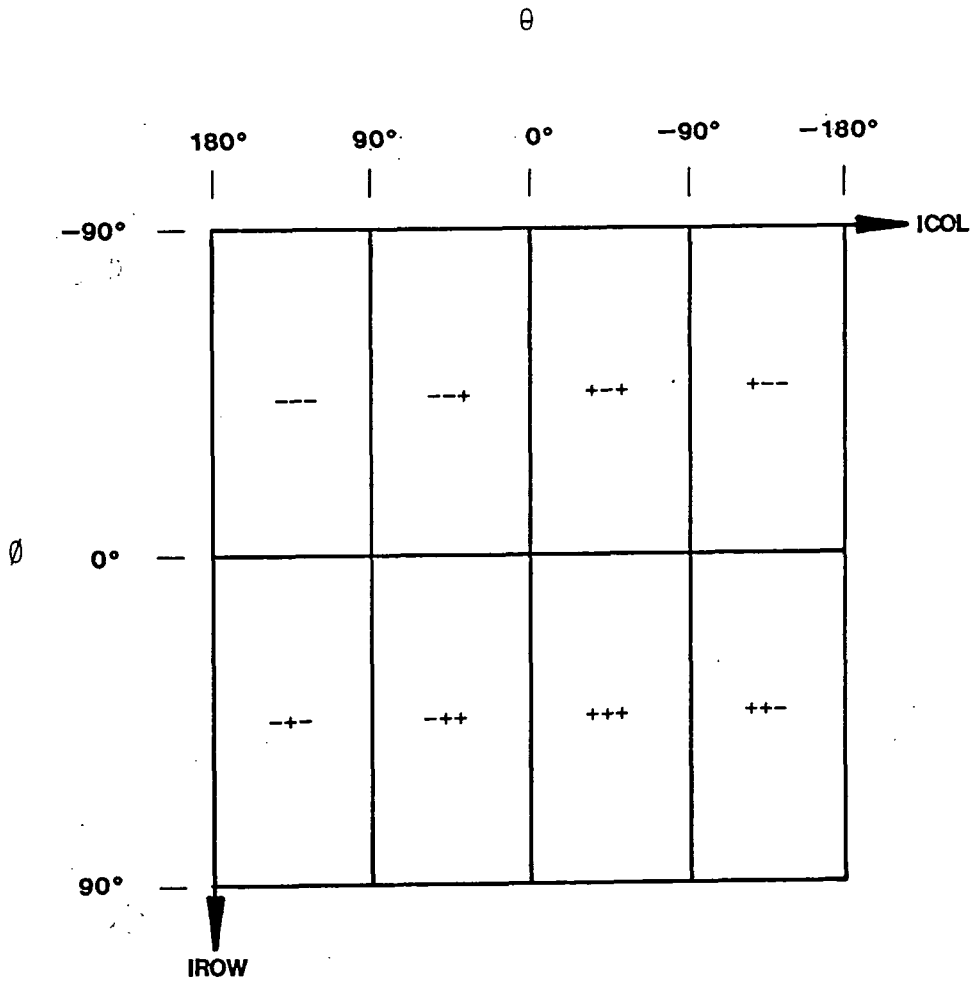


Figure 4.16 The relationship between θ, Ø and the tracker-ball coordinates. The range of both IROW and ICOL is 0 to 255. The eight marked zones correspond to the eight Ordered Surface Lists, and the signs of the direction cosines of the viewing direction are also shown for each zone.

the eight BTF lists and the signs of the direction cosines of the viewing direction. These are expressed relative to the (xyz) axes of the object coordinate system.

While the trackerball remains in the same zone the current list may be used to display the object from any aspect within that zone. Only when the trackerball moves into another zone must a new list be read from disk. Although the same approach could be used with any two dimensional device, a trackerball is ideal, since it combines fine control for delicate adjustment of angle, with rapid movement to any location for inspection of different aspects.

4.7 Conclusions

The choice of object representation and of display technique is not independent, since the performance of the latter depends on the former. In the case of the modified BTF algorithm, it is the display technique itself which dictates the form of the Ordered Surface List object representation. However, a comparison can be made between the display techniques described above, by applying them to the same data using the optimum object representation for each display technique. Figure 4.17 shows the average time taken to display the same eight objects from several different aspects using three display techniques. This comparison does not take into account the pre-processing time required to create any of the specialised object representations used.

The time required for Direct Display of a Binary Object array is essentially independent of object size within the bounds of the array, but would be expected to increase $O(N^3)$ if finer sampling necessitated a larger array size. Display times are fast enough to permit interactive

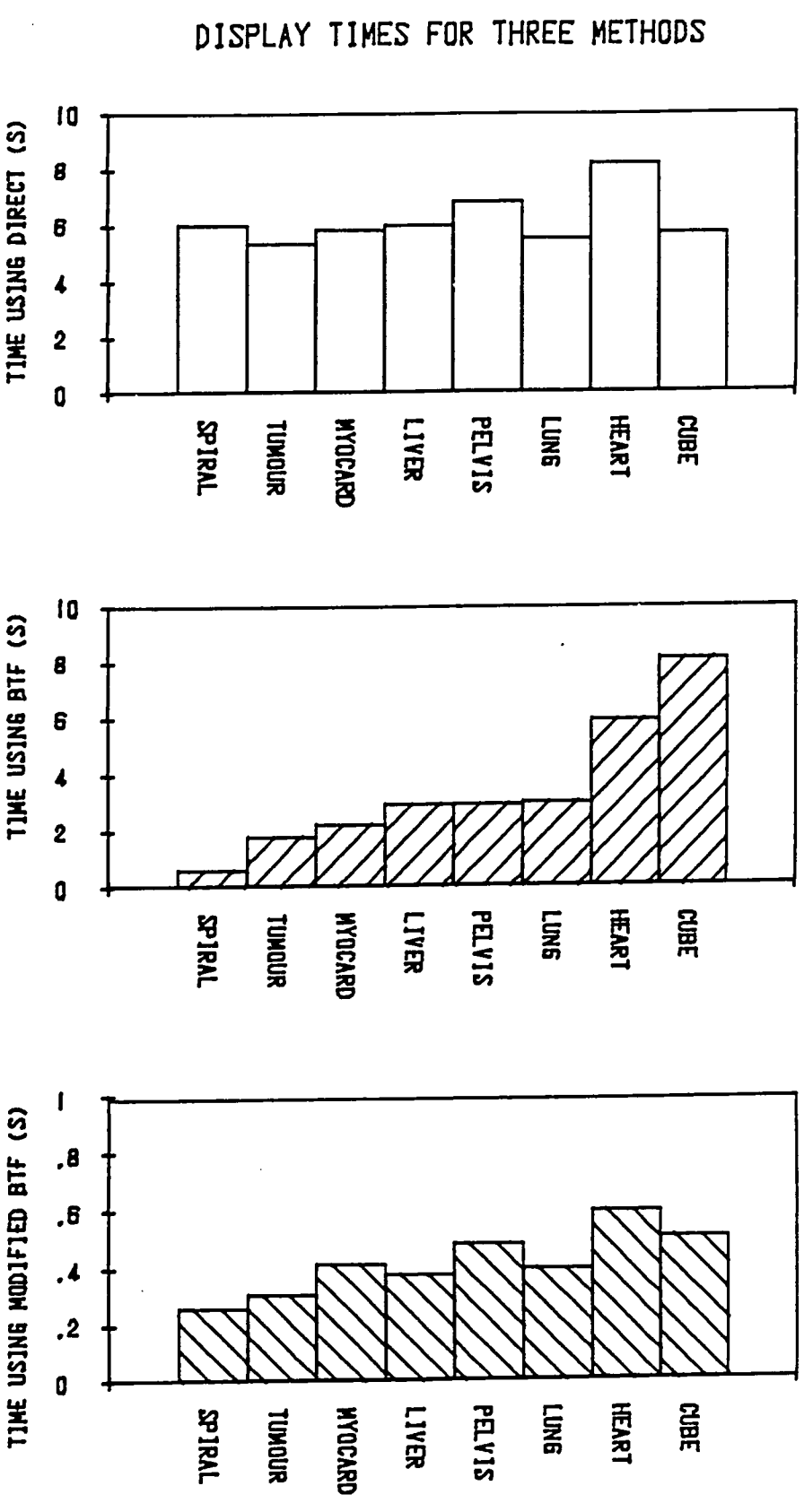


Figure 4.17 The mean time required to determine the visible voxels of the test objects from various aspects, using three techniques :-

Upper - Direct Display of Binary Objects.
Middle - BTF display of segmented lists of voxels.
Lower - Modified BTF display of Ordered Surface Lists.

display, but not real time or kinetic display. However, this technique does enable the observer to re-display an object from the same aspect very rapidly, thereby facilitating some forms of object manipulation.

The BTF algorithm applied to a segmented list of voxels executes in times which are nearly proportional to the object volume, and which will increase $O(N^3)$ for larger arrays. Display times are generally faster than for Direct Display of a Binary Object, but without the ability to re-display rapidly at the same aspect. The modified BTF algorithm applied to Ordered Surface Lists was found to be an order of magnitude faster than either of the other two techniques. Because this is a surface representation, the display time increases only slowly with the size of the object. Figure 4.18 shows the timing data from Figure 4.17 plotted as a function of the object size, expressed as the number of voxels. Constant, linear and power law functions were fitted to the data for the Direct Display, BTF and modified BTF techniques respectively. The agreement between the fitted curves and the data is good, and suggests that the execution time for the modified BTF algorithm will increase only $O(N^2)$ for larger arrays.

The modified BTF algorithms developed were therefore implemented on a more modern nuclear medicine imaging system, with a higher clock rate and a faster bus for transfer to the display memory. Using this system the modified BTF algorithm displayed hidden surface images at a rate of 0.08 ms per voxel, giving frame rates of 2.8 to 8 Hz for the test objects. Such display rates are fast enough to produce smooth object rotation, and have particular value for the display of time varying images of the heart (Gibson 1988).

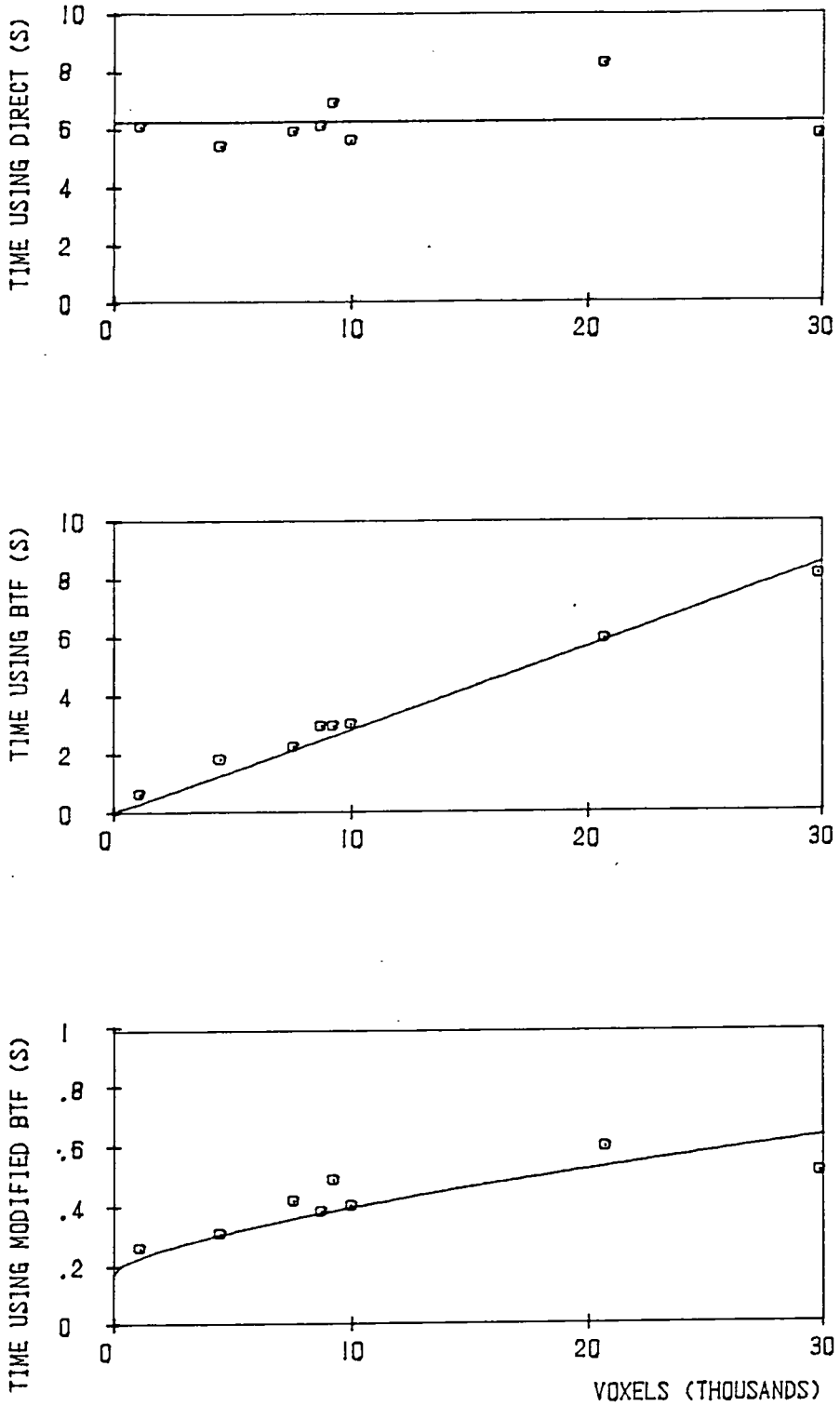


Figure 4.18 The same data as Figure 4.17, plotted as functions of the object size, with constant, linear and power law functions fitted to the data.

Upper - Direct display of Binary Objects,
Time = 6160 (msec)
Middle - BTF display of a list of voxels,
Time = 0.29 [No. of voxels] (msec)
Lower - Modified BTF display of Ordered Surface Lists,
Time = 0.47 [No. of voxels] $^{2/3}$ + 180 (msec)

5. Shading Techniques

5.1 Illumination and Reflection Models

The algorithms described in Section 4 are solutions to the hidden surface problem, and for any observer position and viewing direction they will identify those parts of the object which are visible. This information may take the form of a list of vertices of visible facets (Cook et al 1983), or more usually a two dimensional array in which pixel values specify the z' coordinate of the voxel visible at that point in the image (Gibson 1983, Chen et al 1985b). Such an array is known as a 'depth array', and has been described as a $2\frac{1}{2}$-dimensional model of the object, since it gives partial information about the three dimensional shape of the object, in one direction only. Shading techniques can be applied to this visible voxel data to create an image which will simulate the appearance of a solid object. Whereas the determination of visible voxels is an entirely objective process, surface rendering by means of a shading technique has a significant subjective component. The purpose of any shading algorithm is to convey information to the observer about the three dimensional morphology of the object. Thus, although surface shading techniques are usually based on physical models of illumination and reflection, they need not be bound by physical limitations of the model, nor by the need for complete fidelity to optical processes, providing that a convincing representation of the object is obtained.

A simple model for surface shading was developed for this work and is shown in Figure 5.1. Parallel light is incident along the direction specified by the unit vector **i**. The surface normal vector at the point of incidence is **s**, and the observer has a viewing direction specified by

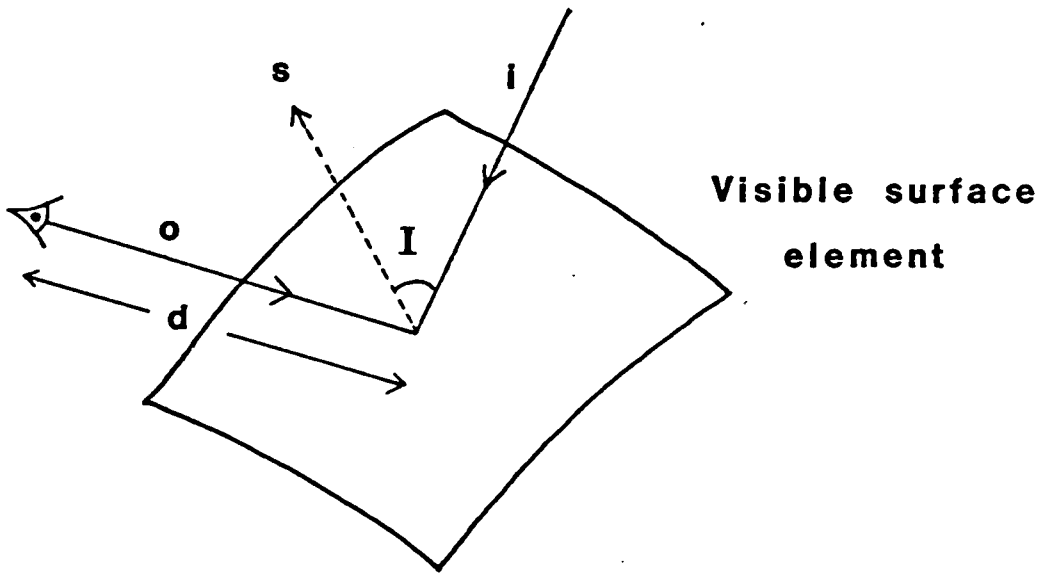


Figure 5.1 Terms used in the shading model :-

i - incident light vector.
s - surface normal vector.
o - viewing direction vector.
d - distance to the observer.
I - angle of incidence.

the unit vector **o**. There is also assumed to be some ambient isotropic illumination which falls uniformly on all surfaces of the object irrespective of their orientation. The relative intensities of directional and isotropic illumination, and the parameters illustrated in Figure 5.1, specify the intensity of light reflected to the observer from points on the surface. Let this intensity of reflected light be the 'shade' to assign to the surface, then a general expression for calculating shade values can be written as :-

$$S = I_s.f(d) + D_r.g(d).h(\mathbf{i},\mathbf{s},\mathbf{o}) \qquad 5.1$$

where S is the calculated shade value for the surface element under consideration, I_s is the intensity of isotropic illumination and D_r is the intensity of directional illumination (all in arbitrary units); f(d) and g(d) are dimensionless functions which describe the attenuation of reflected light intensity with distance from the observer d; and h(**i**,**s**,**o**) is a dimensionless function describing the variation of reflected light intensity with the angles of incidence and reflection. Because the isotropic illumination falls equally on surfaces at all orientations, the corresponding contribution to the shade value is independent of the vectors **i**, **s** and **o**.

The range of values for f, g and h is 0 to 1, hence shade values lie in the range 0 to $(I_s + D_r)$. By choosing appropriate values of I_s and D_r, shade values can be constrained to lie in the range 0 to 255, suitable for direct transfer to a typical image memory with eight bits per pixel. The selection of functional forms for f, g and h will determine the type of reflection modelled, and hence the appearance of the surface. For a perfectly specular reflector, h(**i**,**s**,**o**) will be zero

unless **i**, **s** and **o** are co-planar, and **i.s** = **o.s**. Such a surface would be very difficult to interpret if illuminated by a single parallel beam of light. Medical objects have generally irregular surfaces, and only a few scattered points would have the exact surface orientation to reflect light towards the observer. A more useful model is that of the diffuse opaque reflector, for which Lambert's Law of diffuse reflection holds. In this case the light reaching the observer is independent of **o**, and depends only on the cosine of the angle of incidence:-

$$h(\mathbf{i},\mathbf{s},\mathbf{o}) = -\,\mathbf{i}.\mathbf{s} \qquad 5.2$$

However, empirical modifications to this model have been proposed, designed to reduce the rate of variation of shade with the angle of incidence. A typical example is:-

$$h(\mathbf{i},\mathbf{s},\mathbf{o}) = [\cos(I/M)]^{p} \qquad 5.3$$

where cos I = - **i.s**, and parameters p and M are chosen such that $p < 1$ and $M > 1$ (Chen et al 1984).

This modification achieves the desired aim, of reducing the rate of variation of shade, but is not without disadvantages. Functions which are based on cosines have potentially negative values, which correspond physically to the moment when the angle of incidence exceeds 90^{o}, and the light source moves behind the surface. For such positions $h(\mathbf{i},\mathbf{s},\mathbf{o})$ must be set to zero, rather than a negative value, since no light can be reflected from the surface. From equation 5.1, when $h(\mathbf{i},\mathbf{s},\mathbf{o})$ is zero the shade is determined by the isotropic component only. Therefore, to avoid shading discontinuities, $h(\mathbf{i},\mathbf{s},\mathbf{o})$ must tend to zero as I tends to

90^{o}. Figure 5.2 shows values of h(**i**,**s**,**o**) calculated using equation 5.3 for several values of p and M. As can be seen, marked shading discontinuities will be present for any function with M > 1. The only way to avoid such discontinuities is to ensure that the angle of incidence I is never larger than 90^{o}, and this can be accomplished by restricting the direction of illumination to lie parallel to the z' axis, i.e. along the observer's viewing direction. All visible surfaces will now have angles of incidence less than 90^{o}, since otherwise they would point away from the observer and so be invisible. Although this angle of illumination enables modified cosine functions to be used with M>1, it is less realistic than an oblique angle, since in the natural world the predominant illumination is usually from above.

The choice of functions f(d) and g(d) is also empirical. An inverse square law expression would simulate the actual diminuition of reflected light intensity with distance. In practice, however, a simple linear function is generally used (Vannier et al 1983). If the closest point to the observer has depth (z' coordinate) d_0 and the furthest point has depth d_1, then a suitable form for both f(d) and g(d) is:-

$$f(d) = g(d) = \frac{d_1 - d}{d_1 - d_0} \qquad 5.4$$

This expression will range from 1 to 0 as the depth increases from the front of the object to the back. Strictly this expression would require new values of d_0 and d_1 to be calculated for a new object orientation. In practice fixed values are chosen which will not be exceeded in either direction for all orientations. Substituting for f, g and h into equation 5.1 we can express the shade value as:-

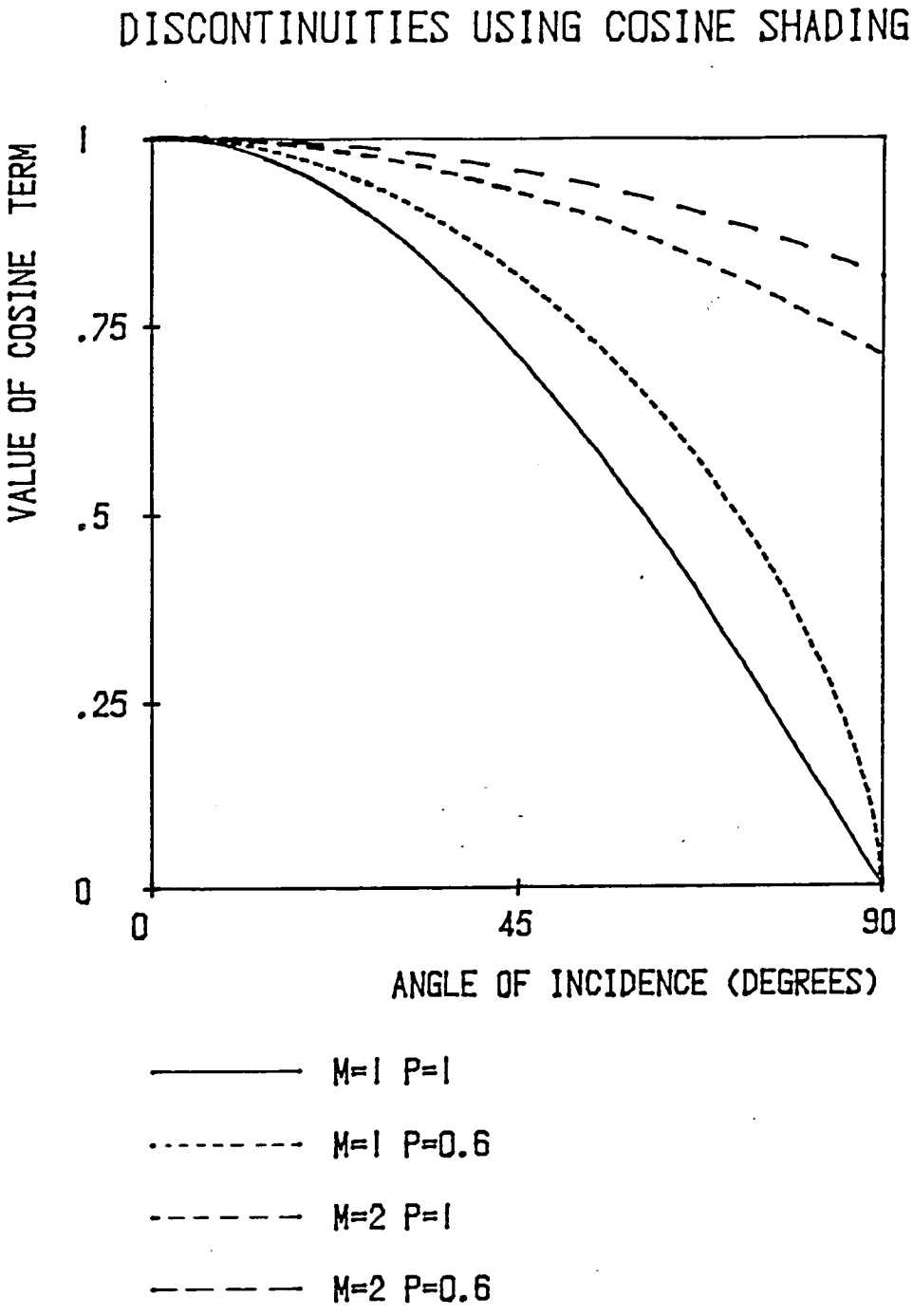


Figure 5.2 Shading discontinuities using cosine terms with values of M greater than one.

$$S = I_s \cdot \frac{d_1 - d}{d_1 - d_o} + D_r \cdot \frac{d_1 - d}{d_1 - d_o} \cdot [\cos(I/M)]^p \qquad 5.5$$

The parameters I_s, D_r, d_1, d_o, p and M are fixed for a given viewing direction and illumination model. However, d and I have to be estimated for all points on the visible surface. Values of d will be available as z' coordinates of the visible voxels, but the estimation of I is less straightforward, since it depends on the surface normal vector **s**. Surface orientation is a powerful visual cue to object shape, and is therefore an important component of a shading model. Several techniques have been developed for determining the surface normal and these are assessed below.

5.2 Estimation of Surface Normals

The use of an object representation scheme which is based on triangular surface facets simplifies the process of estimating surface normal vectors **s**. Each facet is specified by three vertices, from whose coordinates an exact surface normal vector can be calculated (Newman and Sproull 1973). It is preferable to define and fix the illumination direction in the observer's coordinate system, rather than in the object's coordinate system. This ensures that the object will always be adequately illuminated, irrespective of the chosen viewing direction. This is analogous to having a 'miner's lamp' attached to the observer, always pointing in the same orientation relative to the viewing direction. For a triangular facet model, therefore, the surface normal vector **s** is calculated from the coordinates of the three vertices after transforming into the observer's coordinate system.

It is much more difficult, however, to calculate surface normals for cuberille representation schemes. As has been pointed out in Section 4.3, a cubic voxel can have either one, two or three visible faces, each of which has an exact surface normal pointing along one of only three orthogonal directions. If shade values are calculated using these exact surface normal vectors, then there are only three possible values for the angle of incidence. The resulting image is shown in Figure 5.3, with each voxel shaded independently. The overall shape can be determined, but any small surface variations are obscured by the discontinuous changes in surface normal (Herman and Liu 1979).

When using using cuberille representations it is therefore advisable to use approximate methods for estimating surface normals, in an attempt to convey local surface orientation without distracting discontinuities. Methods developed for this work, and those proposed by other workers, may be summarised as follows :-

Depth shading: If the intensity of directional light is set to zero, then equation 5.5 reduces to,

$$S = I_s . \frac{d_1 - d}{d_1 - d_o} \qquad 5.6$$

For this simple shading model the surface normal s is not required. It is therefore particularly useful for the display of irregular surfaces, and has been extensively applied to the display of bone images from CT data (Vannier et al 1984). Display techniques based on the BTF display algorithm (Section 4.5) produce a depth image containing values of d for all pixels in the image. These can be transformed into shade values using equation 5.6, or if a display with a programmable video

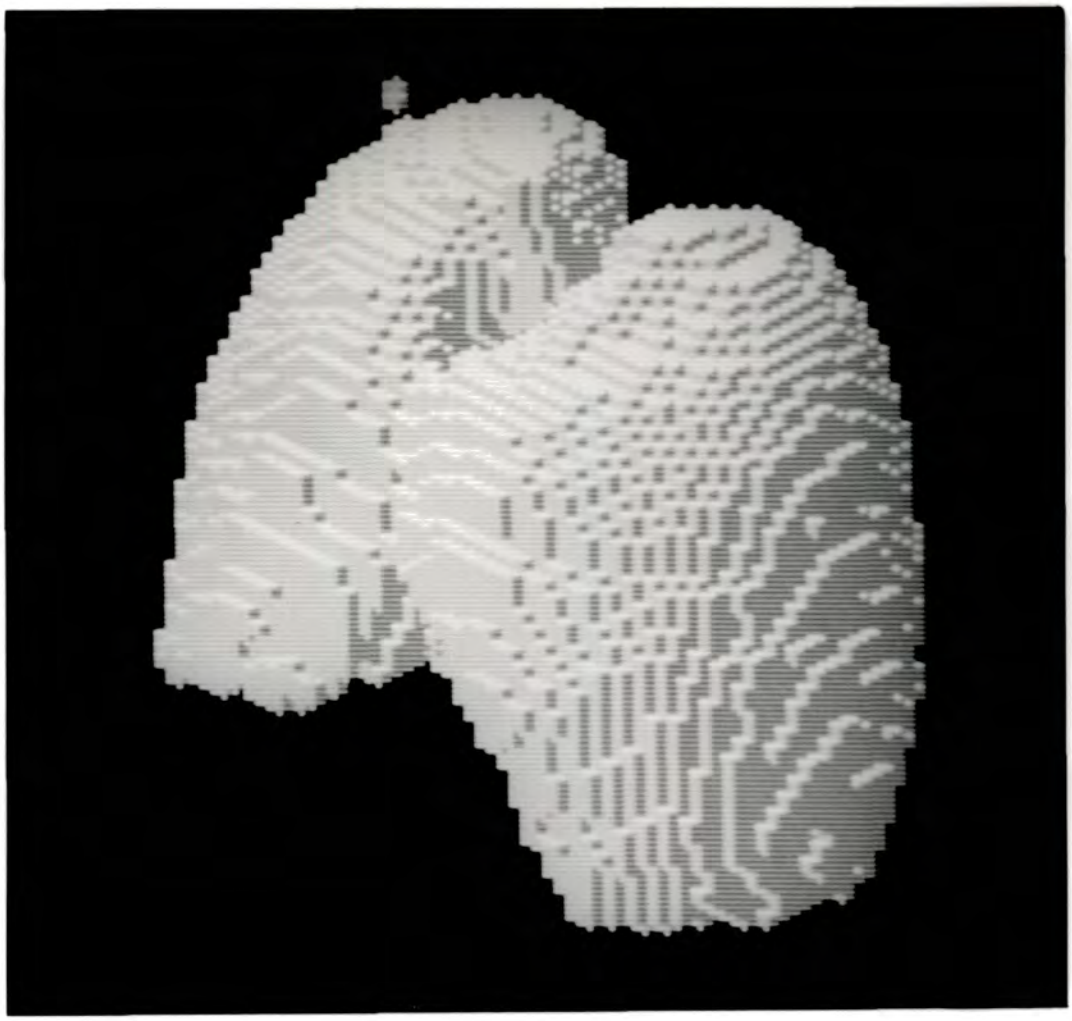


Figure 5.3 A cuberille model of the lungs with individual voxels shaded independently. Only three shade values are possible, depending on the orientation of the three visible facets of each cubic voxel.

look-up table is available the raw depth array may be transferred directly to the display memory. In the latter case the transformation from depth values to shade values is performed at video refresh rates using a look-up table calculated from equation 5.6. In either case this shading model requires little computation, and is ideally suited to rapid display methods (Gibson 1986b, Gibson 1987).

Object space shading: Rather than using the exact surface normal to calculate the shade to assign to each face, an averaged or smoothed value can be used, taking into account the local object orientation (Herman and Udupa 1981). This averaging can be performed in two ways. In the first method, the function $h(\mathbf{i},\mathbf{s},\mathbf{o})$ is evaluated using equation 5.3 for both the face under consideration, and for the four adjacent faces which share an edge. The value used to calculate the shade is the weighted sum of all five values of $h(\mathbf{i},\mathbf{s},\mathbf{o})$. The form of the weights used is chosen to minimise artificial discontinuities between faces. Because each adjacent face can have only three orientations relative to the central face, there are only 81 possible combinations of relative surface orientations, and hence only a limited number of possible shades. It is possible to store an extra byte of object information with each cuberille facet, to indicate which of the 81 local surface conformations applies. This information can then be used as an index into a pre-computed look-up table, in which average values of $h(\mathbf{i},\mathbf{s},\mathbf{o})$ are stored. Unfortunately, the weights used in the averaging process depend on the relative orientations of the faces to the direction of illumination, and so the shades assigned may vary discontinuously as the object is rotated.

To avoid this problem, a smoothing method based on the face

normals, rather than on individual values of h(**i**,**s**,**o**) has been implemented (Chen et al 1984). An average local surface normal **s'** is estimated from a weighted sum of the true surface normals of the current face and four adjacent faces. This average value is then used to calculate an apparent angle of incidence, and hence a shade value. Because the averaged surface normal is now fixed relative to the object, the shade values are invariant under rotation. However, just as for the first method, there are only a limited number of values for **s'**. This means that although the estimated surface normal can be coded by a single byte, with subsequent retrieval of values for h(**i**,**s**,**o**) from a look-up table, there are only a limited number of shades which can be produced.

Observer space shading: The need for additional data storage to contain surface normal vectors can be avoided if the surface normal vector can be estimated from the depth image (Gibson 1983, Gibson 1986a). This technique was developed for this work, and uses information about the object after it has been transformed into observer coordinates, and after the visible surfaces have been determined. A similar technique has been independently described as 'gradient shading', since it depends on estimating surface normals from the local gradient vectors of the depth array (Gordon and Reynolds 1983). No additional data needs to be stored, since the vectors are estimated from the visible voxel depth array.

Figure 5.4 shows a pixel in the depth array and its neighbours. It is intuitively apparent that several adjacent depth values convey information about the local surface orientation. For the case of a continuous analytical surface the surface normal may be calculated exactly using surface gradients. Let the surface be described by a

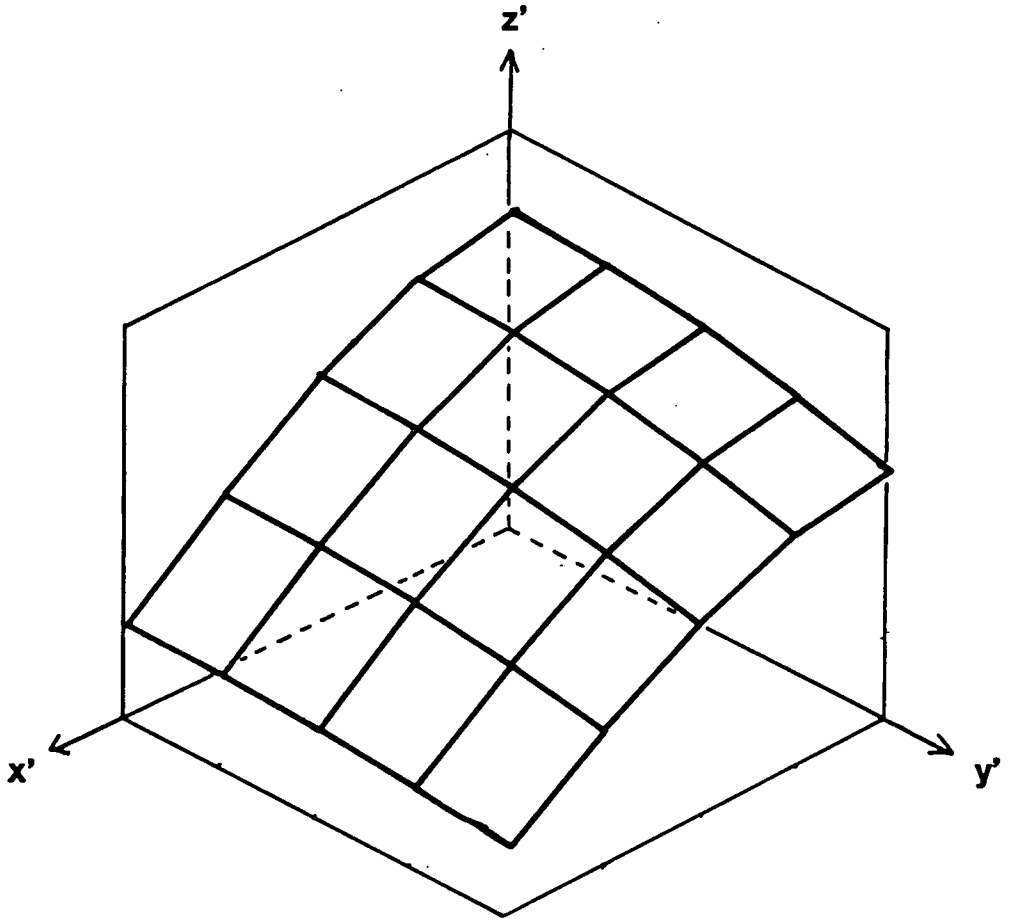


Figure 5.4 A pixel at location (x'y') in the depth array and it's nearest neighbours. Several adjacent pixels provide sufficient information to estimate the local surface orientation.

function $F(x',y')$, i.e. the depth is a single valued function of the observer space coordinates x' and y'. Then define the function $G(x',y',z')$ by:-

$$G(x',y',z') = z' - F(x',y') = 0 \qquad 5.7$$

For any such function $G(x',y',z')$ the unnormalised surface normal vector $\mathbf{S}$ is given by:-

$$\mathbf{S} = \nabla G \qquad 5.8$$

where ∇ is the vector differential operator. Substituting 5.7 into 5.8,

$$\mathbf{S} = \left(-\frac{\partial F}{\partial x'}, -\frac{\partial F}{\partial y'}, 1 \right) \qquad 5.9$$

Equation 5.9 shows that the components of the unnormalised surface normal vector can be obtained by differentiating the surface function with respect to x' and y'. The unit normal vector $\mathbf{s}$ (and hence the shade value) can be obtained from $\mathbf{S}$ by imposing the conventional normalisation condition:-

$$\mathbf{s} = \left(-1.\frac{\partial F}{L\,\partial x'}, -1.\frac{\partial F}{L\,\partial y'}, \frac{1}{L} \right) \qquad 5.10$$

where $L^2 = \left(\frac{\partial F}{\partial x'}\right)^2 + \left(\frac{\partial F}{\partial y'}\right)^2 + 1$

In practice the surface is both discontinuous and discrete. It is formed by a set of sampled points corresponding to the pixels of the

depth image, and there will be discontinuities present, both at the edges of objects and where different portions of the same object overlap for a given viewing direction. Gradient estimation techniques have been developed to solve these problems.

5.3 Gradient Estimators

Three techniques for estimating local gradients in depth images have been compared. Two were developed for this work (Simple forward/backward differences and Polynomial shading), while the third was developed and published independently (Weighted forward/backward differences). These techniques can be summarised as follows :-

Simple forward/backward differences: Using this approach gradients in the x' and y' directions are estimated from the average of the forward and backward differences between the z' coordinates of the central pixel and those of the two nearest neighbours along the x' and y' axes (Gibson 1983). At a discontinuity, the difference between depth values will be large, and so one or both gradients will also be large. This will produce a surface normal vector lying almost in the (x'y') plane, i.e. 'rounding off' the edges of objects.

Weighted forward/backward differences: The same approach is used, but the weighted average is used rather than the simple average, and the weights are themselves functions of the absolute values of the differences in z' coordinates (Chen et al 1984). The weights are chosen so that very large values are produced when the absolute value of the differences exceeds an empirical limit. As for the simpler case, the effect is to round off the edges of objects, but greater flexibility is obtained in the definition of edges by varying the empirical limits on the weighting values.

Polynomial surface shading: Each gradient is calculated from the parameters of a polynomial fitted to the z' values in the neighbourhood of the pixel in question, provided that they form a locally connected surface (Gibson 1986a). The algorithm is essentially one dimensional, and the same procedure is applied to find the gradients along both the x' and y' axes independently. Wherever possible the algorithm attempts to fit a quadratic function to five points centred on the pixel in question, using a least squares criterion. However, the five points must be connected, in the sense that there are no discontinuous changes in the z' coordinate greater than an empirical threshold. If this is not the case, an alternative set of five adjacent points containing the current pixel is used. If no such set of five connected points is present, a linear function is fitted to three points containing the current pixel. The sequence in which fits are attempted is shown in Figure 5.5. This technique can provide smooth shading up to the edges of objects, without attempting to average between points on opposite sides of a discontinuity.

The computational requirements for Polynomial surface shading are less onerous than appears at first sight. There are five possible sets of five points, and three sets of three points, and in all cases the sampling interval along the x' or y' axes is constant. For equally spaced data the fitting of a polynomial can always be accomplished using a one dimensional convolution of the data with a fixed kernel (Savitzky and Golay 1964). For determining the gradient in the x' direction, the polynomial to be fitted has the equation:-

$$F(X) = a_0 + a_1.X + a_2.X^2 \qquad 5.11$$

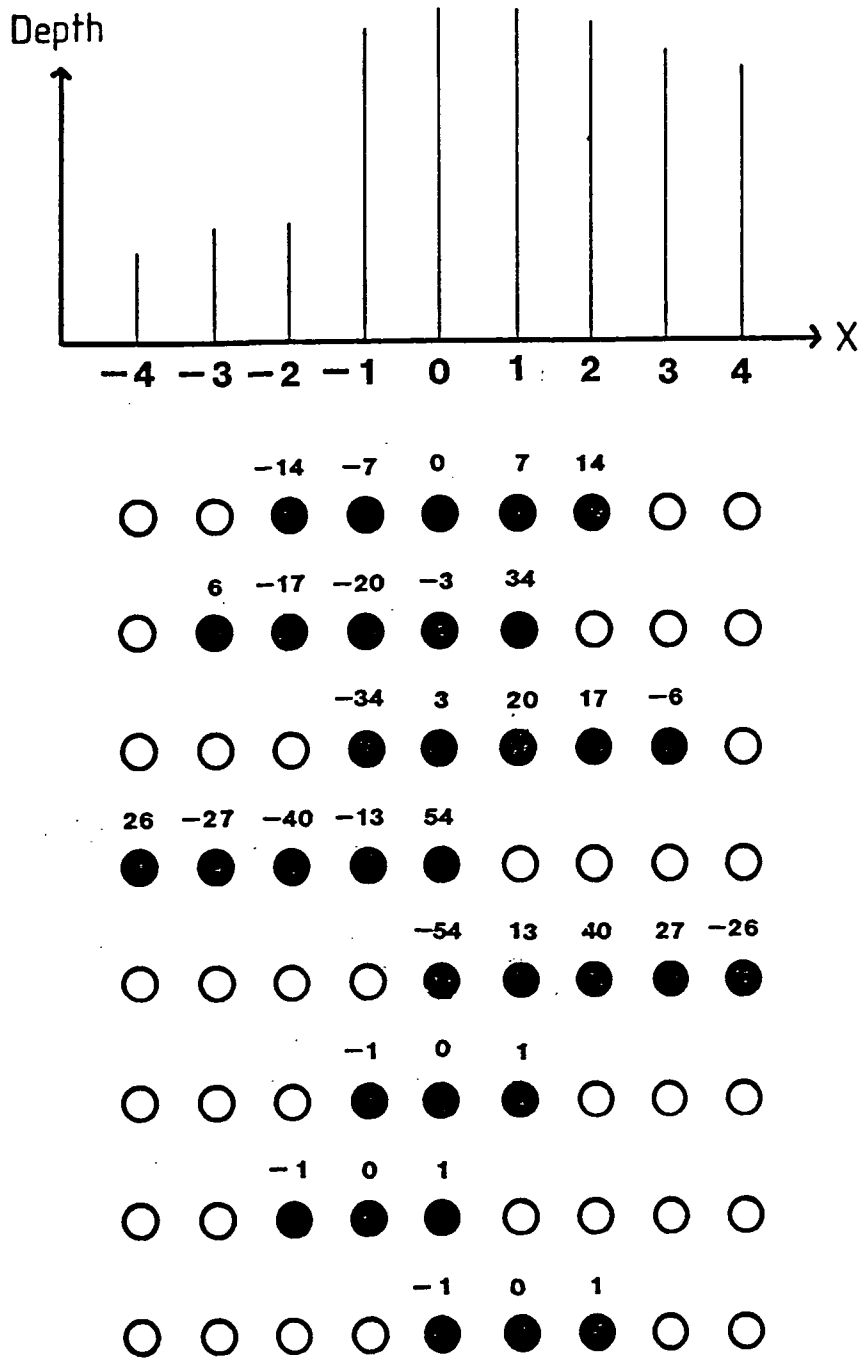


Figure 5.5 A typical section through a depth array, showing a discontinuity at X=-2. The solid circles indicate the sets of adjacent pixels which are considered by the polynomial gradient algorithm, first trying to fit a quadratic function using 5 points and then a linear function using 3 points.

In the case illustrated, the third quadratic function would be fitted, since this uses five 'connected' pixels which do not span the discontinuity.

Weighting factors for the convolution are also shown, with a denominator of 70 for the quadratic functions and 2 for the linear functions.

where X is a dummy variable with origin at the current pixel. The gradient along the X axis at that point is,

$$\frac{\partial F}{\partial X} = \frac{\partial F}{\partial x'} = a_1 \qquad 5.12$$

Hence only one parameter of the polynomial needs to be estimated to obtain the gradient. This parameter can always be estimated using a five point convolution (for a quadratic function) or a three point convolution (for a linear function). Figure 5.5 shows the convolution kernels for each of the five quadratic and three linear possibilities. Taking the case illustrated as an example, where the current pixel with coordinate x' is offset from the centre of five connected points, the gradient at that pixel is given by:-

$$\frac{\partial F}{\partial x'} = \frac{-34.F(x'-1)+3.F(x')+20.F(x'+1)+17.F(x'+2)-6.F(x'+3)}{70} \qquad 5.13$$

where F(x') is the depth coordinate of the pixel at point x'. Exactly analogous expressions were used for cases where the current pixel was at other locations in the connected surface, and for the calculation of the gradient along the y' axis.

The accuracy of these three gradient estimators was assessed using phantom data. Consider a two dimensional semi-ellipse with semi-axes a (along the x' axis) and b (along the z' axis). This can be used to assess the accuracy of the estimates, since all the techniques determine the gradients along x' and y' independently. Only the gradient estimates along the x' axis will be considered here. The equation of

the ellipse can be written as:-

$$Z^2 = b^2 \left(1 - \frac{X^2}{a^2}\right) \qquad 5.14$$

where Z and X are dummy coordinates parallel to z' and x' respectively. The true gradient at any point is given by,

$$\frac{\partial Z}{\partial X} = \frac{-b^2.X}{a^2.Z} \qquad 5.15$$

The unnormalised true surface normal at that point is therefore,

$$S_t = \left(\frac{b^2.X}{a^2 Z}, 1\right) \qquad 5.16$$

Note that there are only two components because this is a two dimensional phantom. If the estimated gradient is G, then the unnormalised estimated surface normal is,

$$S_e = (-G, 1) \qquad 5.17$$

Define the error in the gradient estimate to be the angle between the true and the estimated surface normals :-

$$E = \cos^{-1}(s_t.s_e) \qquad 5.18$$

where s_t and s_e are the normalised forms of S_t and S_e respectively. There are two advantages in defining the error in this way. Firstly, this angle should be a well-behaved function at all points along the ellipse, unlike the gradient which will tend to infinity at both ends of the ellipse (X = a and X = -a), and to zero at X = 0. Secondly, it is the angular error which will determine the effect on the calculated

shade value assigned to the surface.

Figure 5.6 shows a semi-ellipse with eccentricity 0.75, which is a reasonable representation of many organs in the body. The error angle E is also shown, calculated using equation 5.18 with the simple average of forward/backward differences used as a gradient estimator. The error remains insignificantly small for almost all points, rising at the edges of the organ to only 12^{o}. However, this result was obtained using real numbers to represent the coordinates of the ellipse, with a corresponding high degree of accuracy. For a cuberille model extracted from an N x N x N array of voxels, the object coordinates must be quantised into integer values. Although the subsequent transformation into observer coordinates may be performed with high precision, this quantisation effect must persist. A more realistic model, therefore, is shown in Figure 5.7, where the same ellipse is represented by integer values of x' and z', assuming a value for N of 128. The errors are much greater, and occur at all points along the surface of the organ, with a mean value of 8.1^{o} and a maximum value of 24^{o}.

The mean error angle was assessed using the three gradient estimators for a range of semi-ellipses of varying size. The semi-major axis a was increased from 5 to 40 pixels, keeping the eccentricity fixed at 0.75, and using an integer representation in all cases. The results are shown in Figure 5.8, and indicate that the polynomial estimator has significantly greater accuracy than either of the other two techniques, for a wide range of object sizes.

For ECAT data the problem of integer quantisation is compounded by that of noise. Because of the poor counting statistics associated with radionuclide imaging, the outlines of organs extracted from tomographic sections are frequently subject to random variations, even after low

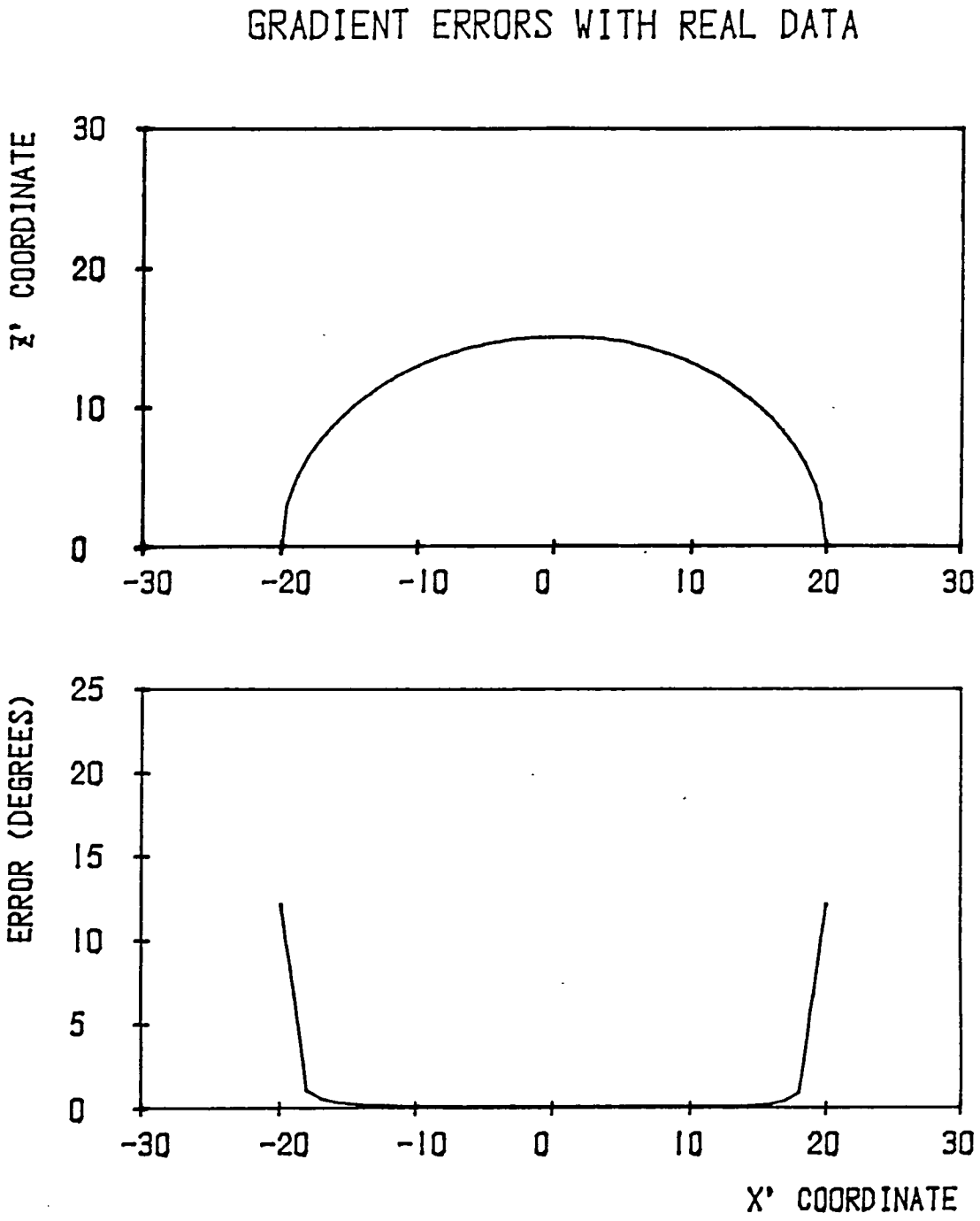


Figure 5.6 The semi-elliptical phantom used to assess the gradient estimators :-

Upper - Phantom using real number representation.
Lower - Error angle calculated using the simple average of forward/backward differences.

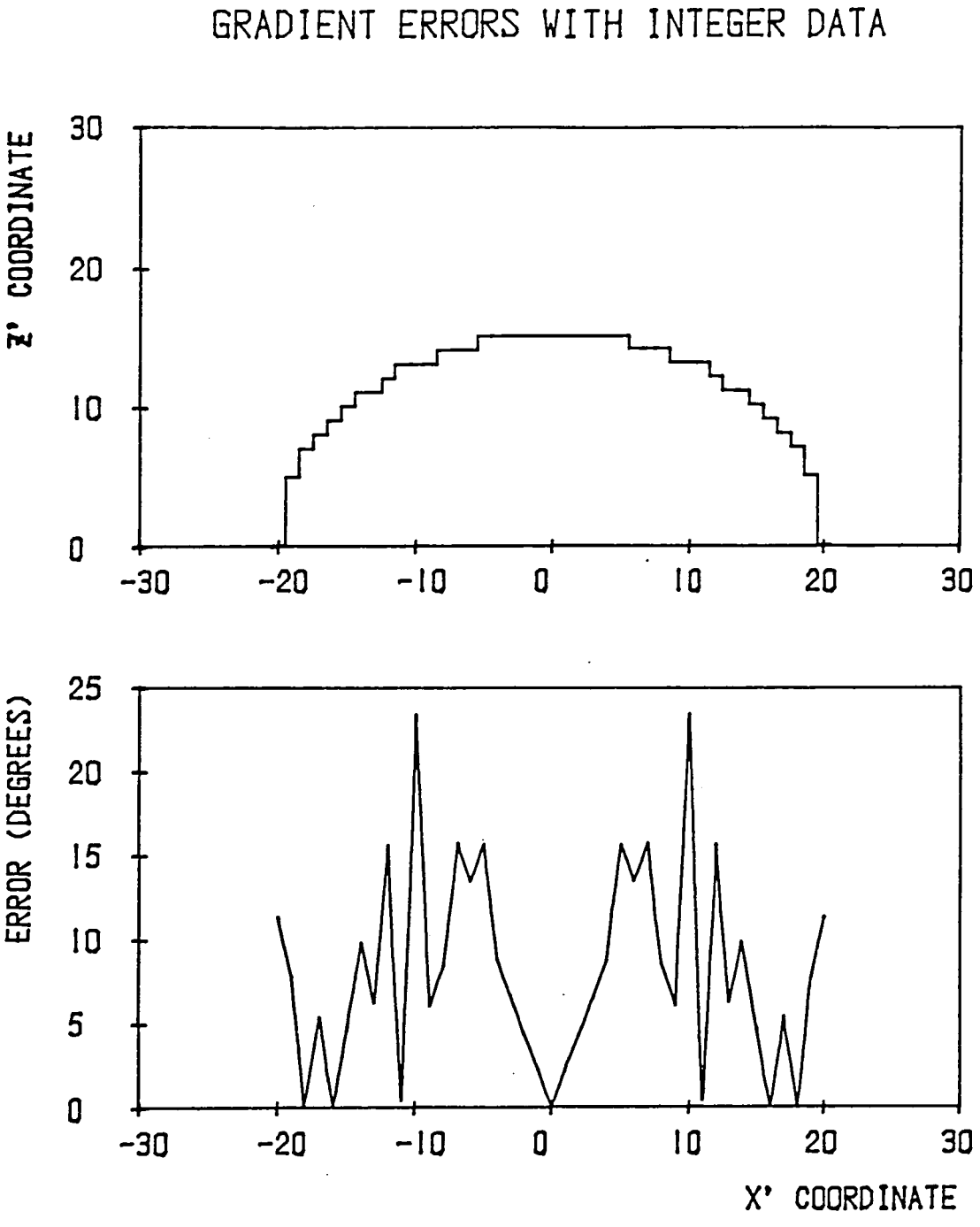


Figure 5.7 A more realistic version of the semi-elliptical phantom :-

Upper - Phantom using integer representation.
Lower - Error angle calculated using the simple average of forward/backward differences.

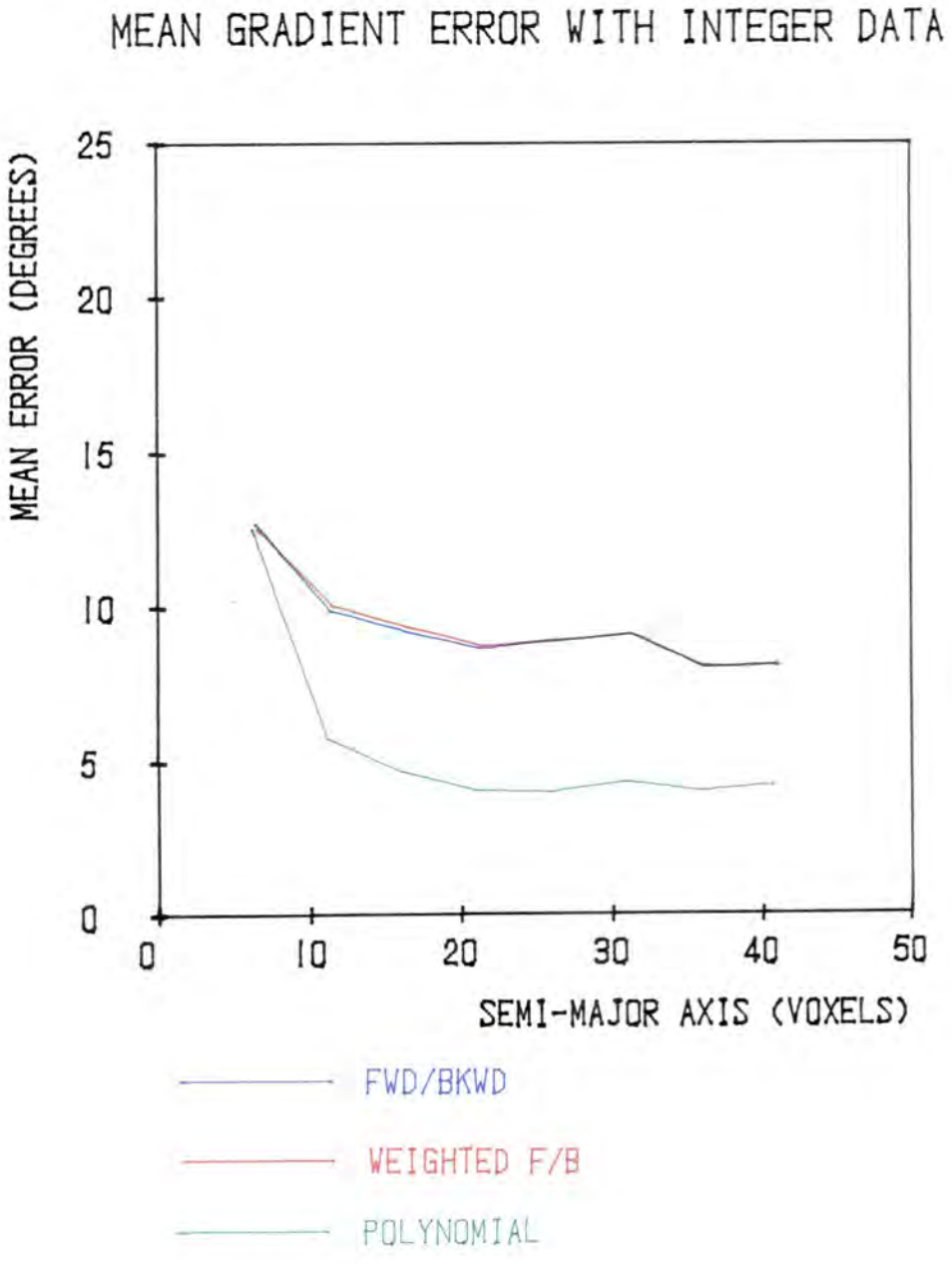


Figure 5.8 The mean error angle using an integer representation as a function of object size. The eccentricity of the ellipse was kept constant at 0.75 for all objects.

pass filtering. This effect was simulated by adding a random offset, uniformly distributed in the range -1 to +1 voxel, to the accurate z' coordinate before taking the integer value. The mean errors for the same range of sizes of semi-ellipse are shown in Figure 5.9, with the addition of noise. Each point shows the mean and one standard deviation for six evaluations using different random numbers. There is an overall increase in the magnitude of the errors compared to the noise free data, but the polynomial algorithm continues to have the smallest error. This is still the case for semi-ellipses with a wide range of eccentricities. Keeping the semi-major axis fixed at 20 voxels, the semi-minor axis was increased from 5 to 40 voxels, and the corresponding mean errors are shown in Figure 5.10. Each point shows the mean and standard deviation for six evaluations.

The polynomial gradient estimator is clearly more accurate than either of the two other methods for a wide range of object sizes and shapes, for both noisy and noise free data.

5.4 Subjective Assessment of Four Shading Techniques

The polynomial gradient estimator has been shown to be accurate, but this does not necessarily imply that shaded images calculated using this estimator are always to be preferred. There are several other parameters in equation 5.5 which may also be varied, and as stated in Section 5.1, shading algorithms also have a subjective component. A panel of eight observers was asked to compare images produced using four shading algorithms. Four of the observers were medically qualified, and four were physicists working in the field of medical imaging. The aims of this assessment were to determine which shading technique was preferred, whether a single technique was suitable for all the objects, and whether the observers were in agreement as to the rank order of the

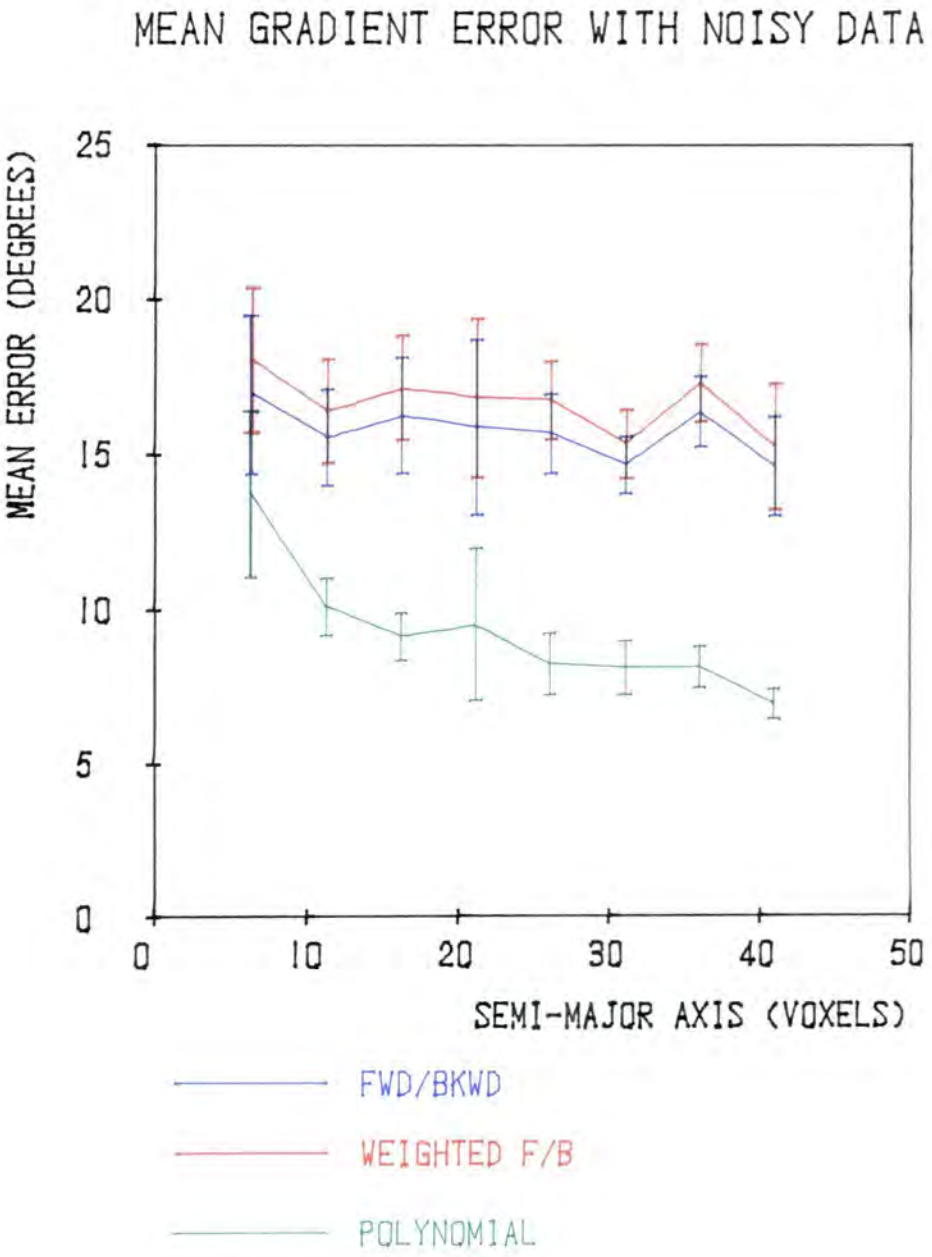


Figure 5.9 The mean error angle using an integer representation with added noise, as a function of object size. The eccentricity of the ellipse was kept constant at 0.75 for all objects.

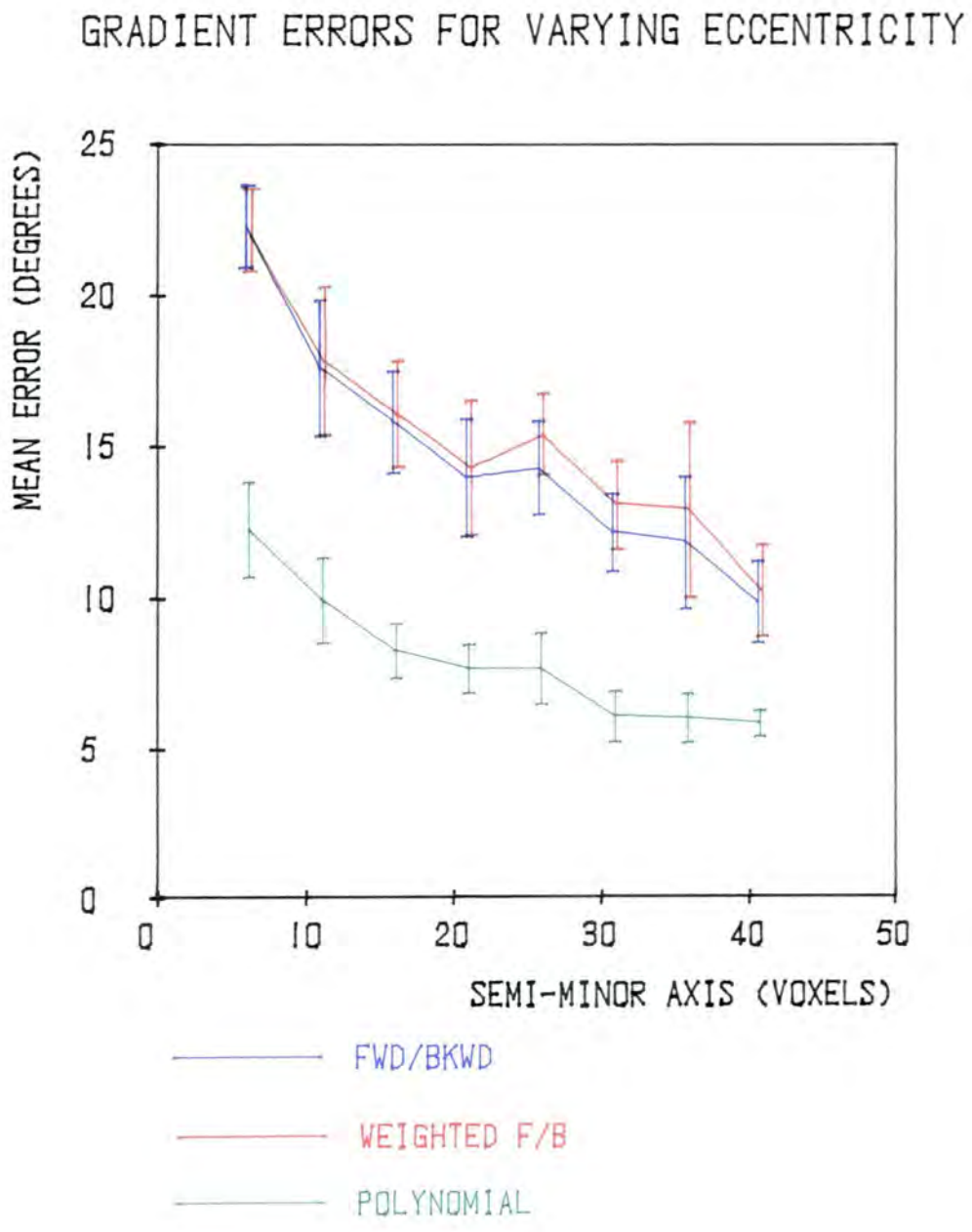


Figure 5.10 The mean error angle using an integer representation with added noise, as a function of eccentricity. The semi-major axis was fixed at 20 voxels for all objects.

shading algorithms. The shading techniques used were :-

Depth shading: calculated using the equation:-

$$S_d = I_s \cdot \frac{d_1 - d}{d_1 - d_o} \qquad 5.19$$

The parameters are as described in equation 5.5. This simple algorithm has been widely used for the display of shaded surface medical images.

Modified cosine shading: calculated using the equation:-

$$S_{mc} = I_s + D_r \cdot \frac{d_1 - d}{d_1 - d_o} \cdot [\cos(I/M)]^p \qquad 5.20$$

This algorithm was developed for the display of CT data using gradient shading, with the weighted average of forward and backward differences used to estimate the angle of incidence (Chen et al 1984).

Cosine shading: calculated using the equation:-

$$S_c = I_s + D_r.\cos(I) \qquad 5.21$$

This algorithm was specifically developed for this work to simulate Lambert's Law of diffuse reflection (Gibson 1983). The simple average of forward and backward differences was used to calculate the surface normal and hence the angle of incidence.

Polynomial shading: calculated using the equation:-

$$S_p = I_s \cdot \frac{d_1 - d}{d_1 - d_o} + D_r \cdot \frac{d_1 - d}{d_1 - d_o} \cdot [\cos(I)]^p \qquad 5.22$$

This technique was also developed for this work. The isotropic component and the directional component both have depth dependence, the cosine term has no discontinuities, and the polynomial gradient estimator was used to obtain the angle of incidence (Gibson 1986a).

Actual values for the parameters used in equations 5.19 to 5.22 are shown in Table 5.1. The incident light vector i was chosen to be (0, 0.7071, 0.7071) for S_c and S_p; and to be (0, 0, 1) for S_{mc}, to avoid shading discontinuities. Figure 5.11 shows the eight test objects shaded using each of the four techniques. Four objects were derived from ECAT data, two from CT data, and two were artificial objects. Forty sets of four images were presented in a random order to each observer. Each set showed one of the eight test objects at one of five orientations, shaded using each of the four techniques. The images were displayed in the four quadrants of the screen, labelled a, b, c and d (Figure 5.12). The placing of techniques (S_d, S_{mc}, S_c and S_p) was randomised, so that the same technique did not appear in the same quadrant every time. For each set the observer was asked to rate the four images in order of preference, ties being forbidden. Observers were allowed to vary the display brightness and contrast as desired, and were left undisturbed to perform the assessment at their own pace. When choosing the preferred order for the four images the following criteria were suggested:-

a) Depth perception.

b) Surface detail.

c) Artefacts.

However, observers were encouraged to express their own preferences, rather than be guided by the investigator. This scheme for the subjective assessment of four display techniques is identical to one

Table 5.1 Values of parameters used in the four shading algorithms

Parameter	Value used in technique:-			
	S_d	S_{mc}	S_c	S_p
I_s	255	40	80	130
D_r	-	200	140	130
M	-	2	-	-
p	-	0.6	-	0.6

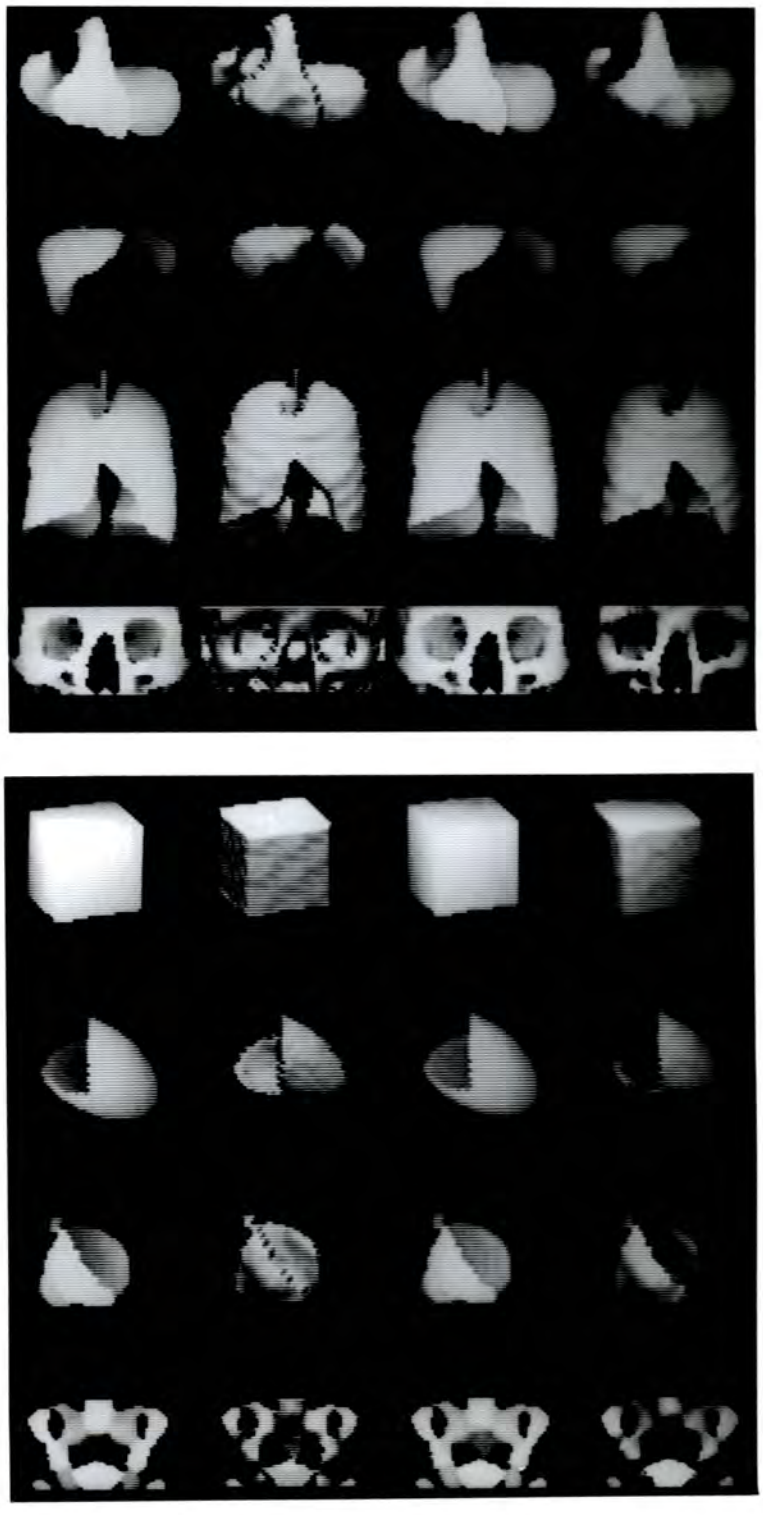


Figure 5.11 The eight test objects displayed using each of the four shading algorithms :-

1st Column - Depth.
2nd Column - Cosine.
3rd Column - Modified Cosine.
4th Column - Polynomial.

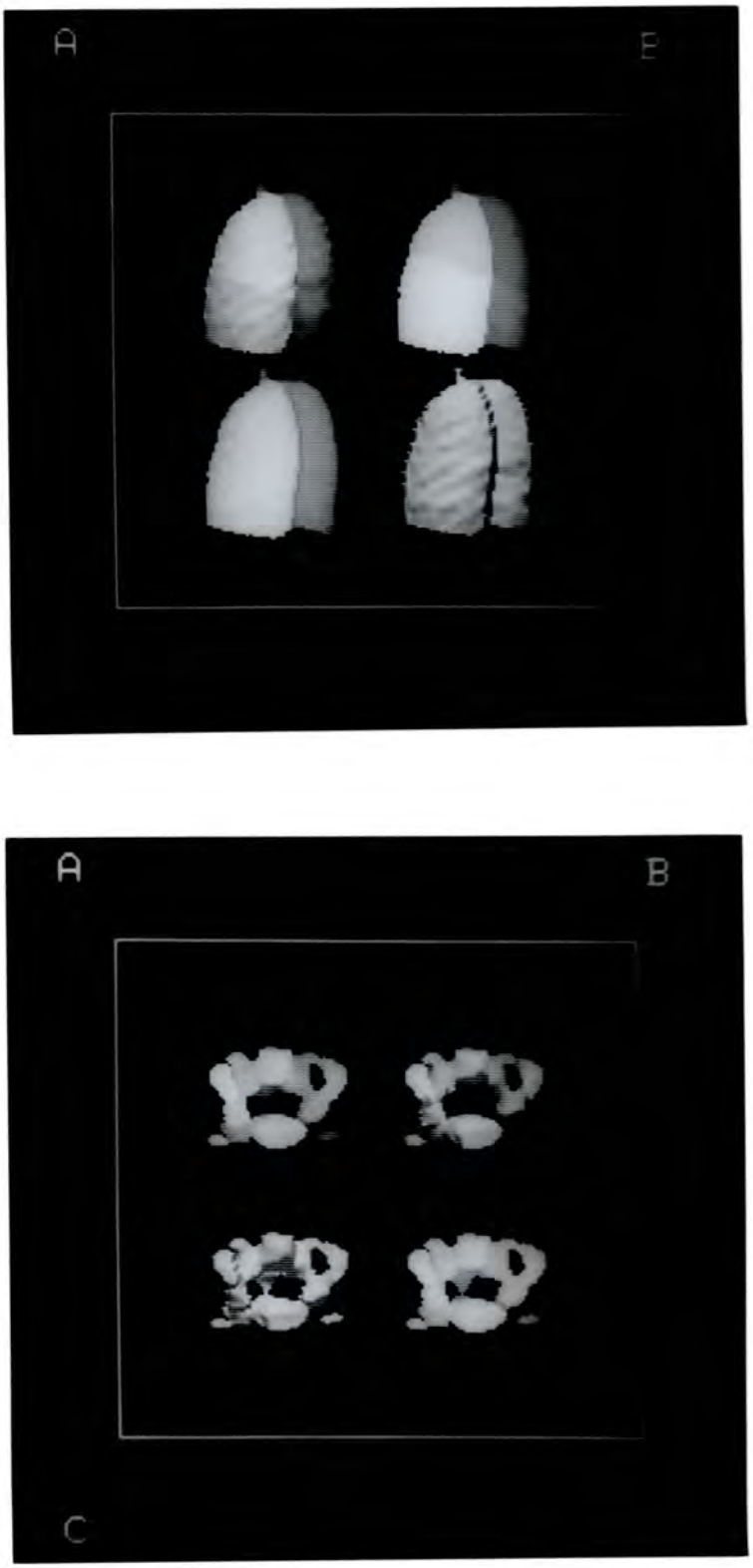


Figure 5.12 Image sets as presented to the observers, showing the random distribution of algorithms in the four quadrants. These two sets represented the extremes of agreement and disagreement respectively.

Upper - Four images of the Lungs, for which the observers were in excellent agreement (W=0.925).
Lower - Four images of the Pelvis, where agreement was extremely poor (W=0.081).

described for the assessment of contrast and display levels using NMR images (Ridgway and Sharp 1986).

The results of this assessment are summarised in Tables 5.2 to 5.7. Table 5.2 shows the number of times each algorithm was rated 1st, 2nd, 3rd and 4th. If all algorithms were considered to be equally good, then this table should contain a uniform distribution. In fact the distribution is very non-uniform, with S_p having the lowest total of summed ranks and S_d the highest. The same pattern of preferences was seen when the objects were analysed by imaging modality (Table 5.3), indicating that the polynomial algorithm was the preferred form of shading irrespective of the type of data being displayed. The Sign Test (Gibbons 1985) may be used to establish the degree of significance to ascribe to the observed preferences. The rankings for each pair of algorithms can be compared, and the number of times that the first technique is preferred to the second recorded. For C_m comparisons between pairs of algorithms where no genuine preference exists, the number of recorded preferences P should follow a binomial distribution with mean $C_m/2$. For reasonably large samples ($C_m > 20$) a standard normal deviate can be constructed:-

$$Z_s = \frac{P - \frac{1}{2} - C_m/2}{C_m/2} \qquad 5.23$$

The significance of any excess number of preferences can be determined using Z_s as a test statistic. Table 5.4 shows the number of preferences and the significance levels for all six pairs of algorithms using pooled data from all observers and all objects ($C_m = 320$). The polynomial shade S_p was preferred to all the other techniques with

Table 5.2 Rankings for the four shading algorithms

Rank Order	Rank frequency for algorithm :- S_d	S_{mc}	S_c	S_p
First	9	3	126	182
Second	28	44	122	126
Third	135	152	25	8
Fourth	148	121	47	4
Sum of ranks	1062	1031	633	474

(Data are for all observers and all objects).

Table 5.3 Variations in rank sum between imaging modalities

Type of data	Sum of ranks for algorithm :- S_d	S_{mc}	S_c	S_p
ECAT	523	524	309	244
CT	280	250	159	111
Artificial	259	257	165	119
All data	1062	1031	633	474

(Data are for all observers)

Table 5.4 Preferences when techniques are taken in pairs (a total of 320 comparisons)

Algorithm pair	No. of preferences for first over second	Significance Level
S_{mc}/S_d	170	NS
S_c/S_d	264	0.1%
S_c/S_{mc}	254	0.1%
S_p/S_d	308	0.1%
S_p/S_{mc}	307	0.1%
S_p/S_c	191	0.1%

(Data are for all observers and all objects)

Table 5.5 Variations in degree of significance of preference between imaging modalities (160 comparisons for ECAT data, and 80 comparisons for CT and Artificial data)

Algorithm pair	ECAT data		CT data		Artificial	
	Pref.	Signif.	Pref.	Signif.	Pref.	Signif.
S_{mc}/S_d	83*	NS	53	0.3%	40	NS
S_c/S_d	133	0.1%	67	0.1%	64	0.1%
S_c/S_{mc}	128	0.1%	63	0.1%	63	0.1%
S_p/S_d	153	0.1%	80	0.1%	74	0.1%
S_p/S_{mc}	153	0.1%	80	0.1%	73	0.1%
S_p/S_c	90	7%	49	3%	52	1%

(Data are for all observers, *Preference was reversed)

significance at least exceeding the 0.1% level. The same pattern of preferences was seen when the data was analysed separately by imaging modality (Table 5.5), although the levels of significance were less. Preference for the polynomial over the cosine algorithm just failed to reach a significant level of probability for ECAT data, but remained significant for both CT data ($p<3\%$) and artificial data ($p<1\%$).

The same trends were observed when the observers were considered separately. Table 5.6 shows the rank sum for each algorithm for each observer separately. The polynomial algorithm was the overall first choice for 7 observers, and second choice for 1 observer, using the rank sum as the assessment parameter. The significance of these preferences for algorithms taken in pairs is shown in Table 5.7, again for each observer separately. The polynomial algorithm was preferred to the depth and modified cosine algorithms by all observers with significance levels at least exceeding the 0.1% level. The polynomial algorithm was also considered preferable to the cosine algorithm by 6 out of the 8 observers, although significance levels for several of these preferences were low.

Although these results strongly suggest an overall preference for the polynomial and cosine algorithms, they do not indicate whether observers are always in agreement as to the rank order for individual images. The coefficient of concordance may be used to quantify the agreement between the observers for individual sets of images (Gibbons 1985). This coefficient W lies in the range 0 to 1, indicating no agreement between rankings, or complete agreement. A test statistic can be calculated from W, and the significance of the observed agreement measured. Values of W and the corresponding levels of significance were calculated for all forty sets of images, and the observers were in

Table 5.6 Variations in rank sum between observers

Observer	Sum of ranks for algorithm :-			
	S_d	S_{mc}	S_c	S_p
1	137	143	52	68
2	125	116	108	51
3	132	130	73	65
4	139	125	82	54
5	131	134	78	57
6	132	142	73	53
7	132	120	89	59
8	134	121	78	67
All	1062	1031	633	474

Table 5.7 Variations in degree of significance of preferences between observers

Observer	Algorithm pair :-					
	S_{mc}/S_d	S_c/S_d	S_c/S_{mc}	S_p/S_d	S_p/S_{mc}	S_p/S_c
1	NS	0.1%	0.1%	0.1%	0.1%	1%*
2	NS	NS	NS	0.1%	0.1%	0.1%
3	NS	0.1%	1%	0.1%	0.1%	NS*
4	NS	0.1%	0.1%	0.1%	0.1%	1%
5	NS	0.1%	0.1%	0.1%	0.1%	NS
6	NS	0.1%	0.1%	0.1%	0.1%	5%
7	5%	1%	1%	0.1%	0.1%	NS
8	NS	0.1%	1%	0.1%	0.1%	NS

(Data are for all objects, *Preference was reversed)

excellent agreement, with measured values of concordance significant at the 0.1% level 19 times, at the 1% level 10 times and at the 5% level 8 times. The image sets illustrated in Figure 5.12 represent the extremes of observer agreement and disagreement respectively. In the case of the lung images, the surface modulated techniques (S_c and S_p) are clearly superior to the others, revealing details of the rib impressions. For the more irregular pelvis, however, excessive surface modulation can become confusing, and the observers were unable to agree on a consistent rank order.

Examination of Figure 5.11 shows that the algorithms fall into two groups. The depth and modified cosine techniques (S_d and S_{mc}) have relatively little surface modulation compared to the cosine and polynomial techniques (S_c and S_p). Overall, the observers showed a clear preference for the techniques where the shade was principally determined by surface orientation. The choice between polynomial or cosine law displays was closer, but a significant preference was expressed for S_p in most cases.

5.5 Special Effects

5.5.1 Colour

The illumination and reflection models described in Section 5.1 are purely monochromatic. They can be extended to include both coloured illumination and coloured reflective surfaces (Phong 1975). In a full colour model both the incident light intensity and the coefficients of reflection are functions of wavelength. However, a simple scheme can give adequate results using an empirical colour model. A conventional raster display memory may have only one byte per pixel, allowing any one of 256 colours to be displayed. The colours to be displayed are often

under software control, using a system of three look up tables for the red, green and blue video guns. In a typical system, each look up table contains 256 elements, and each element is a single byte specifying the gun strength in the range 0 to 255. The total number of possible colours (the 'palette') is therefore 256^3, of which 256 may be displayed at any one time. It is usually easier to specify a look up table using the parameters hue, lightness and saturation (HLS), rather than red, green and blue (RGB) gun strengths. Hue varies cyclically from 0^o to 360^o, running from blue through red and green, and back to blue. Lightness indicates the intensity on a scale of 0 to 1, and saturation the colour purity, also on a scale of 0 to 1. Points in (HLS) space can be mapped into (RGB) space using a simple algorithm (Farrell 1983).

There are several possible uses for colour as an adjunct to shaded surface display:-

Coloured opaque objects: Comprehension of a complex shaded surface image can be facilitated by the use of different hues to identify different objects in the scene. A look up table was defined containing six discrete hues (blue, magenta, red, yellow, green and cyan) at each of forty intensity levels. Scenes containing several separate objects were processed using the technique for Direct Display of Binary Objects described in Section 4.4, to create a two dimensional array containing the depth coordinates of visible voxels. A second array of equal size was created simultaneously, containing coded integer values identifying the object in the scene which was visible at each pixel. The polynomial shading technique was applied to the depth array, with values of I_s and D_r in equation 5.5 chosen to produce shade values in the range 0 to 39. Offsets of 0, 40, 80 etc. were then added to these shade values,

depending on the object visible at that point in the image. When the final image was displayed, intensity values were determined by surface orientation, as in the monochromatic model, but hue appeared as an arbitrary property identifying each of the objects in the scene. Figure 5.13a shows colour coded images of the cardiac blood pool at four orientations, in which the cardiac chambers have been assigned different hues. Successive applications of the direct display algorithm have correctly solved the hidden surface problem for these objects, whilst providing additional information using colour coding. Any display algorithm can be used, providing that information can be preserved concerning the objects visible at all points in the image. A more complex object is shown in Figure 5.13b, in which the lungs have been segmented mathematically to show the approximate locations of the three right and two left lobes.

Coloured interpenetrating objects: The same technique and look up table can also be used to display objects which interpenetrate. Such objects can be extracted from a single set of tomographic sections using different segmentation criteria, or from multiple sets of tomographic sections obtained at different time intervals. The objects are shaded and displayed using two or more hues. The final image shows the outer surface of the composite object, with the hue indicating which object is visible at that point. This technique is particularly suited to the display of cardiac data (see Section 6.3).

Colour as an indicator of function: Conventional 2-D projection images, whether of radionuclide concentration or of xray attenuation, are usually presented in monochrome. However, colour display scales have also been used, especially for the display of radionuclide images. If

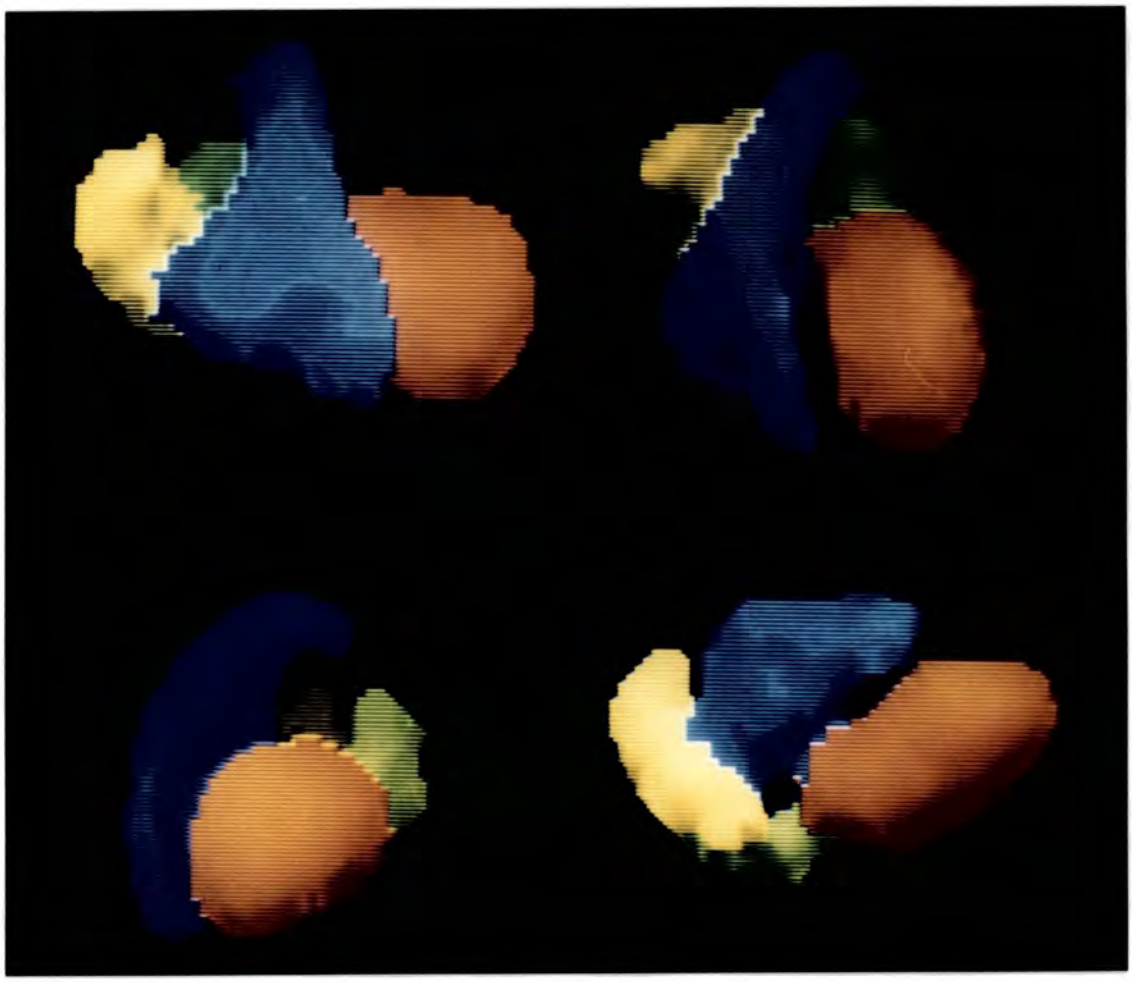


Figure 5.13a Coloured shaded surface images of the cardiac blood pool, showing the left ventricle in red, the right ventricle in blue, the left atrium in green and the right atrium in yellow:-

Upper left - Anterior.
Upper right - Left anterior oblique.
Lower left - Left lateral.
Lower right - Inferior.

Figure 5.13b Coloured shaded surface images of the lungs, showing the left upper lobe in yellow, the left lower lobe in green, the right upper lobe in blue, the right middle lobe in magenta and the right lower lobe in red :-

Upper left - Anterior.
Upper right - Four oblique aspects.
Lower left - Right lateral.
Lower right - Left lateral.

both a projection image and a shaded surface image are available which show identical aspects of an organ, then a composite image may be created in which intensity is derived from surface orientation, and hue from projection image value. To ensure perfect registration between projection and surface images, the projection image was calculated by re-projecting values from the tomograms along the chosen viewing direction (Harris 1982). A colour look up table was created containing all possible combinations of 16 hues and 16 intensity levels. The hues ranged from blue through green and yellow to red, to indicate increasing values of the projection image. Values in the display memory in the range 0 to 15 specified these hues at the minimum intensity, 16 to 31 the same hues at the next intensity level, and so on up to values 240 to 255, which specified the hues at their maximum intensity level. Each byte in the display memory can be split into two 'nibbles' each of four bits. Using this look up table the less significant nibble specified the hue, and the more significant nibble the intensity. Projection images were created and scaled into the range 0 to 15, before being transferred into the less significant nibbles of the display. Corresponding shaded surface images were created using values of I_s and D_r in equation 5.5 which produced shade values in the range 0 to 15, and were transferred to the more significant nibbles of the display. Figure 5.14 shows monochromatic shaded surface images, together with images modulated by hues derived from projection images. The 'doughnut' sign in the lower right image is indicative of a defect in the liver. However, when this technique is applied to images at several angles, the hue assigned to different points on the surface changes as the object rotates (Figure 5.15). This is because the projection image is formed by summation along lines parallel to the viewing direction. It therefore

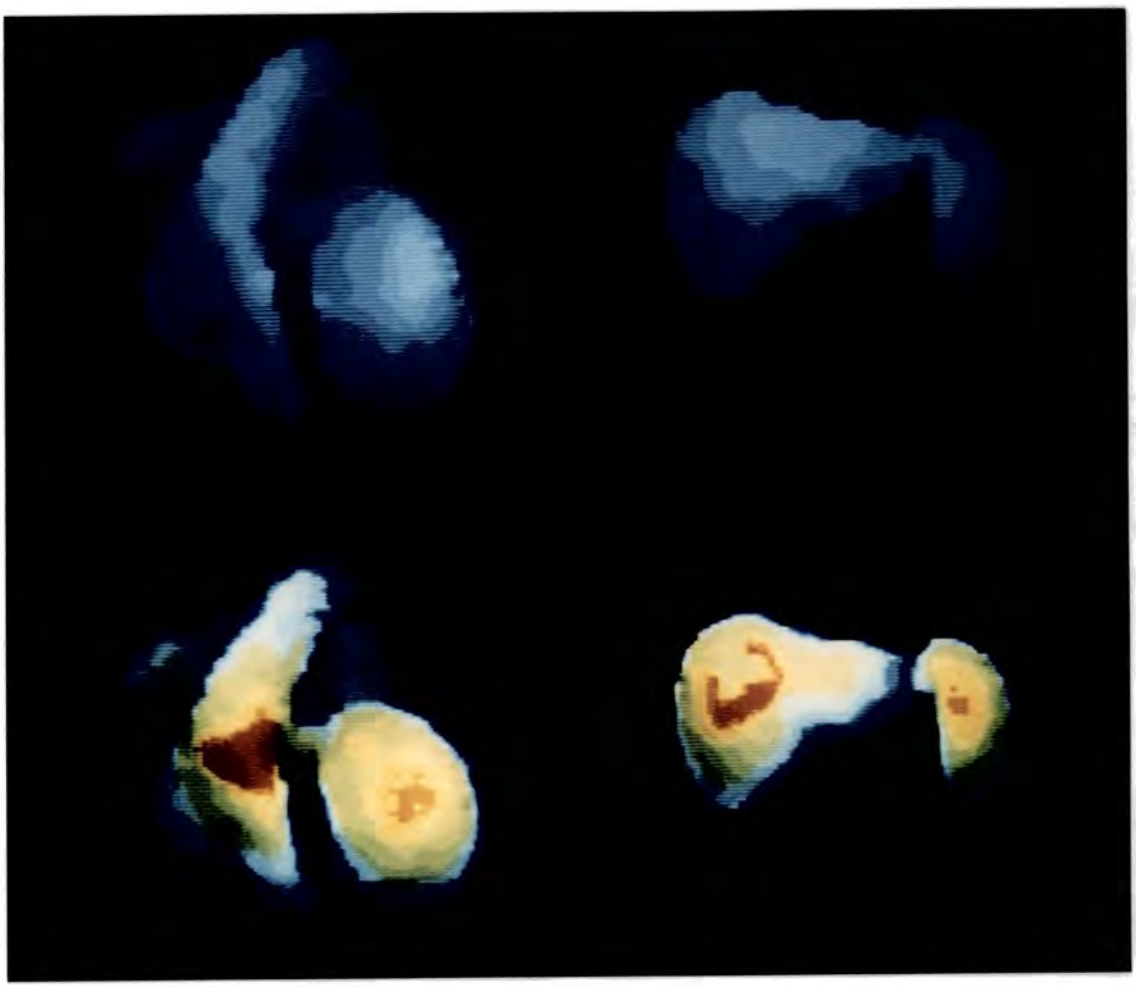


Figure 5.14 Monochrome and colour modulated shaded surface images of the heart and liver :-

Upper left - Heart in the left anterior oblique aspect.
Upper right - Liver and spleen in the anterior aspect.
Lower left - Colour modulated using projection values.
Lower right - Colour modulated, revealing a defect in the right lobe of the liver.

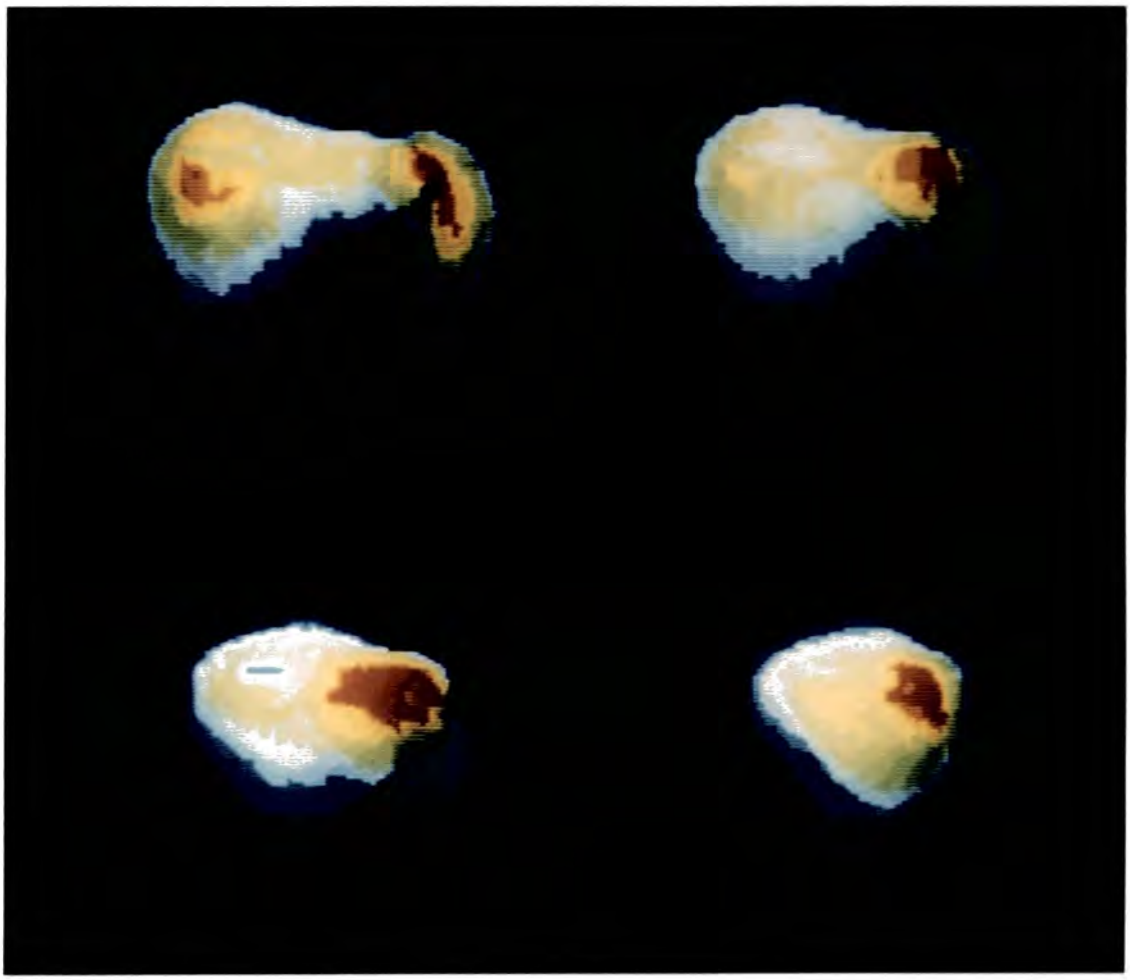


Figure 5.15 Colour modulated shaded surface images of the liver and spleen, rotated from the anterior to the right lateral aspect. The vivid red hue indicates object overlap rather than organ function.

has maxima whenever two objects overlap, and discontinuous changes as they rotate about each other. The effect is seen clearly in the upper right quadrant of Figure 5.15, where the vivid red hue is caused by the overlap of liver and spleen. These composite images are therefore prone to abrupt shifts of hue as the viewing direction alters, and this effect reduces their clinical value. Neverthless, the combination of functional and anatomical information in a single image represents an extremely efficient way of communicating complex data, providing that the limitations of the method are understood.

Colour as an indicator of temporal change: Using conventional 2-D projection imaging, dynamic sequences can be obtained by taking images at regular time intervals. Such sets of images can be processed to give a single 2-D parametric image, in which pixel values are calculated from the time variation of projection image values. Typical parameters are the time of maximum or minimum, and for gated cardiac images, the amplitude or phase (Ratib et al 1982). Phase images are displayed using a cyclic colour scale, containing many hues. Just as in the previous case, these images can be modulated by intensity levels derived from shaded surface images at the same angle. Figure 5.16 shows a phase image, a shaded surface image and the composite image obtained when the two are combined. To ensure perfect registration the phase image was calculated from a set of re-projected images (Harris 1982) obtained from the same blood pool tomograms as the surface data. The effect of this technique is to map phase values onto the surface of the heart. Because re-projection can be performed at any angle, surface phase images can also be created at any angle (Figure 5.17), allowing any apparent phase discrepancies to be checked against the cardiac anatomy.

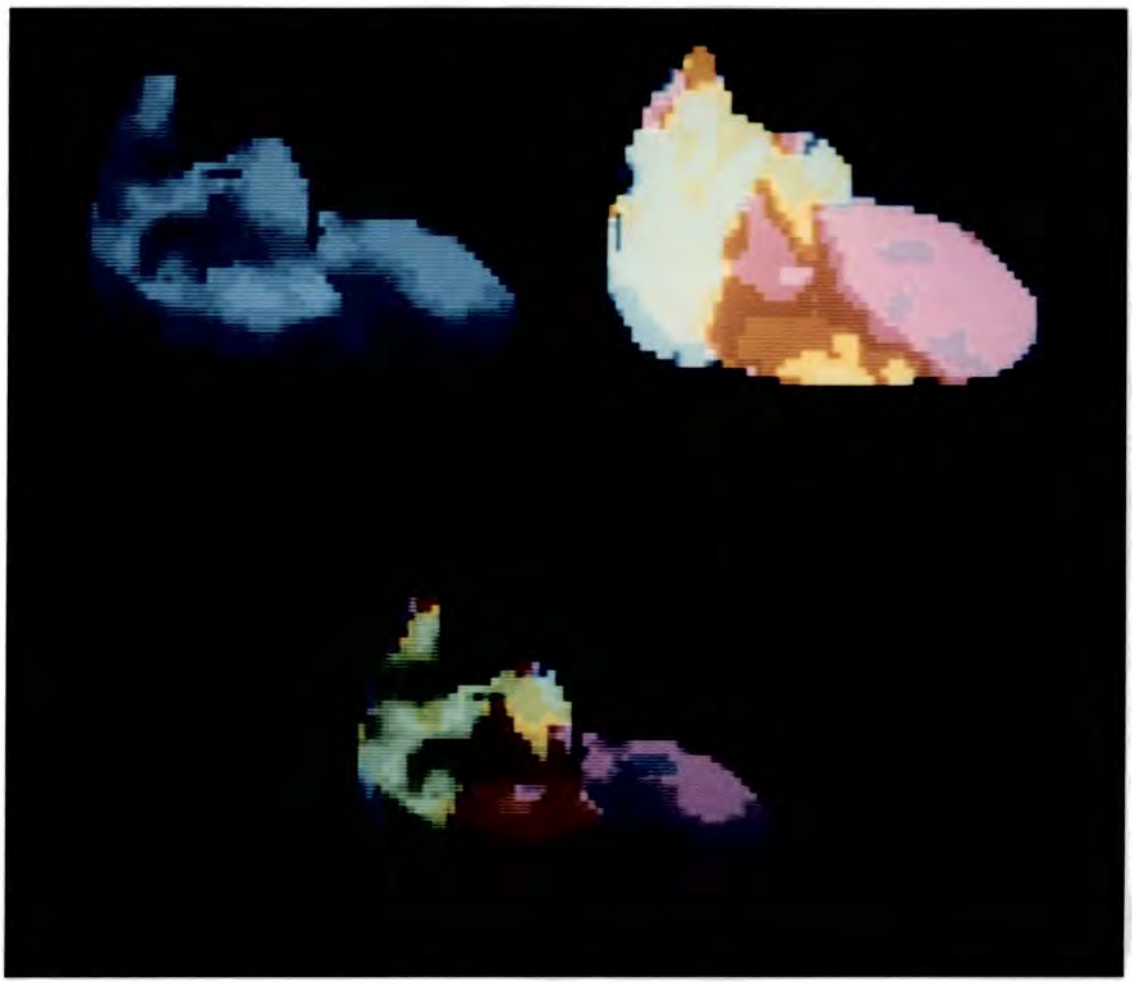


Figure 5.16 Colour modulation by phase values :-

Upper left - Shaded surface image of the heart from the anterior aspect.
Upper right - Phase image obtained at the same angle of view.
Lower - Colour modulated image, in which the hue indicates phase, and intensity indicates surface orientation.

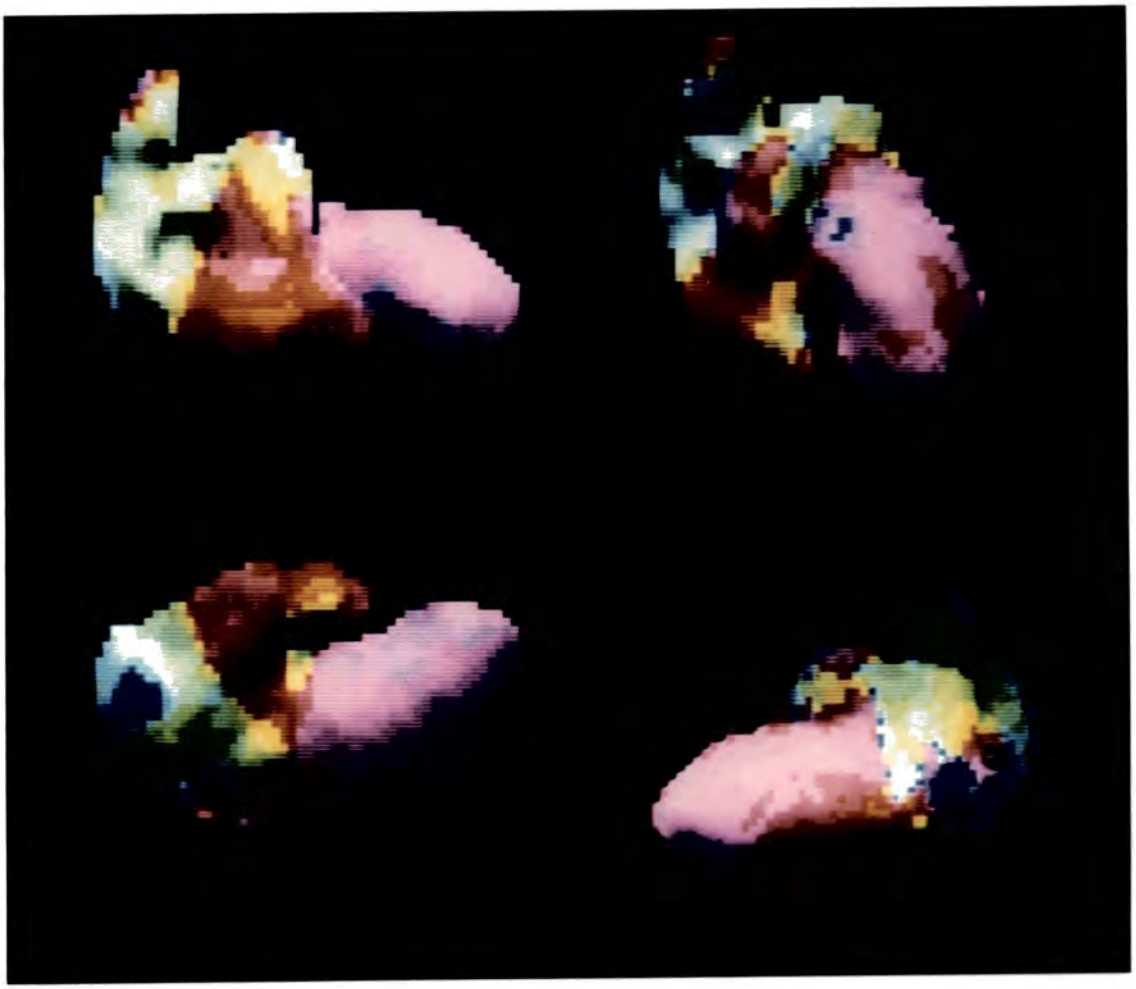


Figure 5.17 Composite phase and surface images at several orientations :-

Upper left - Anterior.
Upper right - Left anterior oblique.
Lower left - Inferior.
Lower right - Left posterior oblique.

5.5.2 Transparency

All the reflection models described so far are based on diffuse reflection from opaque objects. To show multiple surfaces simultaneously requires an algorithm for simulating transparency. Such surfaces are usually modelled using modified specular reflection and parallel incident light. Using the notation of Figure 5.1, a simplified empirical model for the intensity of light reaching the observer is (Phong 1975):-

$$S = D_r.r(I).\cos(J) \qquad 5.24$$

where S is the calculated shade value; I is the angle of incidence; J is the angle between the specularly reflected ray and the observer's viewing direction; and r(I) is a function which determines the fraction of light reflected as a function of the angle of incidence. To simulate a glass surface r(I) is small for angles close to zero, and rises rapidly to 1 as I approaches 90^{o}. For smooth objects, equation 5.24 gives a good simulation of the highlights reflected from glassy surfaces. However, the irregular surfaces of most medical objects are not well suited to this form of display, since only a few parts of the surface reflect sufficient light towards the observer. The algorithm also requires angle J to be evaluated for all visible points on the surface.

A modified version of this algorithm has therefore been developed for this work, using a similar reflection model, but with ambient isotropic illumination rather than directional light. For this model the reflection is constrained to be precisely specular, i.e. the reflected ray must be directed exactly along the observer's line of

sight. If an object is bathed in ambient isotropic light, then for all visible points on the surface there will always be one incident ray which reflects specularly along the observer's viewing direction. This is proved more easily by reversing the direction of the light rays, and imagining the observer as a source of parallel illumination. Every visible point of the object will be illuminated, and the ray will be specularly reflected at an angle determined by the local surface orientation. That 'reflected' ray defines the path along which incident ambient light will reach the observer. The intensity of light reaching the observer is now:-

$$S = I_s.r(I) \qquad 5.25$$

where I is the angle of incidence, which is equal to the angle of reflection, i.e. to the angle between the observer's viewing direction o and the local surface normal s. The function r(I) was modelled using the expression:-

$$r(I) = \exp [\ 2.I - 3.14159\ ] \qquad 5.26$$

for values of I in radians. Figure 5.18 shows opaque and transparent images of the brain, obtained from a set of NMR images. The data was sampled using 128 x 128 x 128 voxels and displayed using the modified BTF algorithm. The use of transparency shading enables simultaneous display of interior structures. Figure 5.19 shows an opaque brain within a transparent head, and opaque ventricles within a transparent brain. The relationship between all three objects can be perceived directly. Comprehension of overlapping surfaces is further enhanced if different

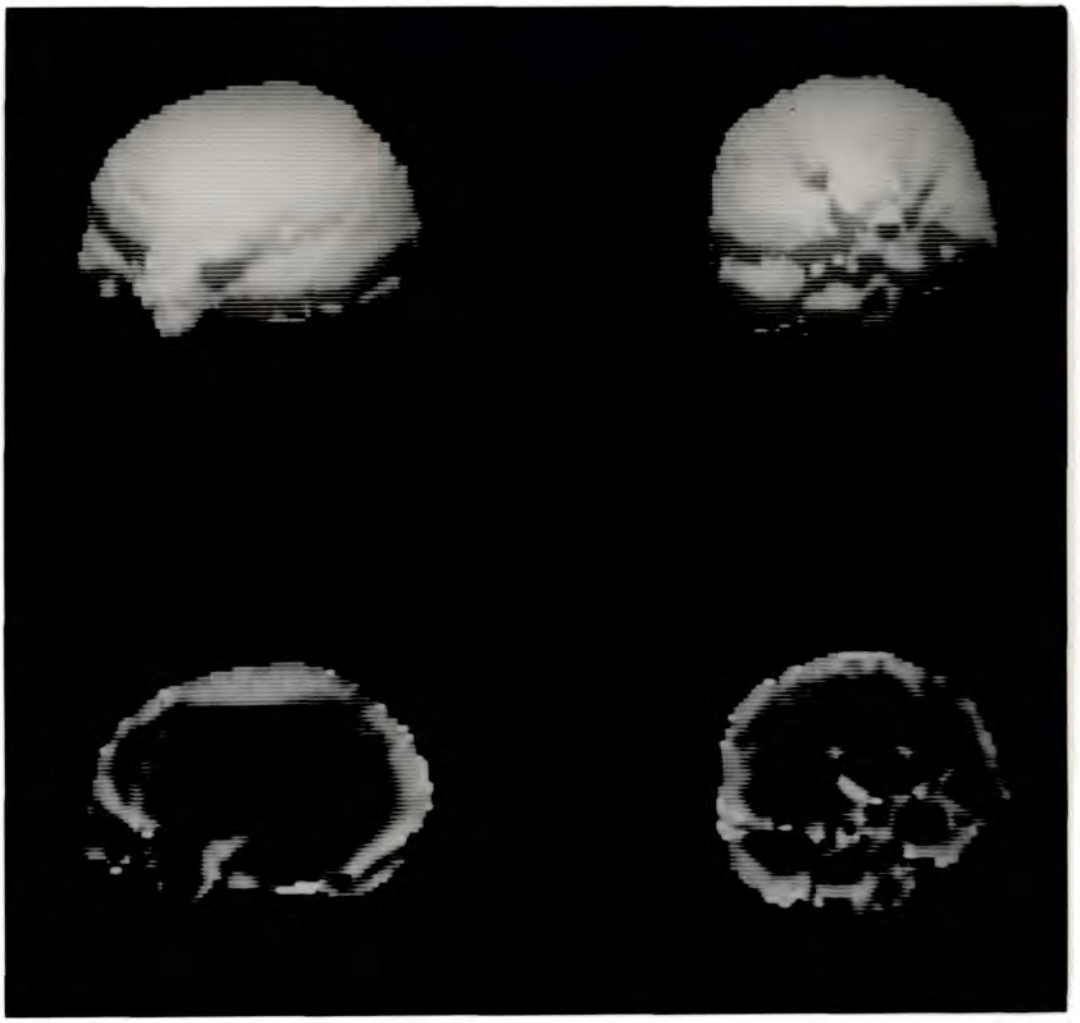


Figure 5.18 Images of the brain derived from NMR data, and displayed at two aspects using the modified BTF algorithm :-

Upper - Using opaque shading.
Lower - Using transparent shading.

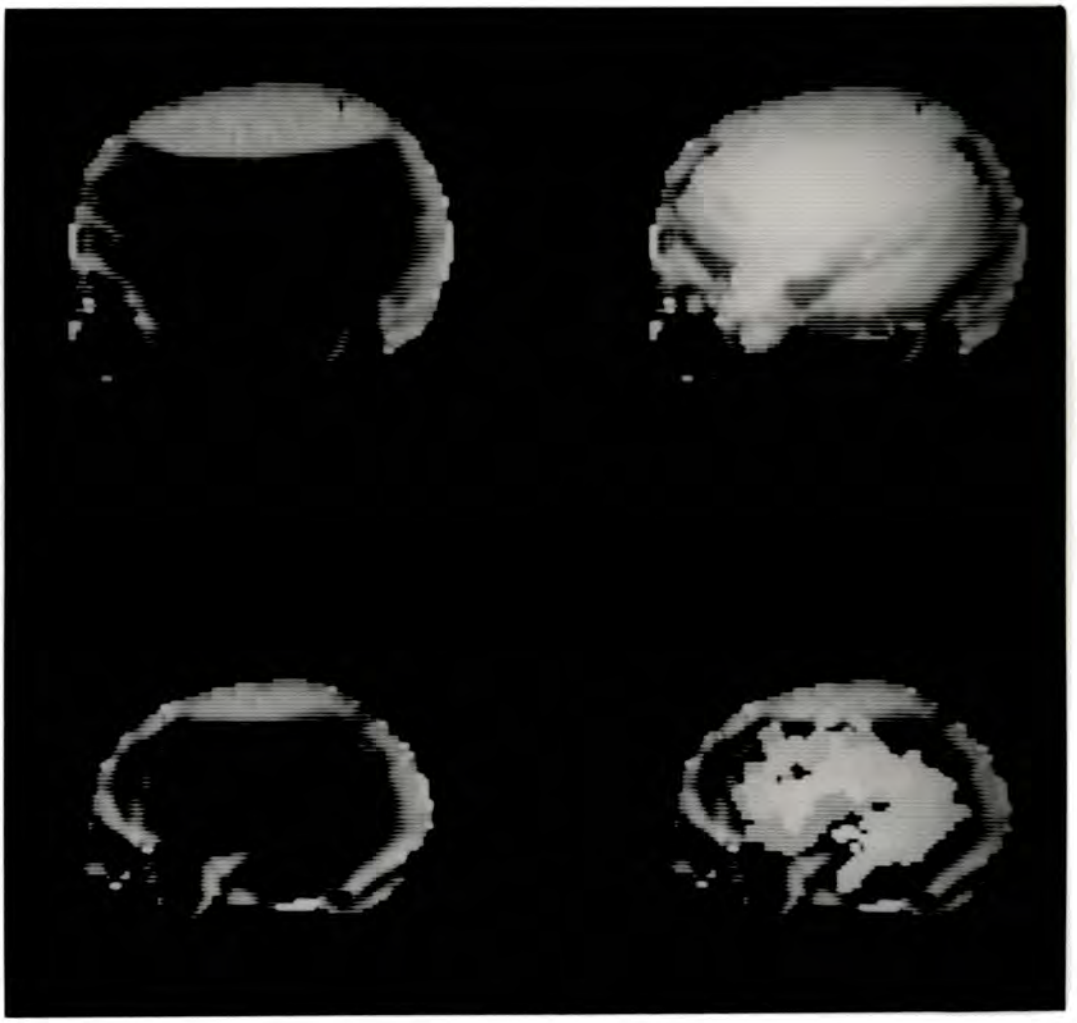


Figure 5.19 Composite opaque and transparent shaded images of the head, brain and ventricles :-

Upper left - Transparent head.
Upper right - Head containing opaque brain.
Lower left - Transparent brain.
Lower right - Brain containing opaque ventricles.

hues are assigned to each surface. The use of a composite shading model, with both opaque and transparent components, enables images to be produced in which the outer surface is gradually rendered transparent to reveal an internal object (Figure 5.20). In this case the head becomes gradually transparent, allowing the opaque shading of the brain to become visible.

The success of this transparency algorithm depends on two factors. Firstly the development of a specular reflection model which assumes isotropic rather than directional illumination, and secondly the assignment of different hues to different objects. Both factors combine to enable the observer to comprehend complex nested structures.

5.5.3 Stereoscopy

The images shown in Figures 5.18 to 5.20 are realistic simulations of the appearance of three dimensional objects. However, they are two dimensional images. Observers perceive the real world using a pair of frontally set eyes with a spacing of approximately 6 cm. The resulting relatively small difference between the two retinal images is used by the visual system to determine the distance and solidity of objects. If two images of the same object incorporating the correct amount of lateral shift are presented separately to each eye, then the object is perceived as a three dimensional solid. This can be of value for the display of large amounts of data, in particular to indicate the dependence of a function on two variables simultaneously (Grotch 1983).

The incorporation of stereopsis in the display of shaded surface images would appear to offer additional depth cues, improving the ability of observers to determine the exact location of lesions. However, the use of a raster display screen with only medium resolution

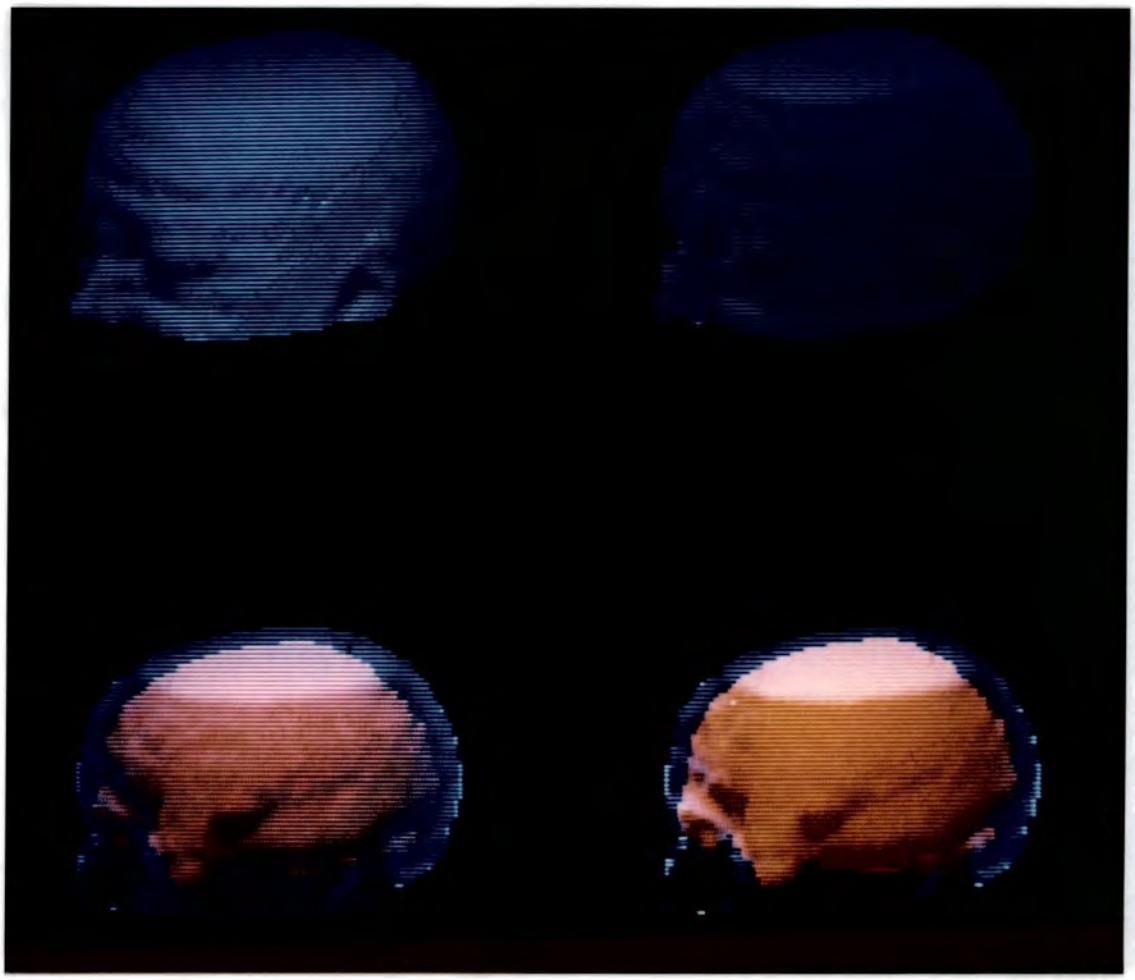


Figure 5.20 Gradual shifting from opaque to transparent shading of the head reveals an opaque shaded brain.

(256 x 256 pixels) imposes severe restrictions. The problems can be illustrated using random dot stereograms. These are artificially created pairs of images in which the left eye image contains black and white dots distributed randomly, and the right eye image contains the same information, but with one or more blocks of dot patterns shifted to the left. The areas of shifted dots will be perceived as lying at different distances, depending on the amount of shift. Such images have been widely used in the investigation of stereopsis (Julesz 1971).

Figure 5.21 shows a random dot stereogram in which two square regions have been shifted 8 and 16 pixels respectively, giving the effect of two raised areas. In Figure 5.22 the amount of shift is determined by the depth of the visible voxels in a three dimensional object sampled at 64 x 64 x 64 resolution. The object consists of a pair of lungs, seen from the anterior aspect and the stereoscopic effect is powerful, especially for the heart cavity and the sloping inferior surfaces. However, using a raster display memory, shift values are limited to integral numbers of pixels. Shifts of more than 32 pixels (in a 256 x 256 screen) cause too great a discrepancy in the retinal images for comfortable stereopsis. Hence objects can only be perceived as lying in one of 32 planes. This is clearly seen in Figure 5.22, where the object appears as a stack of separate vertical planes rather than a solid object.

Stereopsis is a very sensitive phenomenon, with observers capable of detecting depth differences of as little as 0.05 mm at 50 cm (Braddick 1982). A shift of 1 pixel in an array of 256 x 256 pixels is equivalent to a very much larger depth difference, and so individual parts of the object must be perceived at different depths according to the number of pixels shifted. Even high resolution screens will show

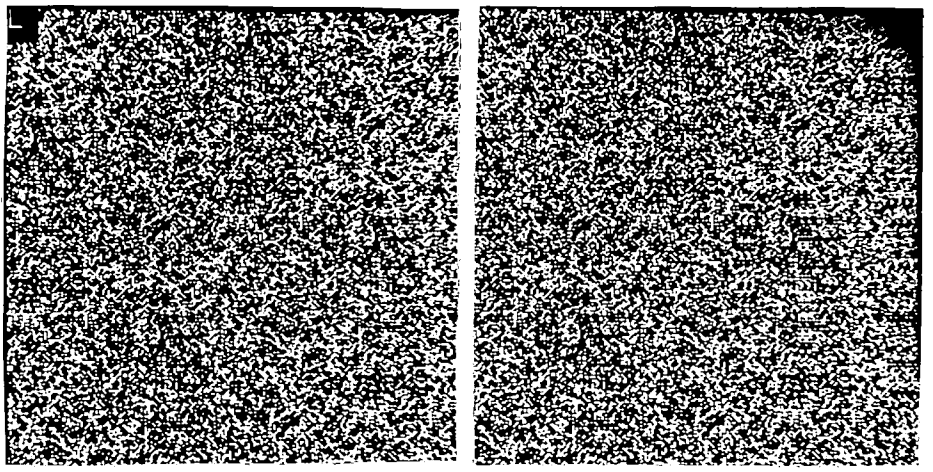


Figure 5.21 A random dot stereogram showing two raised square blocks.

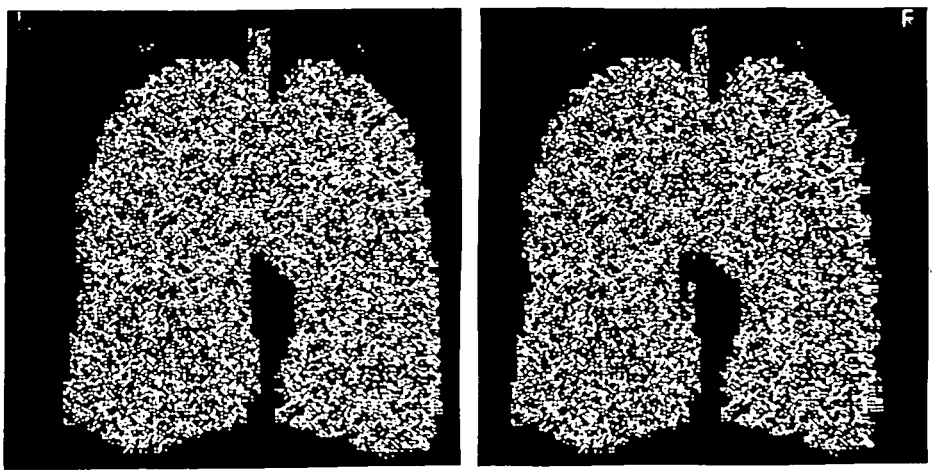


Figure 5.22 A random dot stereogram showing a pair of lungs in the anterior projection.

the same effect, although to a lesser extent. For comfortable stereopsis the angular disparity should not exceed 1 to 2 degrees (Braddick 1982). For a screen of size 12.5 cm viewed at 25 cm distance, a shift of 64 pixels in 1024 pixels corresponds to an angular disparity of approximately 1.8^{o}. Even at that screen resolution, shifts of 1 pixel would produce an apparent depth displacement of 0.5 mm, many times as large as the minimum detectable displacement, and hence the quantised depth effect would still be noticeable. Since only horizontal shifts are required, a screen memory with high horizontal resolution but only medium vertical resolution would provide improved performance, e.g. 2048 x 512 pixels rather than 1024 x 1024 pixels. Although this effect has been illustrated using random dot stereograms, it will still be present using other forms of stereoscopy with raster displays.

5.6 Conclusions

The purpose of surface shading is to convey to the observer the orientation of surfaces, and hence the shape of objects. A surface shading model was developed which formed a consistent framework for the description and comparison of several algorithms. This model used two light sources, providing both parallel directional illumination and isotropic diffuse illumination. For opaque diffuse reflectors, Lambert's Law states that the surface brightness varies as the cosine of the angle of incidence. Empirical modifications to this 'cosine rule' have been proposed, and the disadvantages of some of these were demonstrated. The most realistic illumination falls from above, simulating the natural world, and this can only be implemented using a pure cosine or cosine

power law expression.

Almost all the algorithms required an estimate of the surface normal vector. This can be obtained using object space coordinates, for all points on the surface of the object. However, objects represented using cuberille models have only a limited number of possible surface orientations. An alternative approach, devised for this work and also developed independently, estimates the surface normal from the observer space coordinates after the visible voxels have been determined. This approach has the advantage that no additional data storage is required for surface descriptors.

The estimation of surface normals from an array of visible voxel depth data requires knowledge of the surface gradients along the x' and y' axes. Three gradient estimators were compared using a one dimensional semi-elliptical phantom. Because the true gradient can be calculated at all points, the accuracy of the gradient estimate can be defined in terms of the angle between the true and the estimated surface normals. A polynomial gradient estimator was devised for this work, and was shown to be more accurate than either of two simpler methods, especially for noisy data. The same pattern was seen for elliptical objects with a wide range of sizes and curvatures. Because pixels in the depth array are equally spaced along the x' and y' axes, calculation of the gradient of a best fit polynomial was performed using a simple 5 or 3 point convolution.

However, shading schemes also have a subjective component, and to assess this a panel of judges was asked to rank images produced using four shading techniques. These ranged from depth shading with no surface orientation information (S_d), to a pure cosine law model with no depth information (S_c). Two intermediate forms were also included, a modified

cosine law technique published by other workers (S_{mc}), and the polynomial surface shading devised for this work (S_p). There was a clear preference for the techniques with the greatest surface modulation (S_c and S_p), using a range of objects from two imaging modalities. The polynomial surface shading scheme was the overall preferred technique, although one of the eight observers preferred pure cosine shading. For some of the more irregular objects at certain aspects excessive surface modulation could be confusing, and in these circumstances the observers failed to agree on a preferred technique. There is therefore a case for always making available at least two shading schemes, to allow the observer to select the scheme best suited to the particular image being displayed. Of the four schemes considered, the S_p/S_d pair would be the most appropriate, since depth shading can be calculated extremely quickly, and the corresponding images would form a good contrast with the surface orientation information present in the polynomial shaded images.

As well as monochrome images, colour shading schemes have also been devised. These were most successful when pure hues were used as arbitrary surface properties, to identify parts of a complex object. This was particularly valuable for the identification of complex surfaces in a composite object showing cardiac wall motion. Colour images were less successful when a range of hues with varying intensity values was used to combine functional and structural information. These images were able to summarise large amounts of data, but required careful interpretation.

The shading model devised for opaque surfaces was extended to transparent surfaces, and for the irregular objects found in medicine a model using specular reflection of isotropic illumination was found to

give the best results. The quality of the images produced using NMR data confirmed that monochrome shading could be used to display nested structures. This model was further developed with the addition of colour, which facilitated the identification of multiple surfaces.

Stereoscopy was investigated briefly, but the problems of using a raster display memory with only medium resolution were soon revealed. Even a single pixel discrepancy between the left eye and right eye images corresponded to a depth difference which was much greater than the limit of human stereoacuity. Consequently the stereo images produced suffered from a 'depth quantisation' effect, appearing as a stack of separate vertical planes. Stereoscopic display is an area which would merit further investigation, but which requires a screen memory with a horizontal resolution of at least 1024 pixels.

6. Clinical Applications

6.1 Implementation

The basic philosophy of this project was to develop methods for the display of shaded surface images using equipment currently available in the hospital environment. Consequently the methods described above for three dimensional display were implemented using standard medical image processing systems.

The two systems used were both intended for the acquisition and analysis of nuclear medicine images, but can also be used for general purpose image processing. Details of the two systems are shown in Table 6.1. Conversion of software written for the older system (Dyanne) to run on the newer system (MAPS) was facilitated by the close compatibility between the two systems. Source code written in FORTRAN required only minimal changes and re-compilation, whilst that written in ASSEMBLER required re-assembly only. Both systems used identical display hardware, consisting of an image memory with 256 x 256 pixels, a programmable colour look-up table, and a trackerball.

The choice of programming languages was partly dictated by availability, rather than intrinsic worth. Most medical imaging systems are supplied with FORTRAN, and possibly BASIC. The Dyanne and MAPS units were additionally provided with ASSEMBLER, together with a library of FORTRAN callable subroutines for accessing image data and the display hardware. Hence the shell of each program was written in FORTRAN, with additional ASSEMBLER subroutines where increased speed was necessary. In the latter case the subroutines were written so that they could be called from within a FORTRAN program.

The algorithms developed for this work were implemented on one or both of the systems described in Table 6.1, and in some cases several

Table 6.1 Details of the image processing systems used for this project

Parameter	System 1	System 2
Name	Dyanne	MAPS2000
CPU	NOVA 4x	Fairchild 9445
Clock Speed	5 MHz	20 MHz
Main Memory	256 Kbytes	256 Kbytes
Disk Memory	26 Mbytes	16 Mbytes
Image Memory	64 Kbytes	64 Kbytes

a) Both systems were manufactured by Link Analytical Limited.

b) Only 64 Kbytes of memory was directly accessible using FORTRAN.

c) Neither system had a hardware multiply/divide unit.

versions were written to evaluate different object representation schemes. Rather than include complete listings of all this software, much of which is highly specific to the systems used, selected listings have been included in Appendix 3. The listings chosen illustrate particular features of the algorithms and include much which is device independent. Each listing is documented with details of the input data, action taken and output data produced. The listings are :-

Appendix 3a : Binary Object hidden surface algorithm.

Appendix 3b : Binary Object manipulation/slicing algorithm.

Appendix 3c : Creation of look up tables for coordinate transformation.

Appendix 3d : Coordinate transformation of BTF lists.

Appendix 3e : Polynomial shading of visible voxel data.

Appendix 3f : Transfer of image data to the display memory.

The main aim of this work was to develop and implement techniques, rather than to assess them in clinical practice. Nevertheless, a number of successful applications were made, and these are described below.

6.2 Myocardial Perfusion Imaging

The radionuclide Thallium-201 has been shown to act as a potassium analogue, and when injected intravenously it is distributed in muscle tissue in proportion to local perfusion. The technique of Thallium-201 scanning has been developed for the study of regional myocardial perfusion (Holman 1979). Areas of infarction are permanently damaged tissues, where previous coronary artery disease has led to loss of muscle function. These areas are not perfused, and hence show no uptake of Thallium-201. In areas of ischaemia, however, perfusion is reduced

but permanent tissue damage has not yet occurred. Thallium-201 is normally injected while the heart is being stressed, e.g. during exercise, or following intravenous dipyridamole (Harris et al 1982). Ischaemic areas will show reduced or absent uptake, since they are unable to compete with better perfused areas when all the myocardium requires increased blood flow. Images taken immediately after injection therefore show both ischaemic and infarcted tissue as areas of reduced uptake. However, after 2 to 3 hours the Thallium-201 re-distributes within the myocardium, and will enter ischaemic areas, though not infarcts. Delayed images show the distribution of Thallium-201 which would have been obtained if the injection had been given at rest (Pohost et al 1977) and can be compared to the stress images to diagnose ischaemia. Because of the smaller amount of muscle present, the right ventricle is often not visualised.

Tomographic Thallium-201 imaging has been shown to provide improved sensitivity and specificity for the detection of ischaemia due to arterial disease (Ritchie et al 1982). It has also been suggested that the localisation of observed defects using tomographic images can indicate which of the coronary arteries is diseased (Tamaki et al 1984). Three dimensional display techniques were applied to tomographic Thallium-201 data in an attempt to improve the perception of the size, shape and position of defects in relation to the coronary arteries.

Figure 6.1 shows transverse sections through an ischaemic myocardium, obtained after exercise, and in the resting state. The latter images show the normal 'horseshoe' appearance of a section through the left ventricle, indicating that the defect observed on exercise is due to reversible ischaemia rather than irreversible infarction. Conventional transverse sections are oblique to the

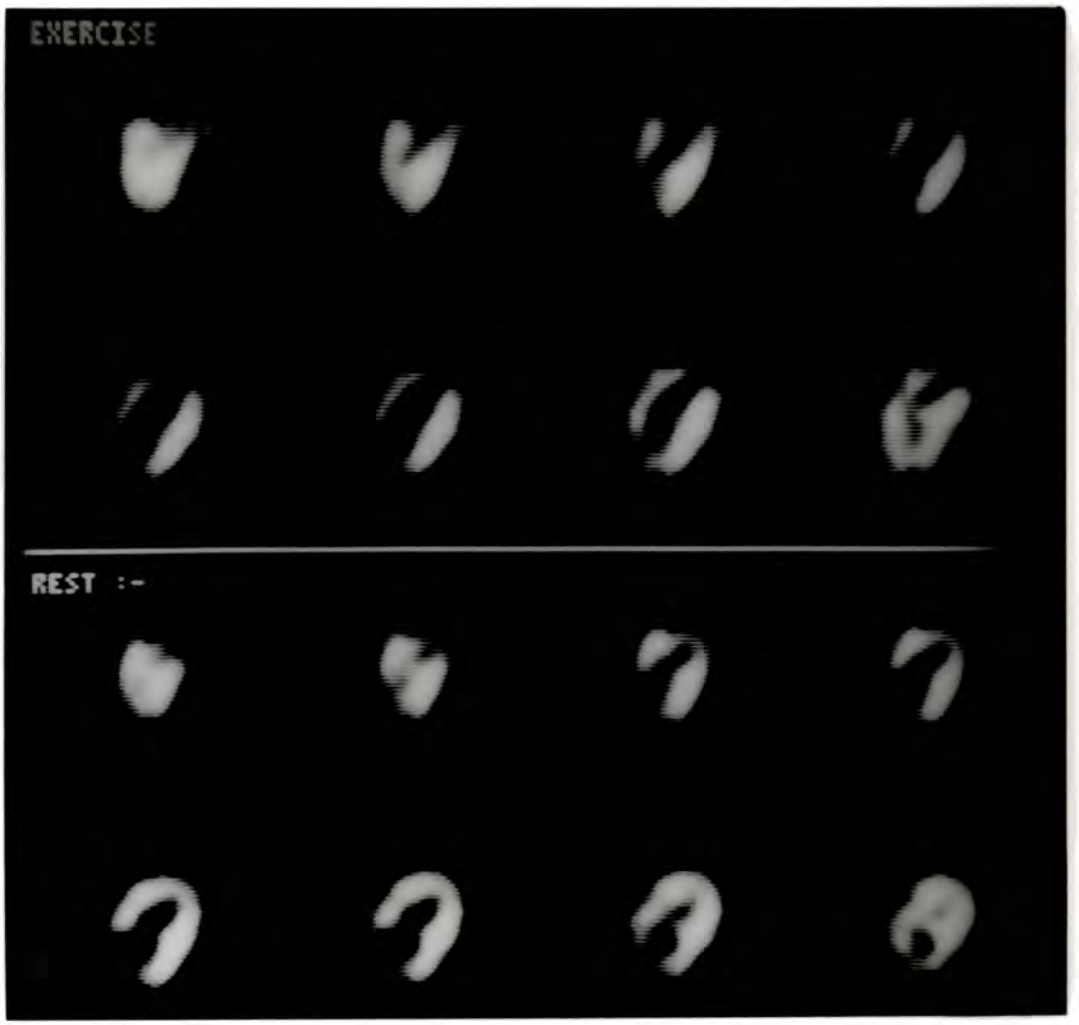


Figure 6.1 Transverse section tomograms through an ischaemic myocardium following injection of Thallium-201 :-

Upper - After exercise, showing an apical defect.
Lower - At rest, showing a normal 'horseshoe' shape.

anatomical axes of the left ventricle, and this causes problems in localising observed defects. A set of contiguous transverse slices can be re-formatted to obtain oblique slices parallel and perpendicular to the long axis of the ventricle (Olson et al 1981). These 'doughnut' sections are more easily interpreted (Figure 6.2). However, three dimensional display obviates the need for reformatting by allowing the observer to examine the heart from any aspect (Gibson et al 1985). Figure 6.3 shows a normal myocardium viewed from several orientations. The segmentation threshold has been chosen to eliminate activity in the surrounding tissues and in the right ventricle, leaving a cup-shaped object. Shaded surface images at several aspects demonstrate reasonably uniform myocardial perfusion at all points around the chamber wall. In contrast, Figure 6.4 shows an infarcted myocardium, with a large defect extending down the superior surface to the apex. The exact location and extent of this lesion was shown more clearly using three dimensional display. Ischaemia can also be demonstrated using shaded surface images (Figure 6.5), by comparing exercise and resting images at the same orientation. These images show a large defect at exercise, extending from the superior through to the inferior myocardial surface. The same images taken at rest, however, show that the defect has almost entirely resolved.

The left ventricular myocardium is a shape with relatively little detail, providing few anatomical cues to orientate the observer. For this reason, and to determine whether observed defects correlated with arterial anatomy, schematic outlines of the coronary arteries were superimposed on the myocardial surface image (Gibson et al 1985). The left ventricular myocardium is supplied with blood by three main arteries, the left anterior descending artery (LAD), right coronary

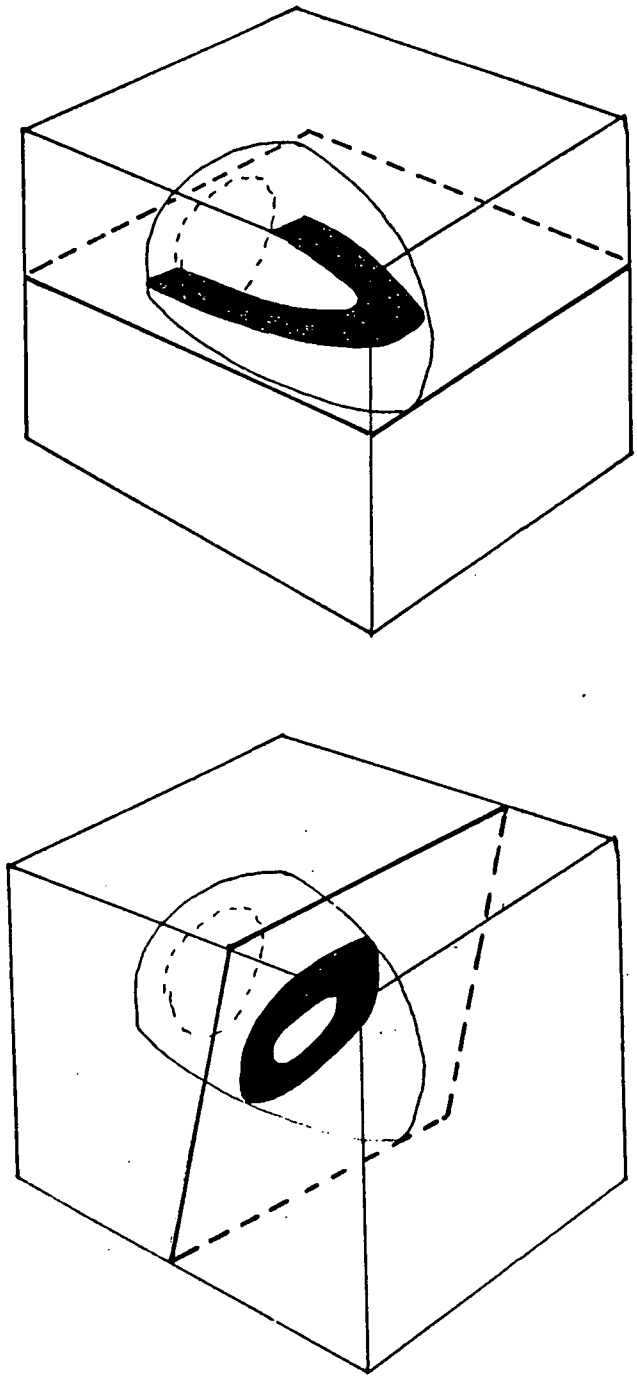


Figure 6.2 Reformatting transverse sections through the chest to obtain oblique slices which are perpendicular to the long axis of the left ventricular myocardium.

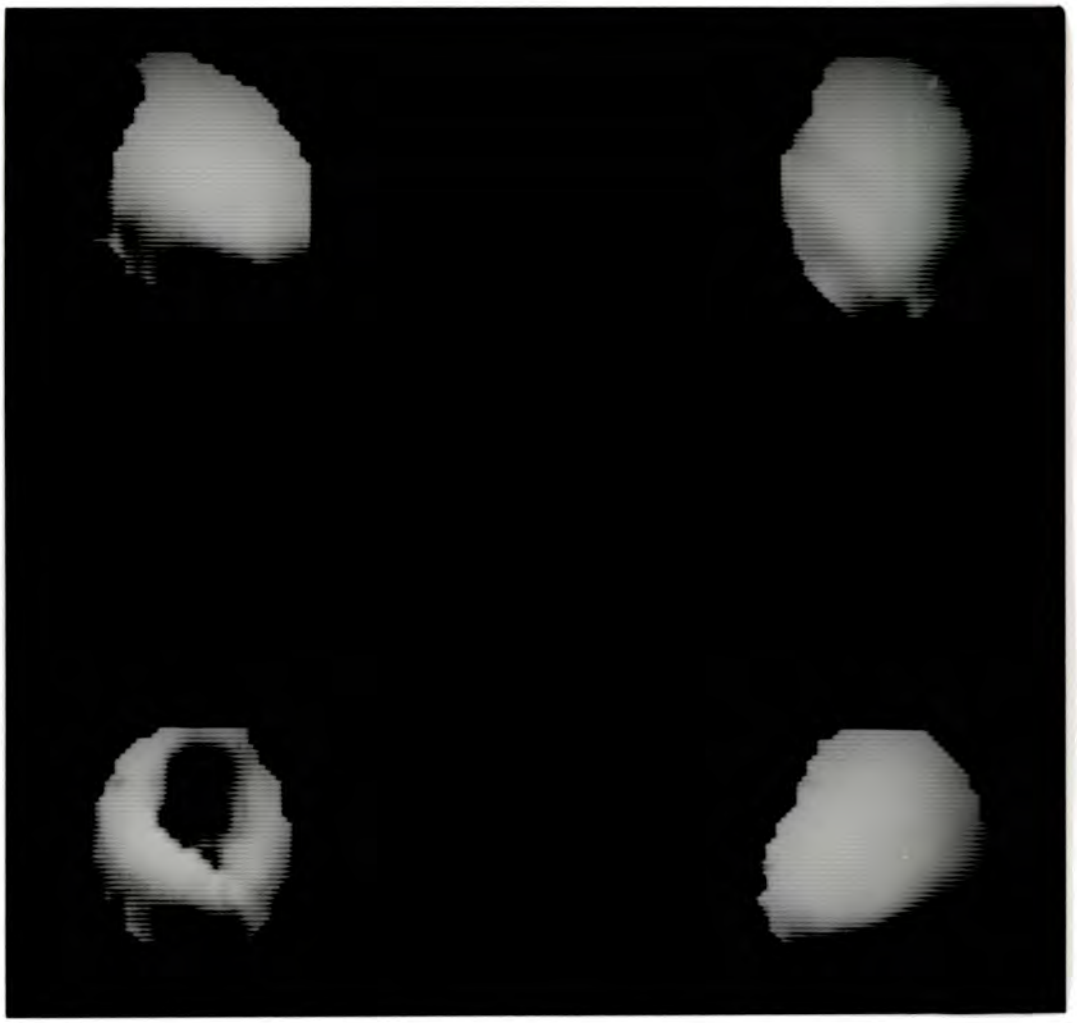


Figure 6.3 Shaded surface images of a normal myocardium, showing uniform perfusion throughout the ventricular wall :-

Upper left - Anterior.
Upper right - Left anterior oblique.
Lower left - Right posterior oblique.
Lower right - Left lateral.

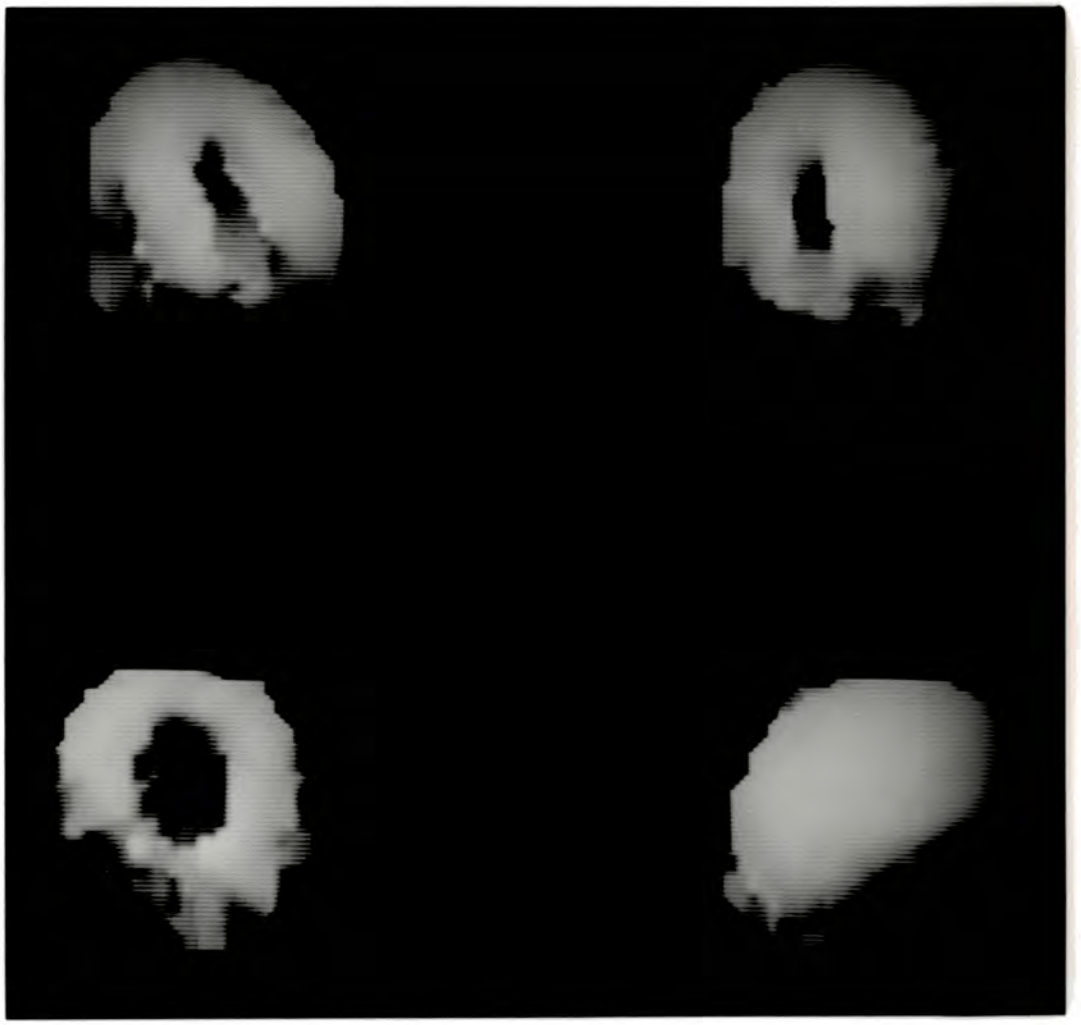


Figure 6.4 Shaded surface images of an infarcted myocardium, showing a large defect in the superior wall :-

Upper left - Anterior.
Upper right - Left anterior oblique.
Lower left - Right posterior oblique.
Lower right - Left lateral.

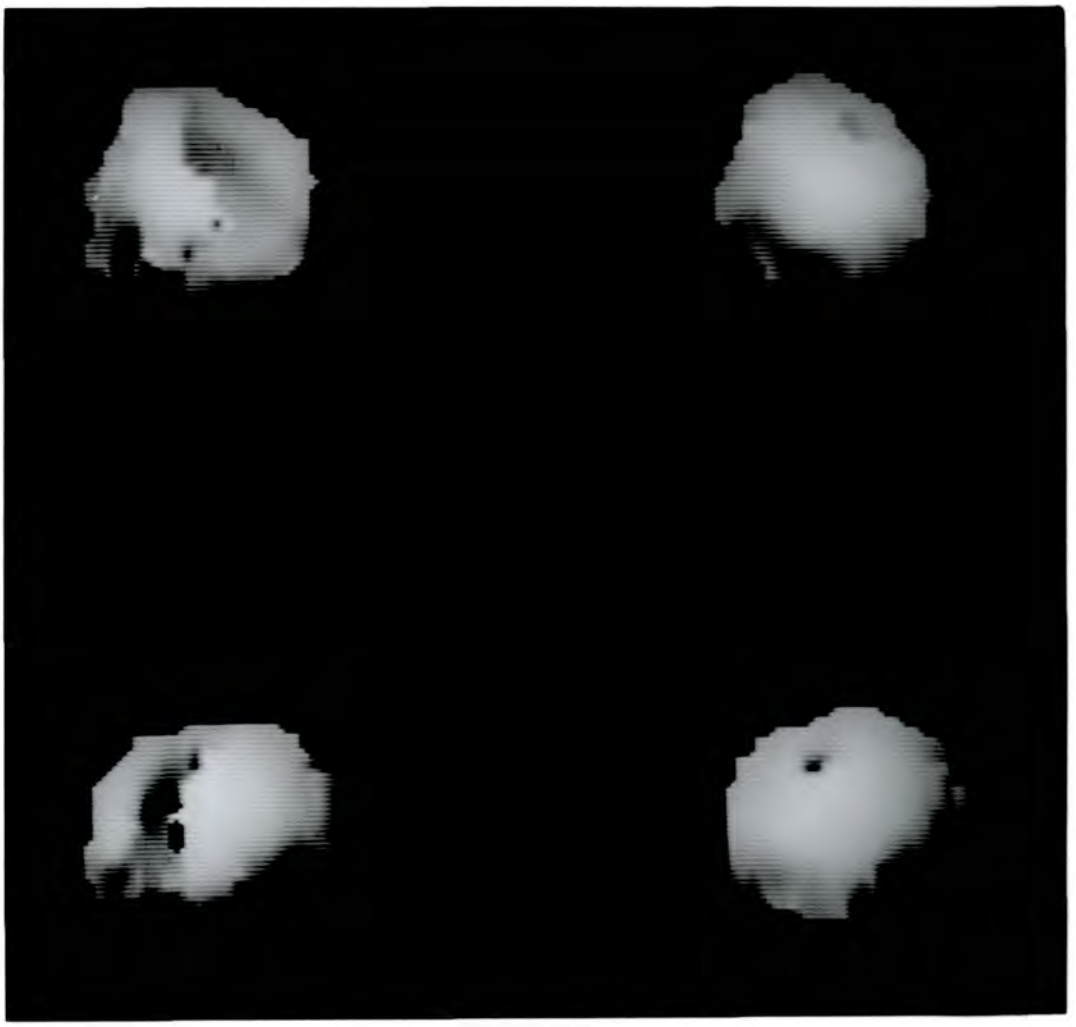


Figure 6.5 Shaded surface images of an ischaemic myocardium after exercise and at rest :-

Upper left - Anterior after exercise, markedly abnormal.
Upper right - Anterior at rest, now almost normal.
Lower left - Left lateral after exercise.
Lower right - Left lateral at rest.

artery (RCA) and left circumflex artery (LCX). A resin cast was obtained showing normal coronary arteries, and using a Reflex Metrograph the positions of the aortic node (the root of the coronary arteries), and the left ventricular apex were digitised in three dimensions. A series of points along each of the main arteries was similarly obtained. The techniques of Section 4 were used to display the digitised arteries from any aspect. For simplicity, the arteries were approximated by straight line segments of constant thickness between the sample points. These segments were shaded so that their intensity was proportional to the cosine of the angle between the line segment and the incident light vector. An arbitrary hue was also assigned to each artery, showing the RCA in red, LAD in green and LCX in blue. Figure 6.6 shows four views of the digitised arteries, in aspects corresponding to those of Figure 6.3. The shape of the digitised arterial model corresponds to that of the original arterial cast, and the cosine shading of the arterial segments provides sufficient visual cues to give an impression of depth.

To facilitate comparison of myocardial defects with arterial anatomy, a technique was developed for superposition of the arterial tree around individual myocardial surfaces. The digitised coordinates of the coronary artery node (**n**) and the ventricular apex (**a**) specified the location and orientation of the arterial tree. Similar points (**N** and **A**) were marked by the observer from the myocardial surface, using a pair of orthogonal views. A transformation matrix was then deduced which made points **n** and **a** coincident with **N** and **A**, without allowing rotation about the line joining **n** and **a**. Figure 6.7 shows orthogonal views of the left ventricle in the anterior/left lateral aspects, and in the left posterior oblique/right posterior oblique aspects. Using one pair, the operator indicated the location of points **N** and **A** on both

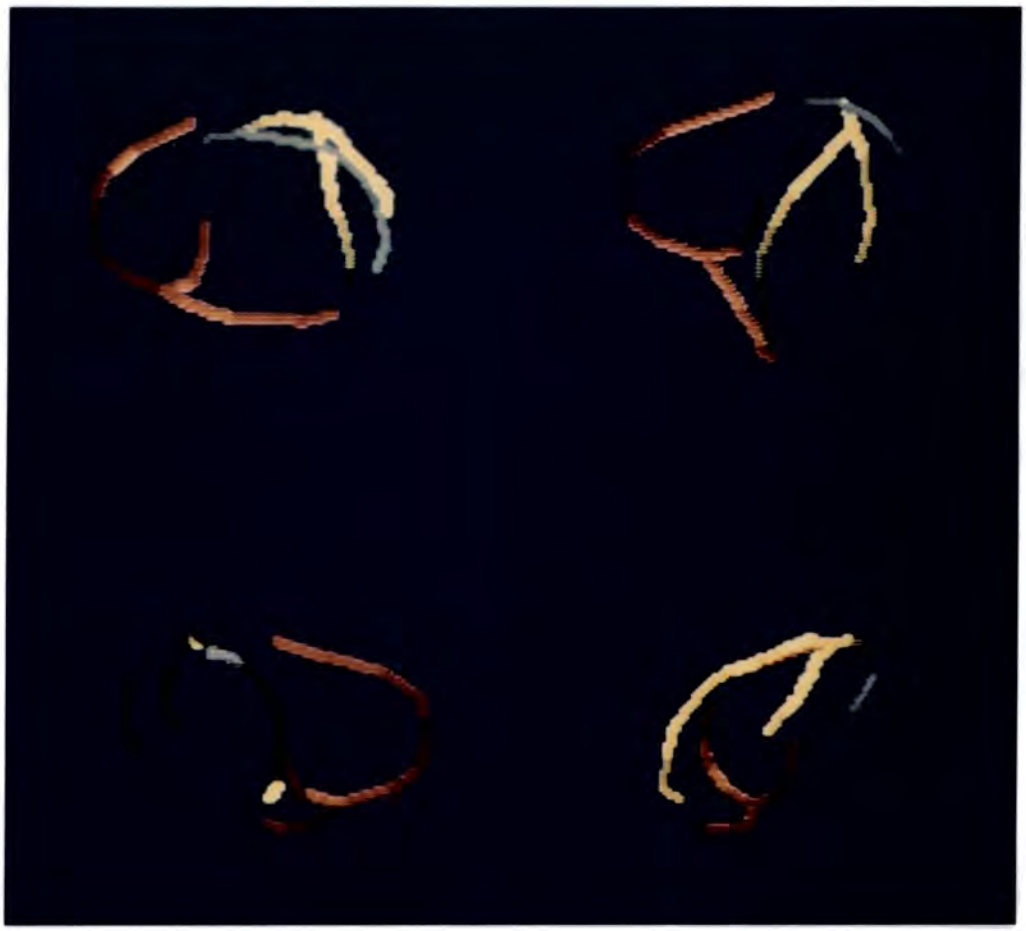


Figure 6.6 Four views of the digitised arterial tree, showing the RCA in red, the LAD in green and the LCX in blue :-

Upper left - Anterior.
Upper right - Left anterior oblique.
Lower left - Right posterior oblique.
Lower right - Left lateral.

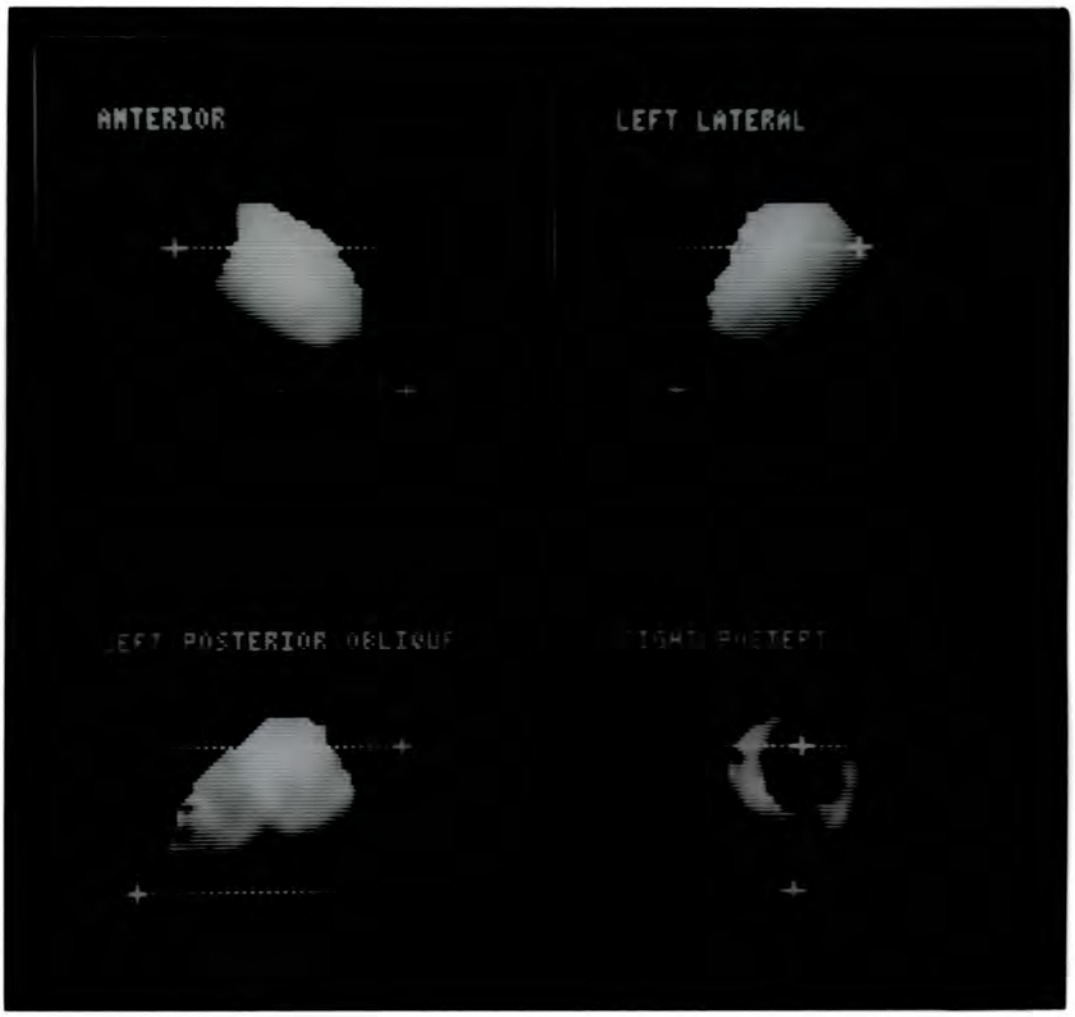


Figure 6.7 Arterial fitting by interactive location of the fixed points (shown by crosses), using two orthogonal images :-

Upper - Using the anterior and left lateral pair the coordinates can be obtained directly.
Lower - Other pairs provide better definition of the location of the left ventricle.

views with the trackerball. The (xyz) coordinates of **N** and **A** were obtained directly if the anterior/left lateral pair was used. However, it was found to be both easier and more repeatable to use the left posterior oblique/right posterior oblique pair. In this case the coordinates can be deduced from the trackerball location using simple trigonometry.

The sequence of elemental transformations which render **n** and **a** coincident with **N** and **A** is illustrated in Figure 6.8. The first rotation is about a vertical axis, until the projections of the lines **n-a** and **N-A** on the xy plane are parallel. The second rotation is about a horizontal axis, until the lines **n-a** and **N-A** are themselves parallel. After scaling the arteries until the distance from **n** to **a** is equal to that from **N** to **A**, the arterial tree is translated until **n** is coincident with **N**. Points **a** and **A** will also be coincident after this translation. These steps can be concatenated to give a single transformation matrix, which is applied to the digitised arterial tree to superimpose it around the myocardium (See Appendix 2). In a real heart the LAD and LCX arteries run along the surface of the left ventricle, whereas the superimposed arteries lie just above the surface. Attempts to 'tighten up' the arterial matching have been unsuccessful, since parts of the arteries disappear beneath the myocardial surface.

The results of this technique will be illustrated with four case histories:-

Case 1 - Normal A man aged 46 was admitted to hospital following a 'collapse'. An exercise electrocardiogram was performed, together with a Thallium-201 scan. During the exercise test he suffered no chest pain, and there were no electrocardiographic changes. Tomographic thallium images showed no evidence of ischaemia, and three dimensional

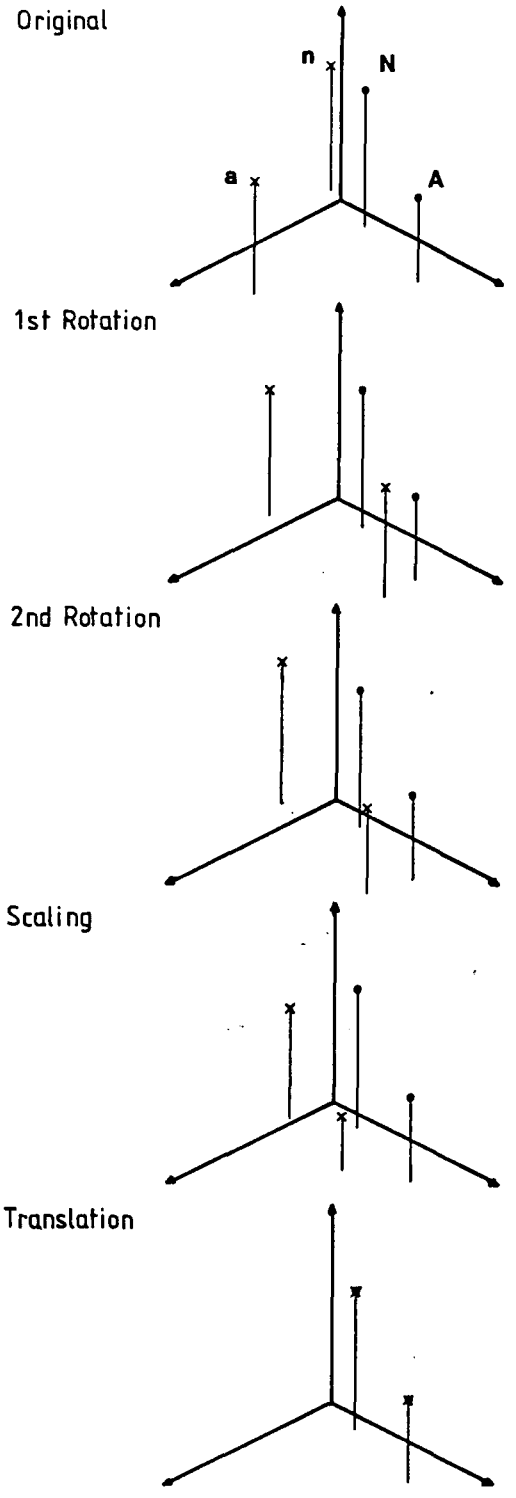


Figure 6.8 The sequence of transformations described in Appendix 2, which render the points n and a (crosses) coincident with points N and A (circles).

images also showed no defects on either the exercise or the resting images (Figure 6.9).

Case 2 - Infarction A man aged 38 with a previous history of infarction presented with chest pain. An exercise electrocardiogram was terminated because of this symptom, but no electrocardiographic changes were seen. Coronary angiography was performed, and the LAD artery was shown to be completely occluded just beyond the first diagonal branch. The tomographic thallium images were interpreted as showing a large apical defect at exercise and at rest. Three dimensional display (Figure 6.10) reveals that the defect extends from the superior surface around the apex, and involves some of the inferior surface also. This defect is almost exactly coincident with the LAD artery after the first diagonal branch, and the results are therefore in excellent agreement with the angiographic findings.

Case 3 - Ischaemia A man aged 50 with mild hypertension was continuing to suffer angina of effort, despite medication over four months. The coronary angiogram showed several areas of arterial disease. The RCA was occluded beyond the right ventricular branch, but the distal portion filled via collateral vessels. The LAD showed several regions of stenosis with distal occlusion, and again collateral filling was observed. The LCX had regions of severe stenosis distal to the obtuse marginal branch. The interpretation of Thallium-201 images in cases of triple-vessel disease is difficult. Three dimensional images (Figure 6.11) show defects in the septal and apical regions on exercise. These are indicative of probable RCA and LAD disease, in the latter case affecting the distal portion. The septal defect is considerably smaller at rest, and the apical defect is not seen. This may be interpreted as

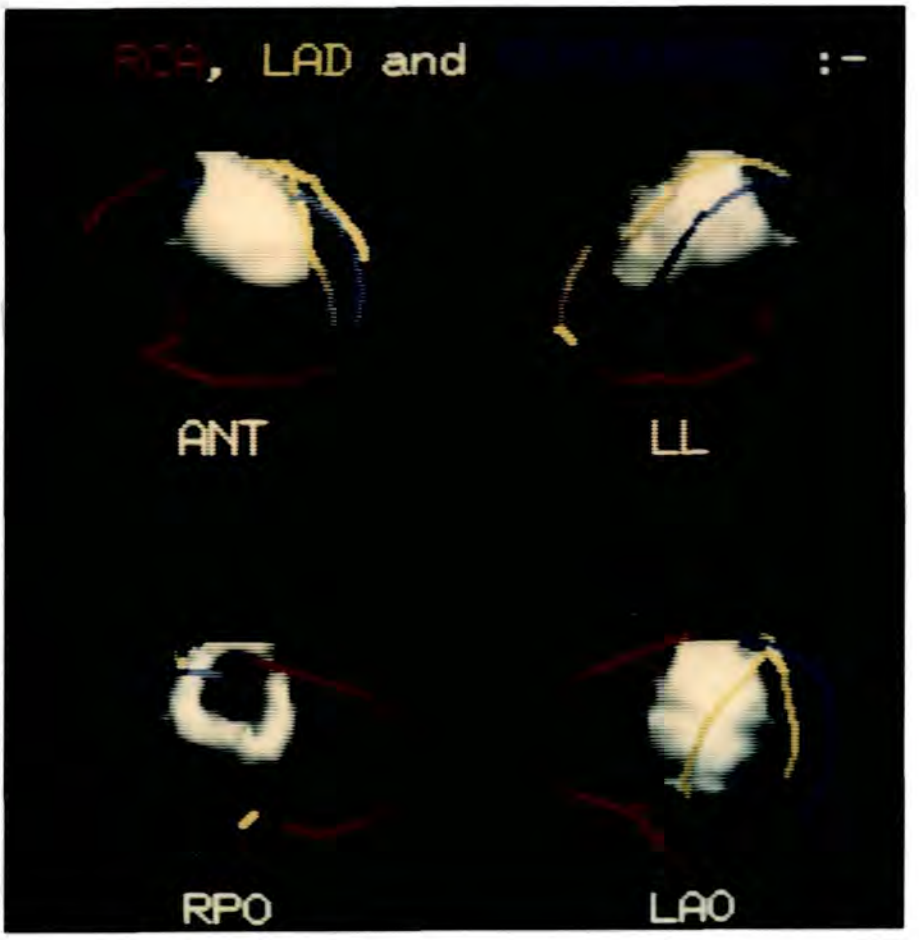


Figure 6.9 Case 1 - Shaded surface images of a normal myocardium with superimposed coronary arteries. Colour coding of arteries as for Figure 6.6 :-

Upper left - Anterior.
Upper right - Left lateral.
Lower left - Right posterior oblique.
Lower right - Left posterior oblique.

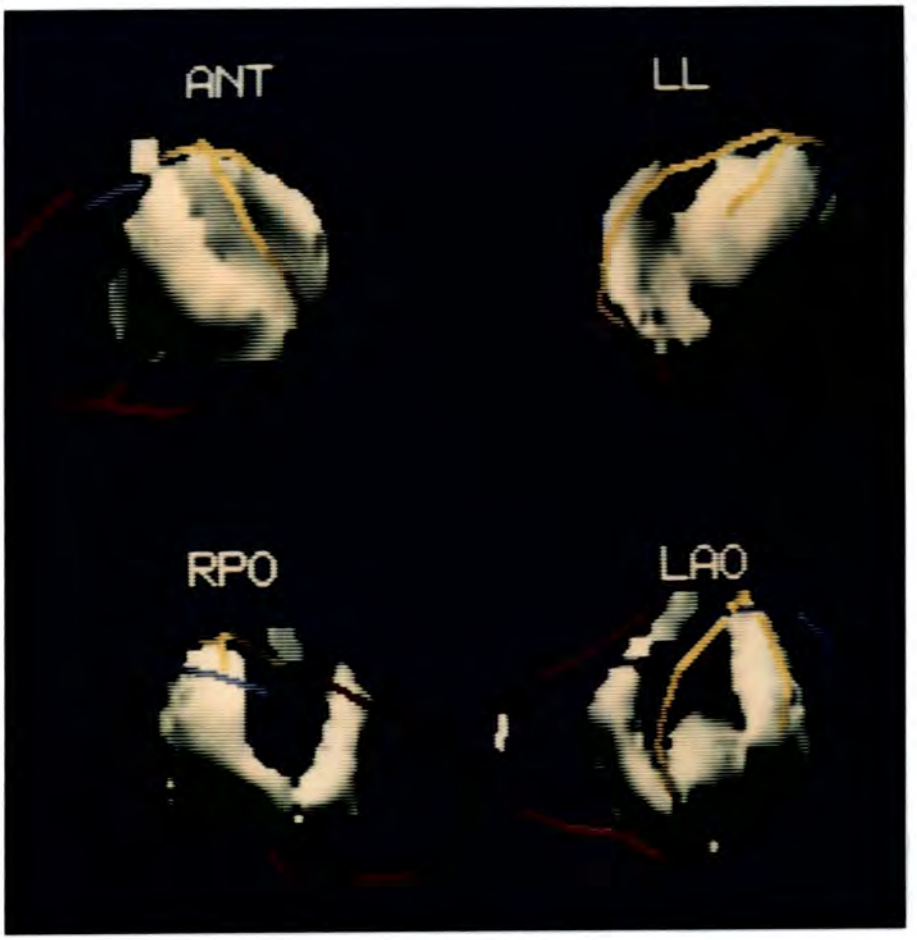


Figure 6.10 Case 2 - Shaded surface images of an infarcted myocardium with superimposed coronary arteries. Colour coding of arteries as for Figure 6.6 :-

Upper left - Anterior.
Upper right - Left lateral.
Lower left - Right posterior oblique.
Lower right - Left anterior oblique.

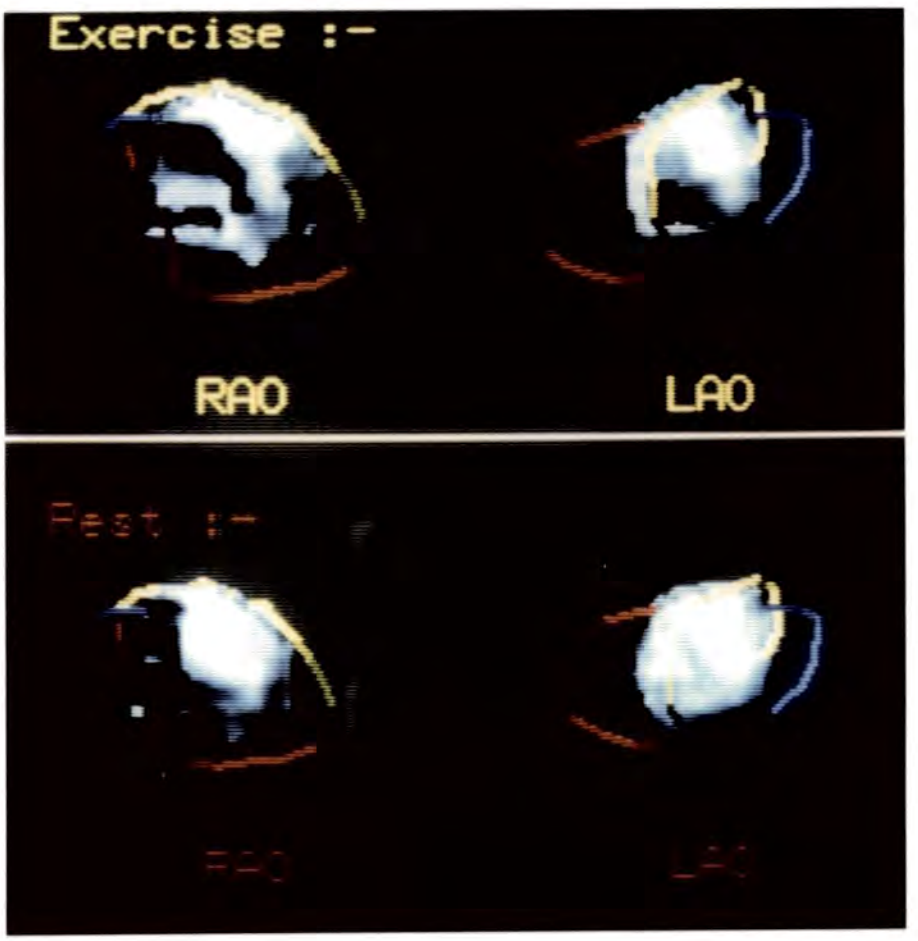


Figure 6.11 Case 3 - Shaded surface images of an ischaemic myocardium with superimposed coronary arteries. Colour coding of arteries as for Figure 6.6 :-

Upper - Oblique images under stress conditions.
Lower - Corresponding images at rest.

showing that the collateral circulation seen during angiography is adequate at rest, but is insufficient to meet the needs of the myocardium during stress. This interpretation is consistent with the patient's symptoms, and shows how functional as well as anatomical information can be provided by three dimensional Thallium-201 images.

Case 4 - Probable ischaemia A man aged 52 had angina of effort, and sometimes at rest, for a period of one year. An exercise electrocardiogram showed ST segment depression, and he is awaiting coronary angiography. Thallium-201 tomograms showed a large apical defect on exercise, which was not present at rest. Three dimensional reconstruction (Figure 6.12) indicated that the defect involved the inferior surface also, and was almost, though not completely, re-perfused at rest. The site of the main defect is exactly coincident with the LAD artery, and these images unequivocally suggest arterial disease at that site. The patient continues to show clinical symptoms of ischaemic heart disease, but is still not well enough to undergo angiography.

Two assumptions have been made in order to create three dimensional images of myocardial perfusion. Firstly, it is necessary to assume that the distribution of Thallium-201 contains definite areas of uptake which may be enclosed by an isocount contour. In practice the uptake often varies continuously from low to high values. The simple thresholding technique used here can distinguish areas of appreciably reduced uptake, but will probably fail to detect small areas of slightly reduced uptake, e.g. due to intramural infarction. Secondly, the spatial distribution of the coronary arteries corresponds to that of one normal subject only. The arteries displayed on each image have been scaled and rotated to

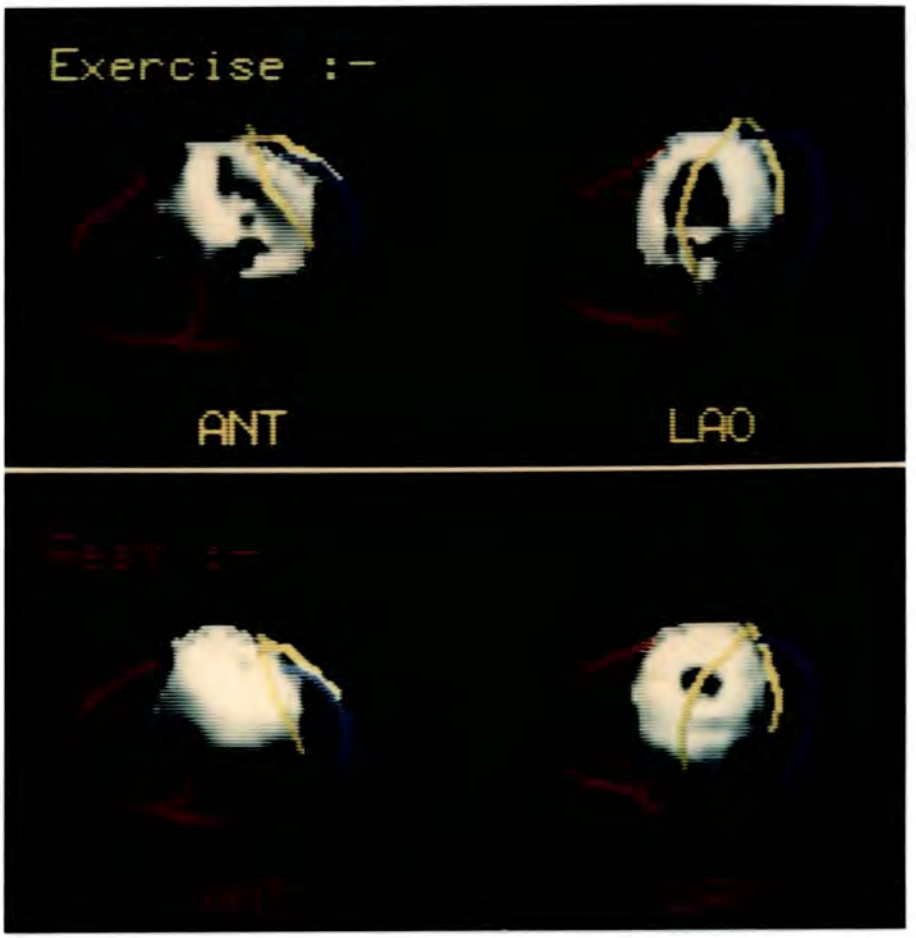


Figure 6.12 Case 4 - Shaded surface images of a myocardium indicating probable ischaemia, with superimposed coronary arteries. Colour coding of arteries as for Figure 6.6 :-

Upper - Anterior and oblique images under stress.
Lower - Corresponding images at rest.

match the individual myocardial surfaces, but they should only be interpreted as indicating the probable site of the actual coronary arteries. Nevertheless, three dimensional myocardial imaging provides a readily comprehensible summary of the shape, location and extent of perfusion defects, and may be of particular value for the evaluation of patients who are unable to undergo more invasive procedures.

6.3 Cardiac Blood Pool Imaging

The technique of gated blood pool scanning is now an established diagnostic tool (Strauss et al 1979). The patient's red blood cells can be conveniently labelled in vivo using the short lived radionuclide Technetium-99m (Pavel et al 1977). Static gamma camera images of the blood pool may then be obtained to give some information about the size and shape of the cardiac chambers. However, the principal use of blood pool imaging is for the evaluation of ventricular function using gated imaging (Folland et al 1977). In this technique images are obtained in synchrony with the electrocardiographic signals received from the heart. A typical set of 16 images is acquired over 5 to 10 minutes. The set represents one cardiac cycle (one beat), and is formed by summation of data received during many individual beats. Images obtained in the left anterior oblique projection show the best separation of the left and right ventricles.

Analysis of these images is both quantitative, to determine the ejection fraction, and qualitative, to determine whether areas of the ventricular wall have reduced, absent, or paradoxical motion. In the last case, a portion of ventricular surface bulges outwards while the rest of the ventricle contracts, indicating the presence of an aneurysm. However, it is not possible to assess all parts of the ventricular wall

from any single view, and so several sets of images must be obtained at different aspects.

Tomographic blood pool imaging offers the possibility of three dimensional reconstruction of the cardiac chambers. If ungated tomograms are obtained, the results will show the average volume and position of the chambers. More usefully, however, gated blood pool tomography allows the cardiologist to examine a four dimensional model of the heart, with complete information about wall movement (Moore et al 1980).

As was the case for myocardial tomograms, conventional transverse sections are oblique to the anatomical axes of both ventricles. The problem of interpretation is far worse, however, because the ventricle changes both shape and position during the cardiac cycle. A slice fixed relative to the chest wall will contain different parts of the ventricle during the cycle, and during some periods it may even contain no ventricle at all. The assessment of wall motion can be aided using reformatted 'doughnut' sections, but this will require an extremely large number of images. A typical data set would consist of sixteen transverse slices, eight slices parallel to the long axis of the ventricle, and twelve slices parallel to the short axis. If this is repeated for both end-diastolic and end-systolic time periods (i.e. dilated and contracted ventricles), then the observer must examine 72 images.

Three dimensional display techniques were applied to ungated and gated blood pool tomograms, in an attempt to improve the perception of chamber size and ventricular function respectively.

Ungated tomograms: Anterior views obtained from four patients are shown in Figure 6.13. They show a normal appearance, some ventricular

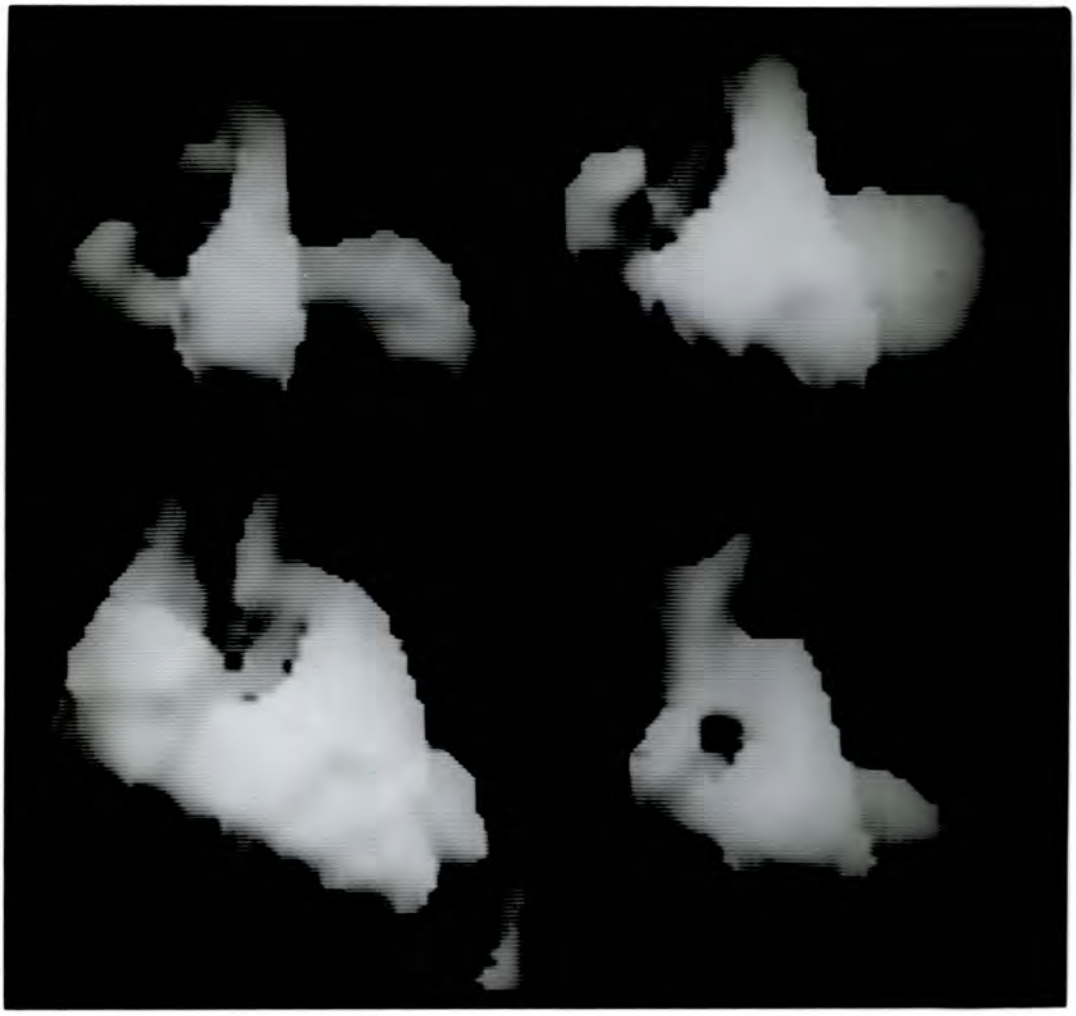


Figure 6.13 Four ungated blood pool images in the anterior projection :-

Upper left - Normal.
Upper right - Some ventricular enlargement.
Lower left - Marked atrial enlargement.
Lower right - Right ventricular enlargement.

enlargement, marked atrial enlargement, and enlargement of the right ventricle alone. These images confirm that shaded surface techniques are extremely effective for conveying information about both shape and size. The use of an isocount threshold to separate object from background is readily justifiable for blood pool images. After an initial mixing period the concentration of radioactive material in the blood is reasonably constant throughout the body. An isocount threshold isolates only those structures where the volume density of blood is above some corresponding level, i.e. the heart and great vessels. There will be little variation of count density within the selected isosurface, and so very little useful information is lost when the surface representation is created (Jackson et al 1984).

Gated tomograms: Each set of tomograms represents the blood pool at a fixed time interval. Identical segmentation criteria must therefore be applied to each set to enable valid comparisons to be made between three dimensional representations. Figure 6.14 shows sixteen images in the anterior projection, indicating the movement of the ventricular and atrial surfaces. The data can be rotated and tilted to facilitate evaluation of ventricular wall motion (Figure 6.15). The rapid display techniques described in Section 4.6 were principally developed for the display of gated blood pool images (Gibson 1987). The modified BTF algorithm enables the observer to quickly select any viewing position, and to oscillate between end-diastolic and end-systolic representations at that aspect. Such kinetic displays are extremely useful, since the motion parallax effect provides an additional depth cue.

However, hard copy images are still required for recording purposes, and colour techniques were developed to facilitate this.

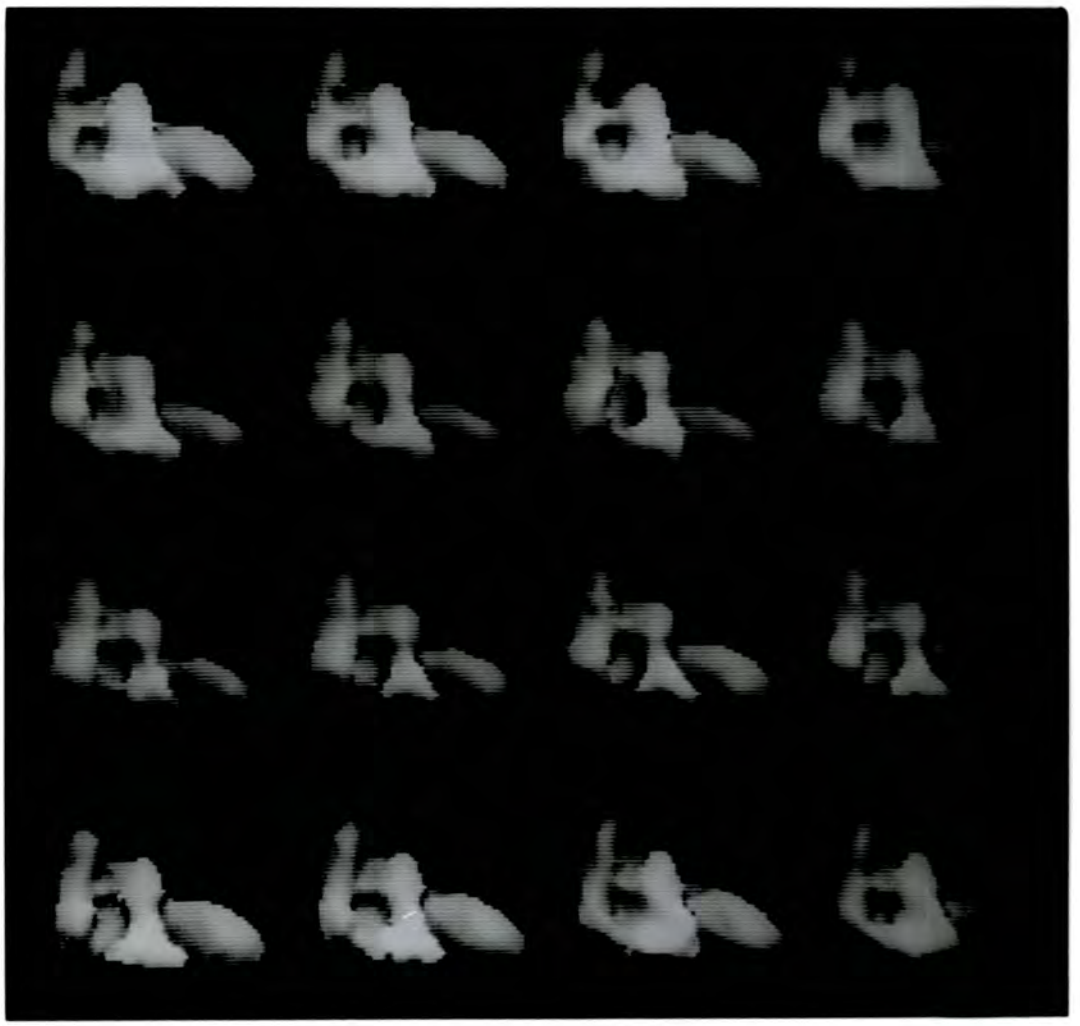


Figure 6.14 Sixteen gated blood pool images in the anterior projection, showing one heart cycle.

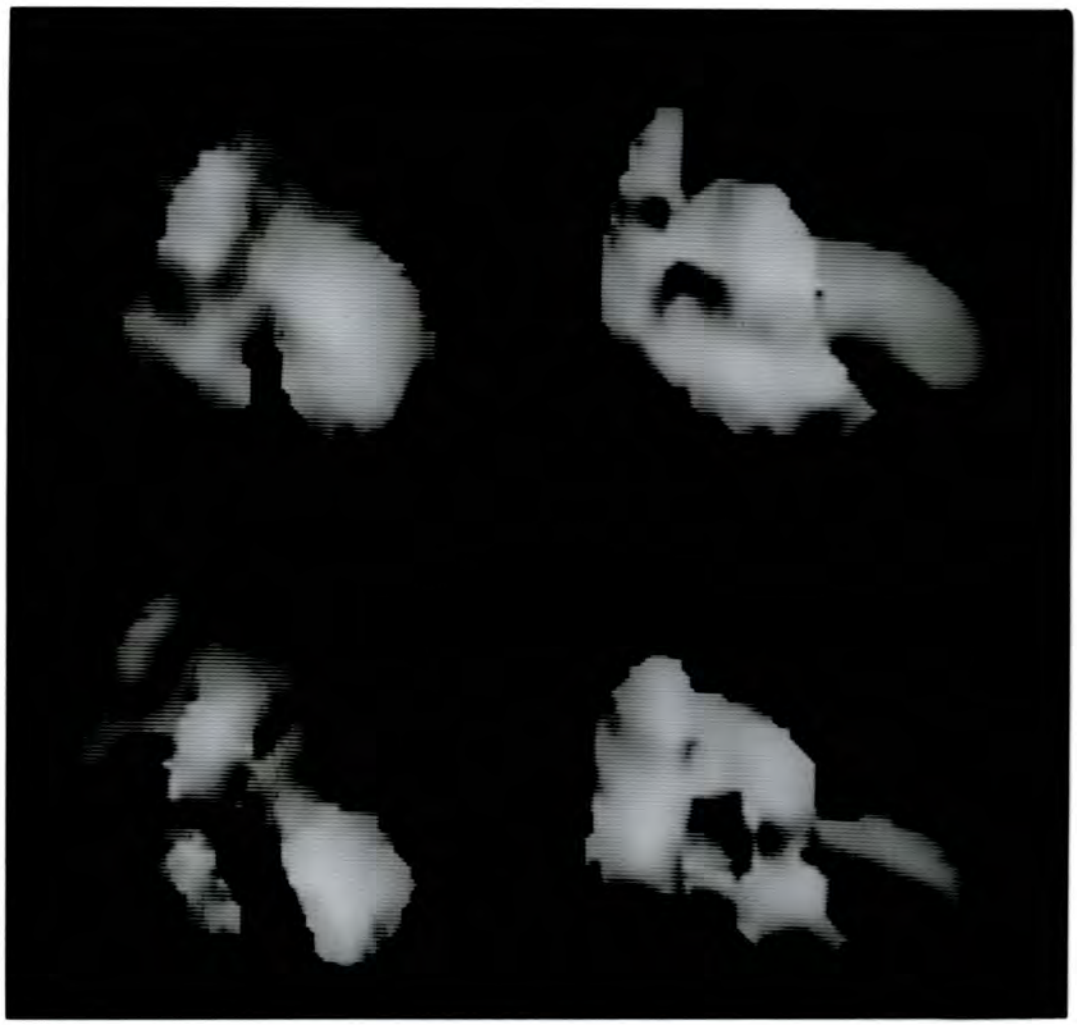


Figure 6.15 Pairs of end-diastolic (ED) and end-systolic (ES) blood pool images :-

Upper - End-diastole at two orientations.
Lower - End-systole at the same orientations.

Figure 6.16 shows end-diastolic (ED) and end-systolic (ES) images in shades of red and green respectively. When the two objects are 'merged' together (see Section 5.5.1) the composite image shows the surface as either red or green, depending on whether the ED or the ES surface is the outermost. Since the ventricles are dilated at ED, and the atria at ES, the normal appearance shows red ventricular surfaces and green atrial surfaces (Figure 6.16). Patients with a ventricular aneurysm, however, may have a bulging portion of ventricular wall at ES, as shown in Figure 6.17. The composite image clearly shows the extent and location of the aneurysm, which affects the apical and lateral portion of the left ventricle. The composite image is therefore a three dimensional analogue of two dimensional wall motion outlines obtained from conventional gamma camera images. It has the advantage that the viewing direction may be varied to display wall motion from any aspect.

6.4 Other Applications

The techniques described in Sections 3 to 5 have also been applied to a wide range of other tomographic data. Some of these applications are described below, illustrated by individual cases.

Polycystic Liver Disease : A woman aged 53 presented with known polycystic liver disease. A radioisotope liver scan was performed to evaluate the extent of disease prior to surgery, and tomographic sections showed a large number of cystic areas in the liver. By selecting only those tissues with concentration of radiocolloid above a threshold value, functioning liver can be separated from the cysts. Figure 6.18 shows the remaining liver tissues. Most of the left lobe has been displaced by a large cyst, with several further cysts in the

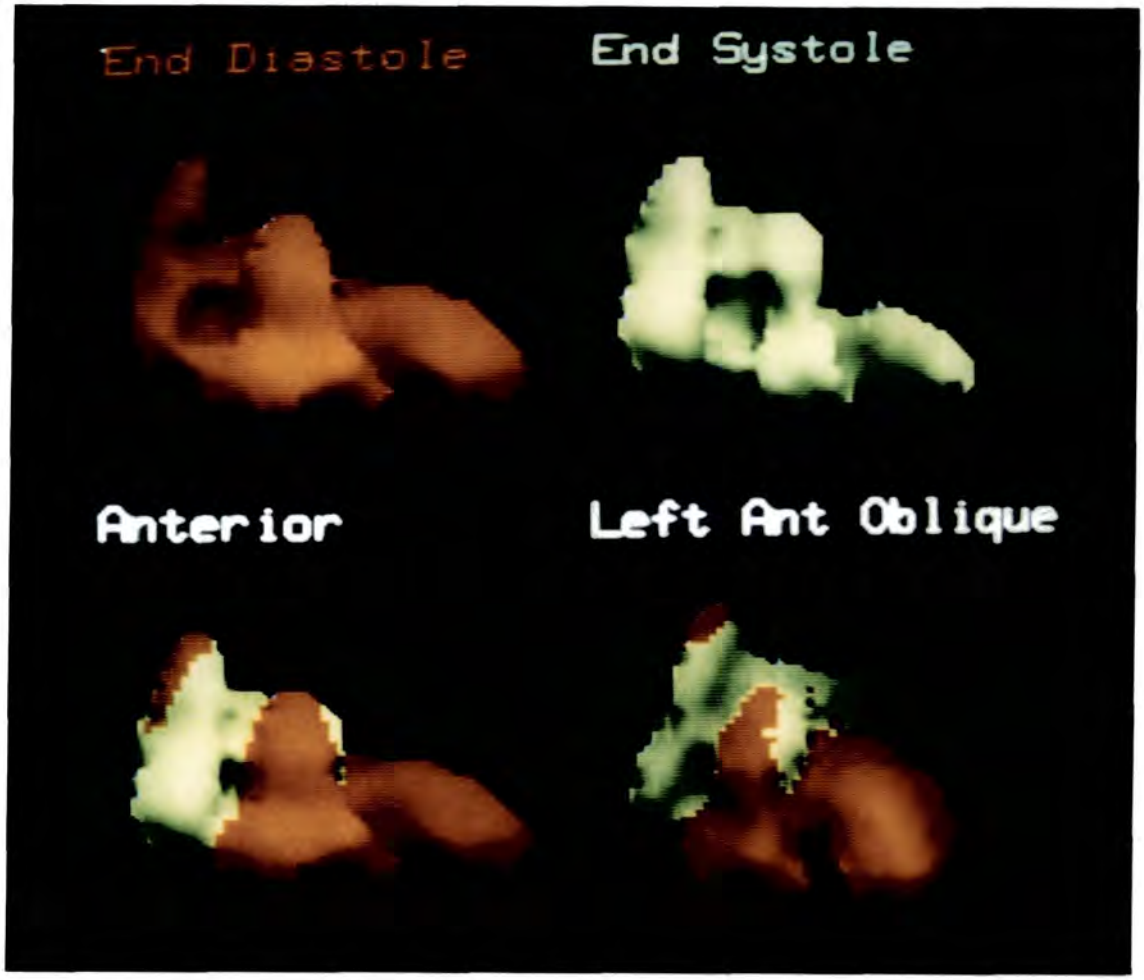


Figure 6.16 Colour coded shaded surface images of a normal heart :-

Upper left - Anterior at end-diastole.
Upper right - Anterior at end-systole.
Lower left - Composite image in the anterior projection.
Lower right - Composite image in the left anterior oblique projection.

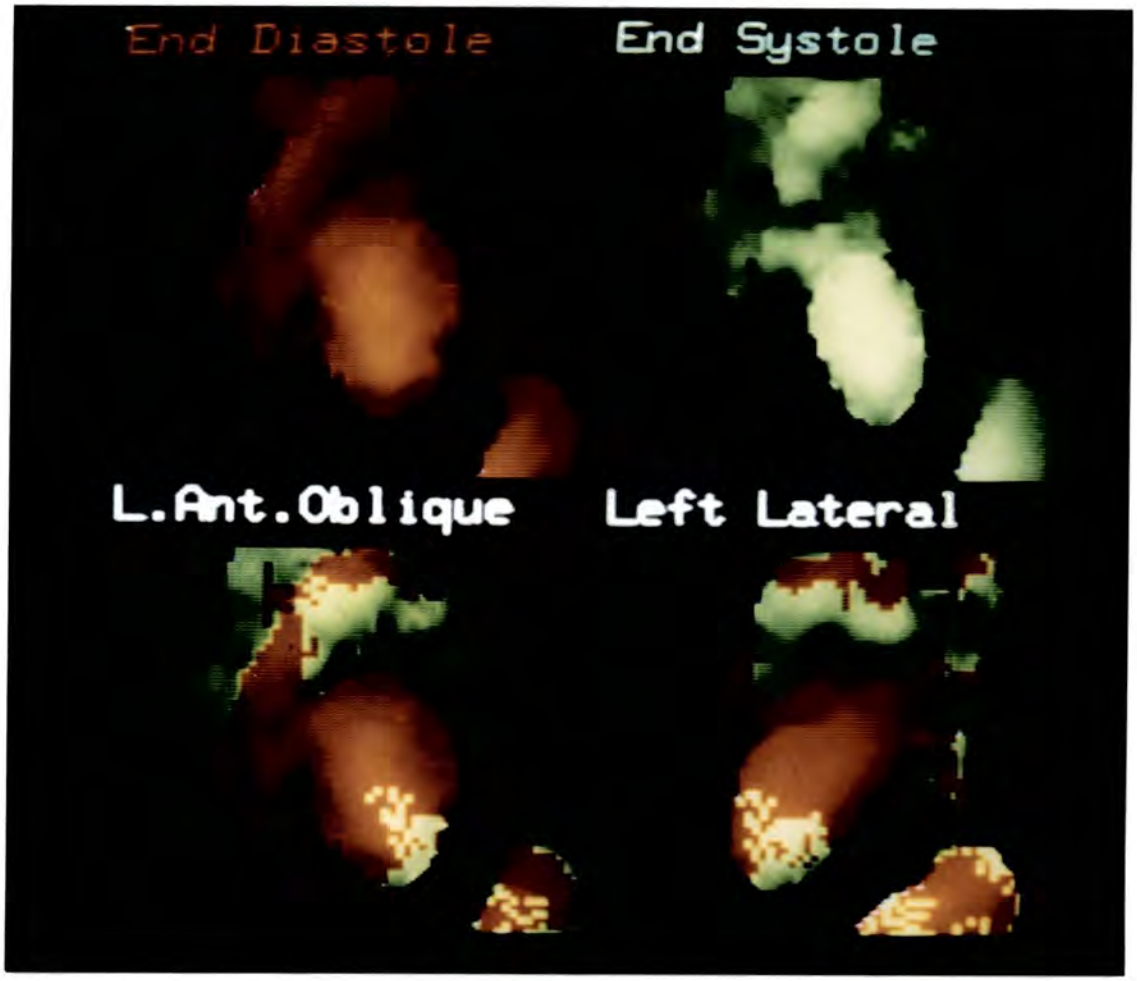


Figure 6.17 Colour coded shaded surface images of a heart with a left ventricular aneurysm :-

Upper left - Left anterior oblique at end-diastole.
Upper right - Left anterior oblique at end-systole.
Lower left - Composite image in the left anterior oblique projection.
Lower right - Composite image in the left lateral projection.

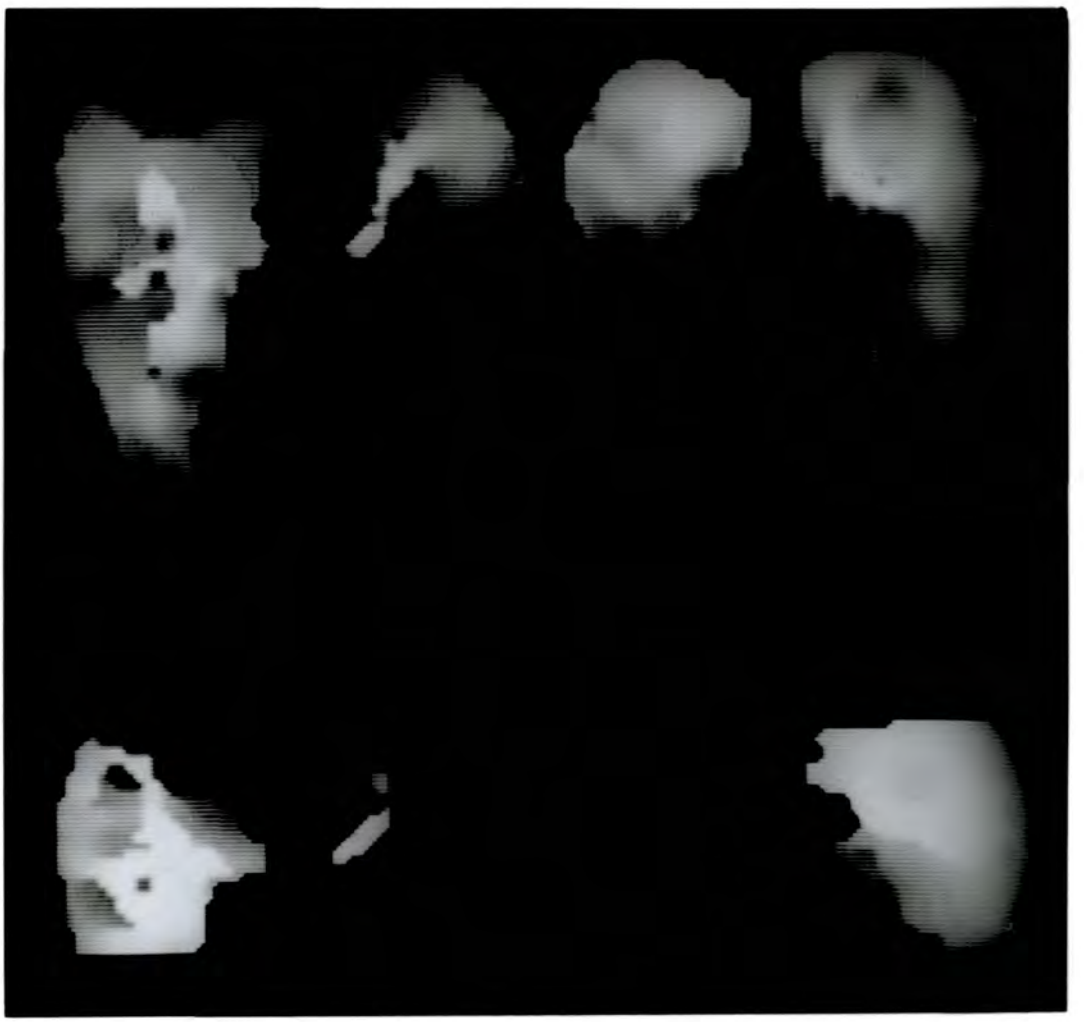


Figure 6.18 Polycystic liver disease imaged using two modalities :-

Upper left - Anterior ECAT study.
Upper right - Posterior ECAT study.
Lower left - Anterior CT study.
Lower right - Posterior CT study.

right lobe. This patient was referred for a CT investigation, and the three dimensional representation of the CT data is also shown in Figure 6.18. The tomographic data obtained by the two modalities are not identical, and the three dimensional representations differ correspondingly. There is a change of scale caused by the use of different sample intervals, the segmented CT data does not show the spleen, and the isolation of liver tissues was a more complex process for the CT data. Interactive outlining was required to separate the liver from surrounding tissues without including the cysts. However, the three dimensional images show reasonably corresponding defects, and either image is of value for pre-operative planning of surgical drainage.

Bony Trauma : Computed tomography is often used to assess possible fractures in complex bony structures. Figure 6.19 shows a normal skull isolated using a simple thresholding algorithm, together with three views of a second patient who has undergone facial trauma. The fracture of the left cheek bone is clearly evident, together with the downward displacement of the bone. The 'holes' which appear in the object (especially at the back of the eye sockets) are a normal feature (Roberts et al 1984). They occur whenever the bone is sufficiently thin for the partial volume effect to occur, and the resulting 'fenestration' shows those voxels where the original pixel value fell below the expected range for bone. Because of this effect, small fractures in thin bones are difficult to display. The use of a more sophisticated segmentation algorithm may obviate this problem. Nevertheless, three dimensional display techniques have the ability to convey morphological information about complex structures and these images show their

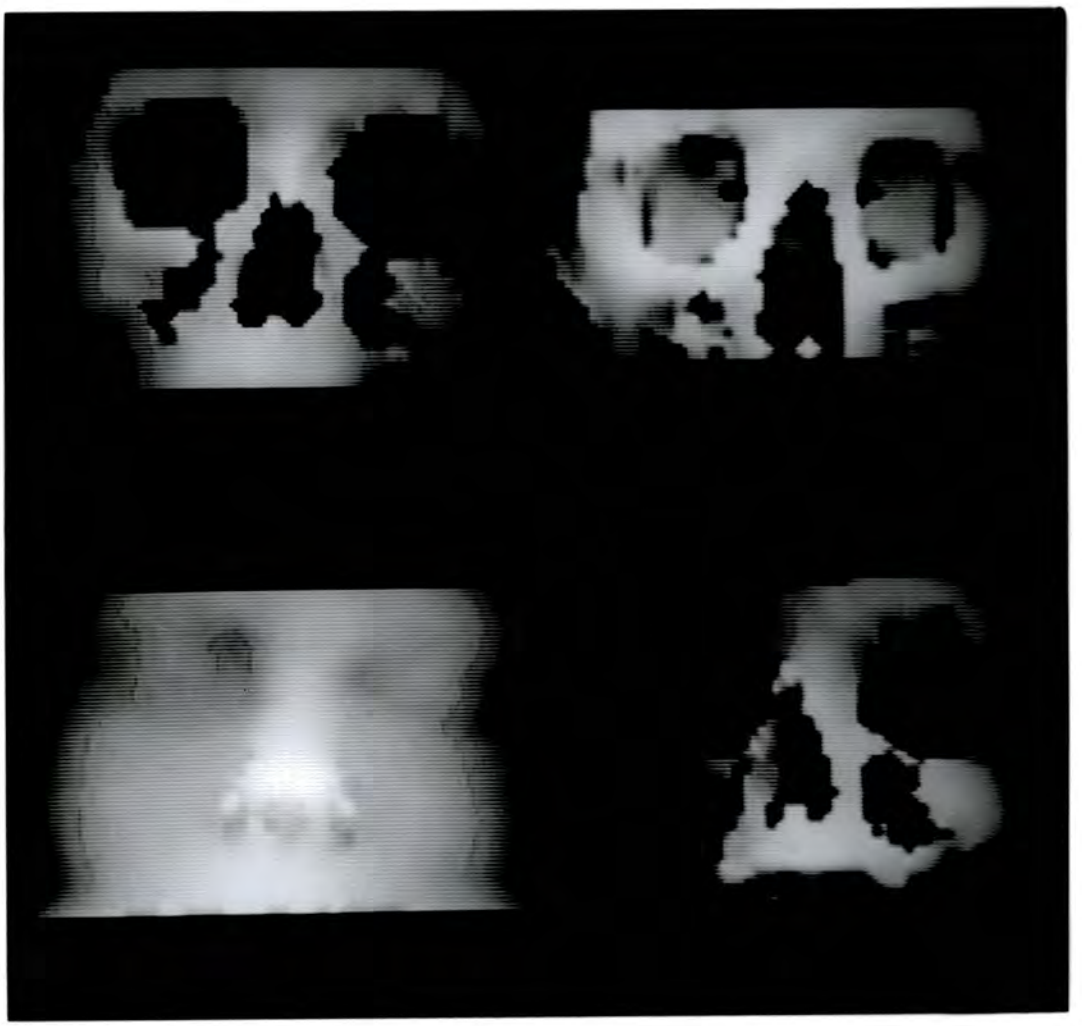


Figure 6.19 Bony trauma demonstrated using shaded surface images :-

Upper left - Skull with fractured cheekbone.
Upper right - Normal skull for comparison.
Lower left - Soft tissue image shows only minimal swelling at the site of the fracture.
Lower right - Oblique view of the fracture and eye socket.

potential value in the interpretation of trauma.

Surgical Planning : Computed tomography is also used to examine the bones of the face and skull prior to oral and maxillo-facial surgery. Figure 6.20 shows bony and soft tissue representations obtained from pre-operative CT scans. To facilitate rapid operator interaction, the data were obtained at only 64 x 64 x 64 resolution, giving the slightly coarse appearance shown. The planned operation would have brought the mandible forward approximately 1.5 cm, with a corresponding steepening of the facial angle. This was simulated by moving forward all the bony voxels within a region selected by the operator. The trackerball was used to mark points on orthogonal views, thus defining planes bounding the volume to be shifted. The resulting bony and soft tissue objects are also shown in Figure 6.20. Although these simulated post-operative objects resemble the expected post-operative result, they are not entirely satisfactory. A finer sampling is certainly necessary to provide the surgeon with better quality images, providing that the interactive capability can be retained.

A frequent problem with elective facial surgery is to convey to the patient the probable results of the planned operation. In particular, patients would like to see realistic images of their post-operative soft tissue outline. To be useful for this purpose the images must be shaded with far more realism than that achieved in Figure 6.20. The use of texture and colours to simulate hair, skin tones and eye colour is a possibility. However, there are also severe problems in predicting soft tissue movement from bone displacements.

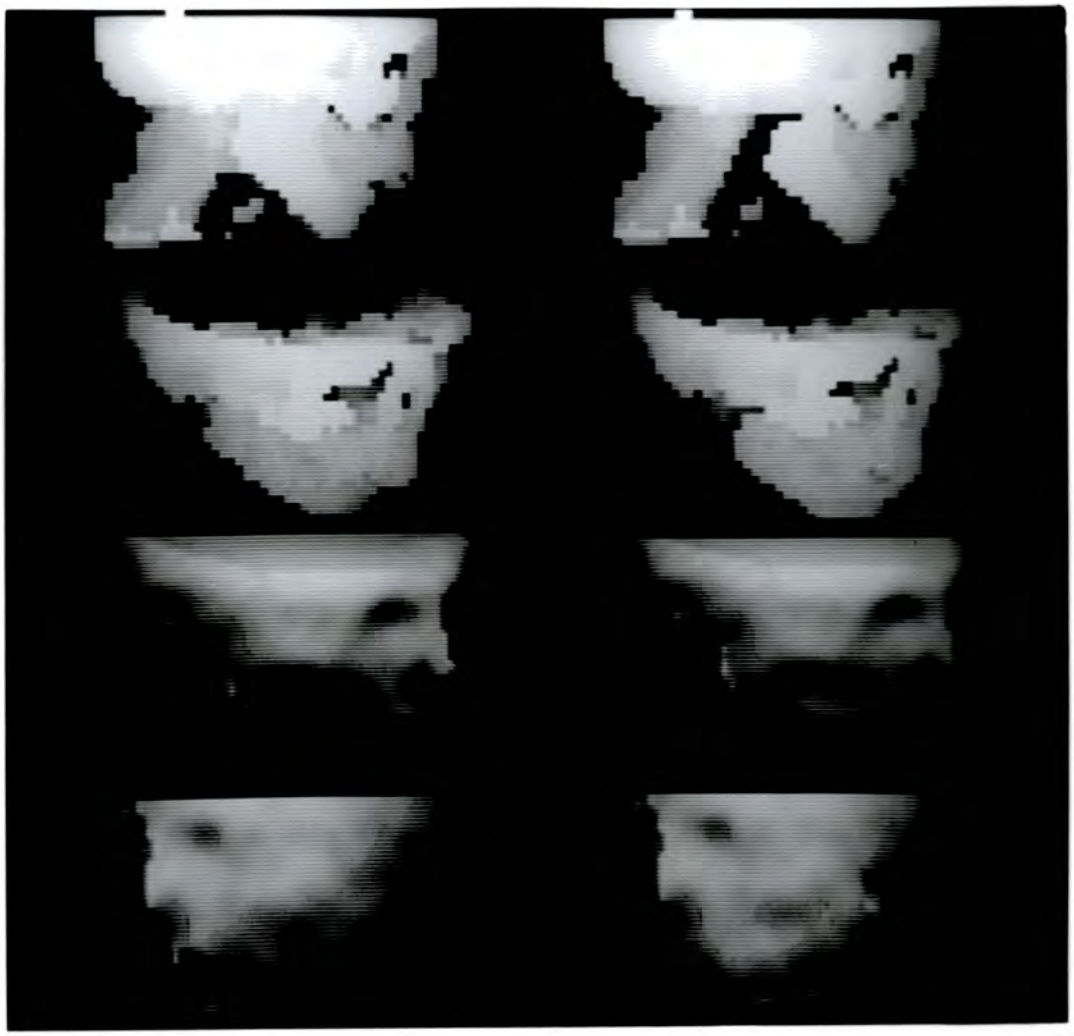


Figure 6.20 Surgical planning using coarse sampling to facilitate interactive object manipulation (left column pre-operative, right column after manipulation) :-

Upper - Bony tissues only.
Lower - Predicted appearance of soft tissues.

Cerebral Perfusion : A variety of radiopharmaceuticals have been developed recently for imaging cerebral perfusion (Blau 1985). In all cases tomographic images are obtained, in which absent uptake indicates areas of under perfused brain tissue. In some cases it may be useful to identify the cerebral artery which supplies such an area. There are three main arteries concerned, the anterior, middle and posterior cerebral arteries. However, unlike the coronary arteries described in Section 6.2, the cerebral arteries lie deep within the cerebral hemispheres, and cannot be visualised using superposition techniques. Instead, a series of schematic outlines were obtained, showing the tissues supplied by the three arteries (Bories et al 1985). These were digitised, and interpolated slices added to obtain three dimensional representations of the areas supplied by each artery. Figure 6.21 shows the composite object coloured blue, red and green to show the distributions of the anterior, middle and posterior cerebral arteries respectively. As well as rotation, to show the surface arterial distribution, the object can be sliced to show the distribution along any plane. These images demonstrate that the techniques developed for the display of actual tomographic sections can also be used with schematic sections, and show the value of colour coding the several parts of a complex object.

Lung Ventilation : The use of radiolabelled aerosols enables tomographic images to be obtained showing the distribution of ventilated lung tissue (Smye and Unsworth 1985). Areas of reduced ventilation appear as areas of diminished or absent uptake. The quality of the tomographic sections is poor, both because of the low initial activity in the lung, and because of the loss of activity through the alveolar wall. Neither of

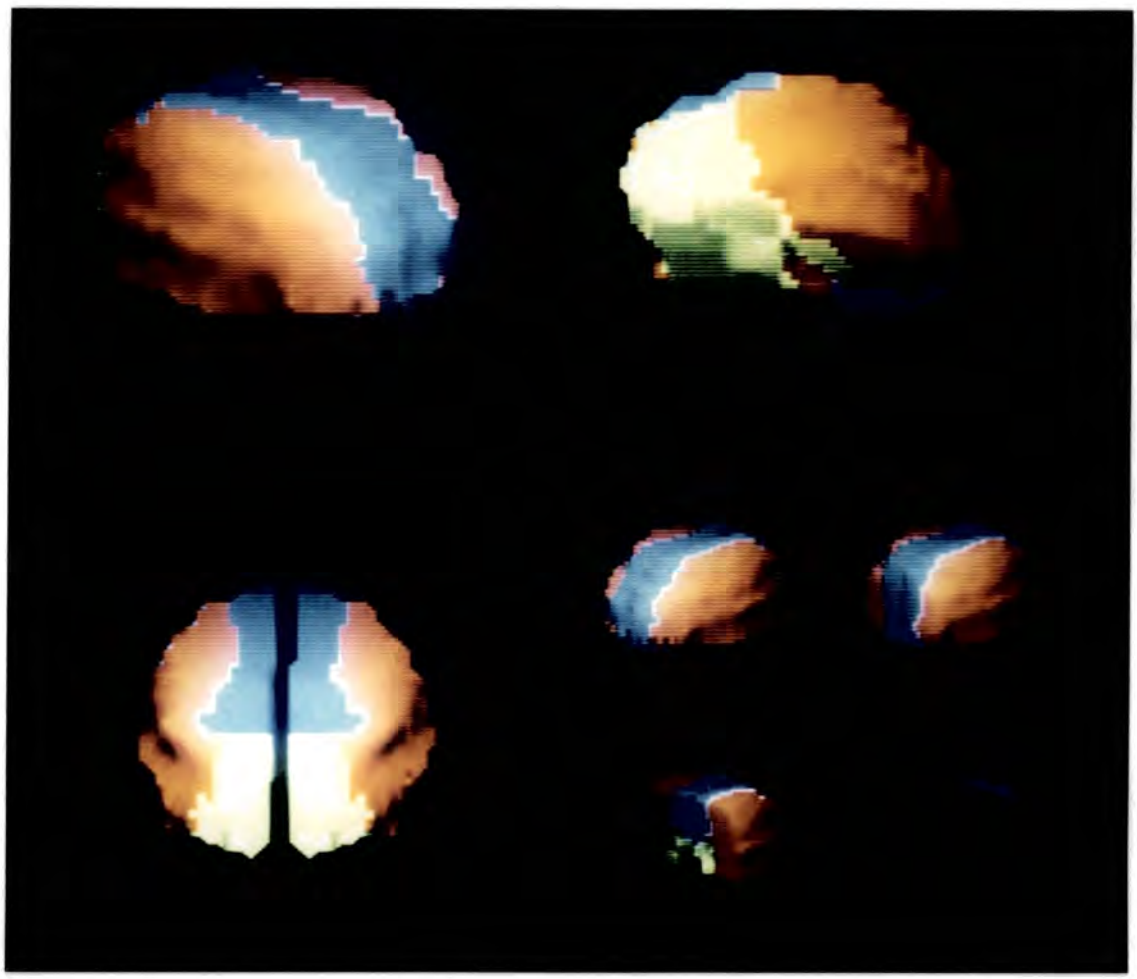


Figure 6.21 Coloured shaded surface images of cerebral perfusion, showing the distribution of the anterior cerebral artery in blue, the middle cerebral artery in red and the posterior cerebral artery in green :-

Upper left - Right anterior oblique, superiorly.
Upper right - Right posterior oblique, inferiorly.
Lower left - Posterior, superiorly.
Lower right - Left anterior oblique, with parts removed to show internal distribution details.

these problems occur if the recently developed ultrafine aerosols are used (McClaren 1987). However, even using conventional nebulised aerosols, three dimensional display techniques can be applied to the tomograms, providing that additional low pass filtering is applied before segmenting the object. Figure 6.22 shows conventional and three dimensional images obtained on two patients. The upper row of four images show conventional gamma camera images of a patient with reduced ventilation to both upper zones. The second row shows corresponding shaded surface images at the same orientations, confirming the location of the unventilated areas. The lower two rows show similar data from a patient with more widespread disease. In this case the location of the ventilation defects is better demonstrated using the shaded surface display. The correspondence between the conventional and the surface displays indicates that the latter can be applied even when the original tomographic data is of poor quality.

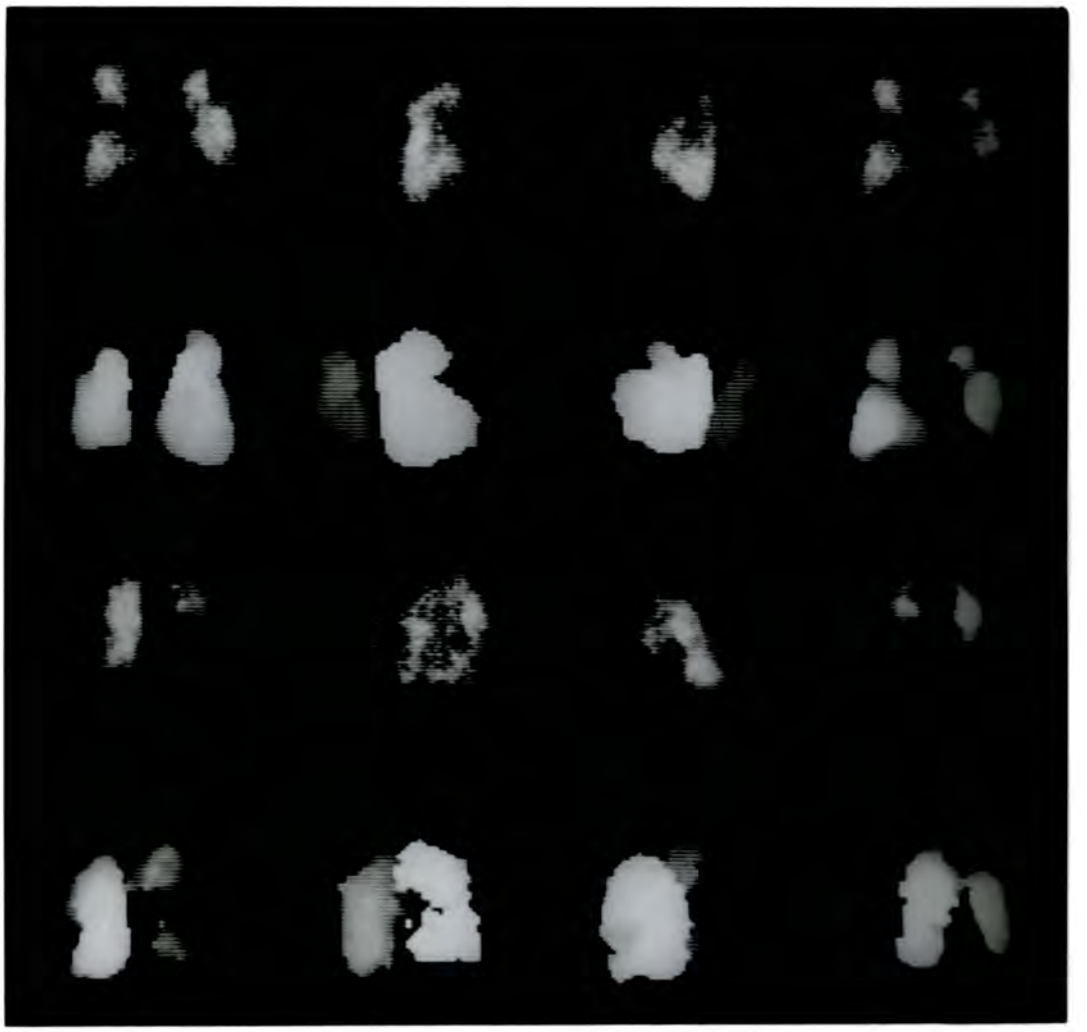


Figure 6.22 Conventional projection images showing Tc-99m aerosol in the lungs, with shaded surface images at the same orientation :-

Upper eight images - A patient with large ventilation defects in the upper zones.

Lower eight images - A patient with widespread smaller defects in both lungs.

7. Conclusions and Future Developments

The main aim of this work was to develop and implement an interactive three dimensional display facility using a conventional medical imaging computer. This aim has been achieved, using a combination of novel and existing techniques, and the system has been successfully applied in several clinical situations.

A fundamental problem of three dimensional display is related to human perception. There are no intuitive or natural processes for perceiving structures in a three dimensional scalar field. This problem was not addressed in this work. Rather, the assumption was made that the display of object surfaces, shaded to simulate natural processes of illumination and reflection, would enable the observer to perceive object morphology. This assumption has been made by many workers, and is justified by their results and those presented in this work.

When observers look at real three dimensional objects they infer shape from several visual cues: stereoscopy, head movement parallax and local surface brightness. However, they are usually able to infer the shape of an object from a photograph of a scene, demonstrating that although stereoscopy and head movement parallax may be useful cues, they are not essential. Shaded surface images by themselves are a powerful method for conveying shape information, and have the advantage that they can be presented on a conventional two dimensional display.

At the outset of the project cuberille schemes were selected for object representation. The almost unlimited topological flexibility of these schemes more than outweighs their inability to represent curved surfaces exactly. They have the additional advantage of being simply and

directly related to the stack of adjacent tomographic sections which forms the input data. Two novel forms of object representation were developed for this work, both being cuberille schemes.

The first representation developed was the Binary Object technique, in which an array of N x N x N bits was used to indicate the location of object voxels throughout the volume of interest. This technique is a solid model representation, rather than just a surface representation, and this enables the observer to explore the interior of the segmented object. The detailed implementation of the technique made use of the fact that each word of memory contained 16 bits, representing 16 voxels, which could be processed simultaneously. This property was exploited to develop a rapid hidden surface algorithm, and a rapid form of operator interaction to remove overlying tissues.

The Binary Object representation scheme enables all objects sampled with a resolution of N x N x N voxels to be represented using $N^3/8$ bytes, irrespective of the actual size or complexity of the object. This constant storage requirement is clearly only efficient for objects which nearly fill the array. In practice, medical objects are not cubic, and large amounts of storage are wasted. Typically only 10 to 30 % of the Binary Object array contains object voxels, with the remainder being background. Neverthless, there are advantages to the programmer in being able to define the size of the object data storage area in advance, and for small values of N the entire array can still be held in main memory. For ECAT data, N is usually only 64, and this sample size can also be sufficient to represent some forms of CT data. Using this sample size the Binary Object array required only 32 Kbytes, and was stored in main memory as a single array of 16384 integer words.

The second novel object representation scheme developed was the Ordered Surface List technique. This is an extremely compact representation, making use of the intrinsic priority ordering of a cubic array to classify object voxels into eight groups according to their visibility from any direction. The technique retains all the topological flexibility of a cuberille scheme, but is only a surface representation. Operator interaction to reveal internal structures requires the Ordered Surface Lists to be recreated from the raw data. However, this process was simple and rapid, enabling the observer to iteratively select a segmentation threshold with a cycle time of less then 10 seconds, for data sampled at 64 x 64 x 64 resolution.

When using this technique the object is represented both as a solid object by the raw data array, and also as a set of surfaces by the Ordered Surface Lists. The creation of the lists can be regarded as the first stage in the display process, in which the observer interactively selects segmentation levels, and performs spatial bounding or manipulation. Once created, the lists are ideally suited to display using a modified form of the back-to-front (BTF) algorithm. Indeed, the Ordered Surface List representation was suggested by the form of this algorithm.

Display algorithms are closely linked to object representation schemes, and vice versa. In all cases transformation matrices are used to display the object as seen from an alternative aspect, and two transformation matrices were developed for this work. The first required six parameters to specify the observer's location and viewing direction, and enabled images to be produced showing any desired aspect. The second matrix required only two parameters to specify the observer's location, from which the viewing direction was deduced, and this transformation

was therefore ideally suited to interaction using a trackerball. Only a limited range of aspects could be produced using this transformation, showing the exterior of the object at a fixed distance from the observer, at any angle. This was found to be entirely adequate for most of the medical images displayed in this work, although it is probable that the general transformation would be of greater value for planning complex surgical operations, where interior views may be required.

In common with other workers, look-up tables were used for rapid evaluation of trigonometrical functions. Integer representation of both object space and image space coordinates was used, with sub-voxel precision achieved by performing all calculations with scaled integers. A scaling factor of 256 was used, so that the most significant byte of each 16 bit word corresponded to the coordinate value, with the least significant byte available for any fractional component. No significant loss of precision was observed using this technique, and it enabled rapid Assembly language coding of the otherwise time consuming procedure of coordinate transformation.

The inherent priority ordering of a cuberille array leads naturally to special purpose hidden surface algorithms. As described in Section 2, these can be based either in 'image space', using rays to interrogate the object, or in 'object space', projecting each voxel onto a display screen. Two special purpose hidden surface algorithms were devised for this work, corresponding to the two object representation schemes. The first algorithm was used for Direct Display of the Binary Object representation, and is an example of an image space technique. It uses interrogating rays which are aligned parallel to one of the principal axes of the object array, and hence executes rapidly with no risk of missing any visible portions of the object.

The execution time for a Binary Object array of N^3 voxels was found to be essentially constant, irrespective of the object size. This would not be the case for a ray-tracing algorithm using an oblique angle, and is a feature of the correspondence between the format of the data storage and the viewing direction. It is difficult to extend the algorithm to larger values of N unless significantly more memory is used. The Binary Object array requires storage to increase $O(N^3)$, and an intermediate array of order $O(N^2)$ is also required for both object rotation and visible voxel determination. For this reason the Direct Display technique was applied to objects sampled at 64 x 64 x 64 resolution only.

The second display algorithm devised was a modified form of the back-to-front (BTF) algorithm, and was applied to the Ordered Surface List representation. It is an example of an object space technique, and was the most rapid display algorithm assessed in this work. When applied to Ordered Surface Lists obtained with N equal to 64, the display rate was sufficiently fast to give the appearance of real time rotation, with consequent enhancement of depth perception by the motion parallax effect. Display rates of 2.8 to 8 Hz were achieved on the faster of the two computer systems used in this work. Since this is a surface representation, the storage space required for each list will increase $O(N^2)$. The time taken to display an object was shown to be proportional to the list length, and so this will also increase $O(N^2)$. Display rates for objects sampled using N equal to 128 should therefore be approximately 0.5 to 2 Hz, using the same computer system, and preliminary results suggest that this is the case. Although this is hardly real time rotation, it is certainly fast enough for interactive display.

It must be pointed out, however, that creation of the Ordered Surface Lists is an $O(N^3)$ procedure, since all the object voxels must be processed. For the original BTF algorithm the whole display process is similarly an $O(N^3)$ procedure, and this demonstrates a second advantage of the modified algorithm. Not only can real time display be achieved using a surface representation, but the relative speed advantage of the modified BTF algorithm increases in proportion to the sample size N.

For rapid display and interactive procedures the visible voxel image was shaded according to distance from the observer to produce 'depth shading'. Surface shading gives a more realistic appearance and has been used by most workers for hard copy images. The technique of Image Space shading was developed for this work, enabling the surface orientation to be estimated from an intermediate 'depth image', containing the distance from the observer to the visible portions of the object. This obviates the necessity to precompute and store surface orientation information with the voxel coordinates. The validity of this approach has been confirmed not only by the results presented here, but also by independent development of a similar technique.

The choice of illumination direction affects the appearance of the object, and the most realistic results were obtained using a light vector slanting obliquely down from above and behind the observer. This was preferred to the simpler technique of illumination along the viewing direction, and a shading scheme was devised which enabled naturalistic 'overhead' illumination without discontinuous changes in the surface brightness. For image space shading the surface normal is estimated from the gradient along two orthogonal axes of the depth image, and the performance of three gradient estimators was assessed using a one dimensional semi-elliptical phantom. A local polynomial fitting

procedure was developed for this work, and this was found to be more accurate than either of two techniques which used averages of forward and backward differences.

The polynomial fit required more computation than the other two procedures, but this was not a significant disadvantage. Computation of the gradient using a polynomial fitted to equally spaced points in the depth array only required a 5 or 3 point convolution, and the whole process of shading a depth array is an $O(N^2)$ procedure, applied relatively infrequently for the production of detailed or hard copy images. The additional time required for gradient evaluation was compensated for by improved image quality, especially for the noisy data from ECAT studies.

Shading is the most subjective component of any three dimensional display procedure, and the techniques developed were accordingly assessed by a panel of judges. Eight observers were asked to compare four shading algorithms ranging from pure depth shading (S_d) to pure surface shading (S_c), with two intermediate forms containing differing amounts of both depth and surface information (S_{mc} and S_p). Two of the techniques were developed for this work, and the other two were developed by other workers. The techniques containing the most surface information were clearly preferred to those containing mostly depth information, and of those two the smoother polynomial shading S_p was just preferred to the pure cosine shading S_c, although levels of significance were not always high for this comparison. The judges were in excellent agreement, with seven observers rating the polynomial algorithm first overall, and one observer rating it second to the cosine algorithm.

Colour was also used in an attempt to convey functional as well as morphological information. This was found to be a difficult task, since the observer was required to distinguish between bright hues representing genuine organ function, and those which were due to overlap between organs. More succesfully, colour was also used to identify sections of complex objects, taking hue as an arbitrary surface property. This was applied to both single complex objects composed of several sections, and to interpenetrating multiple objects. In particular, the use of contrasting red and green hues to show simultaneous end diastolic and end systolic surfaces produced a three dimensional image of cardiac wall motion, unique to this form of display.

Colour was also used as an adjunct to transparency, to display internal structures. A shading technique was devised for simulating glassy surfaces which was based on pure specular reflection of diffuse illumination, rather than pseudo-specular reflection of directional illumination. This enabled even irregular organs to be displayed as glassy surfaces. Although colour was an important factor in the final implementation of this algorithm, adequate results were also obtained using a monochrome display.

Stereo display was not successful using the 256 x 256 screen memory available. The extraordinary stereoacuity of the eye means that even single pixel discrepancies between the left and right eye images were perceived as noticeable depth differences, giving a depth quantisation effect. This could be reduced by using a display screen with a much higher horizontal resolution, in which case stereo display techniques may be able to provide an additional depth cue.

The future development of three dimensional display techniques will be strongly influenced by developments in computer hardware. Conventional computing systems will continue to increase in power, with a probable decrease in real costs. Additional memory will have an impact on the choice of object representation schemes. Very large Binary Object arrays will be feasible, enabling objects sampled at 256^3 resolution to be represented in a single 2.1 Mbyte array. With even larger amounts of memory, the entire set of input slices, consisting of 256^3 voxels, could be stored in 16.8 Mbytes, providing that each voxel could be represented by a single byte.

These large object arrays will provide accurate models for even highly complex shapes. However, to maintain the same speed of display the processing power of the computer must increase substantially. The time taken to process a 256^3 Binary Object array will be approximately 64 times longer than for a 64^3 array, although for corresponding Ordered Surface List representations the increase in processing time is only a factor 16. Even though processing speeds will increase, it is likely that the increase in memory will exceed the speed-up factor, at least for laboratory or desk top computer systems. Hence the Ordered Surface List scheme will be even more attractive for larger object arrays.

A more exciting prospect for rapid display algorithms is parallel processing. The algorithms developed for this work exhibit at least some degree of parallelism in their structure, which could be exploited by even a modest array of processors. Assume that an array of 64 identical processing units was available, each with local memory and with access to a communal memory. The Binary Object representation could be created in the main memory and then distributed to the local memory of individual processors as a set of independent slabs. A 256^3 array could

be arranged as 64 horizontal slabs, each containing 256 x 256 x 4 voxels. Each processor could simultaneously determine the visible voxels in its slab, including any rotation about an axis perpendicular to the slab. This information could be returned to the main memory as a set of 64 image strips, each 256 pixels wide by 4 pixels high. The expected speed-up factor would be 64, since the processors can operate independently.

Subsequent shading of the 256^2 visible voxel array can also be performed as a largely parallel process. The image could be divided into 64 sub-images each 64 pixels wide by 64 pixels high, which can be shaded simultaneously. However, in order to operate on this data without problems of data access contention amongst the processors, it may be necessary to transfer the sub-images to the local memories. If this is too time consuming then shading can be partially completed using the 256 by 4 pixel depth array strips associated with each processor. An additional step will then be required to ensure continuity of shading between adjacent strips. Rotation of the object about an axis parallel to the slabs is more difficult, and will probably require the Binary Object to be re-sliced into vertical slabs. The precise implementation of this procedure will depend on the relative speeds of data transfer to local memories and of multiple processor access to the communal memory. It seems likely that the average speed-up factor for the Binary Object scheme will be less than 64, although still quite large.

Display of Ordered Surface Lists using the modified BTF algorithm appears to be an entirely sequential procedure, but there are several stages at which parallel processing may be implemented. A raw data array of 256^3 voxels could be split into 8 horizontal slabs, each 32 voxels thick. Copies of the lowest slab would be issued to the first 8

processors, copies of the next slab to the second 8 processors, etc. Each slab can then be processed simultaneously by 8 processors to produce all 8 lists of potentially visible voxels. This reduces a laborious $O(8 \times 256^3)$ procedure to an $O(32 \times 256^2)$ procedure, a speed-up factor of 64. However, the lists must then be concatenated in correct order, and many voxels discarded which were only considered visible because of the separation of the object into slabs. The overall speed-up factor for creation of Ordered Surface Lists will be less than 64, but not greatly so.

Subsequent display of the lists can also be performed using at least some parallel processing. A list can be split into 64 approximately equal segments and issued to the processors, which can independently calculate the transformed voxel coordinates for their segment according to the current viewing position. However, a correct hidden surface image will only be produced if the transformed voxels are added to the output array in strict sequential order, so the overall speed-up factor will again be less than 64.

The conclusion to be drawn from this brief analysis of future hardware developments is that processing power must increase substantially to make use of more detailed object representations. Even a modest increase of a factor four in the linear sampling density produces a factor 64 increase in object volume. For such objects it is likely that parallel processing will be required to significantly surpass the display rates achieved in this work using coarsely sampled objects.

As well as the further development of display algorithms there will also be an increasing number of applications. As with all new ideas, there has been an initial period of development until the basic

strengths and limitations have been understood. Three dimensional display methods have now reached the stage where a controlled clinical trial would be justified, to evaluate the benefits to be gained against the costs incurred.

The emphasis throughout this work was placed on the development and assessment of techniques, rather than on evaluation of their applications. Neverthless, a substantial number of clinical applications were studied, including representative examples from three imaging modalities. In particular, this work was amongst the first to demonstrate three dimensional display using ECAT data, and marked success was obtained using gated blood pool tomograms. Shaded surface display techniques are ideally suited to this situation, since the movement of the cardiac wall is shown most clearly in three dimensions. Successful images of myocardial perfusion were also obtained with Thallium-201, a radionuclide with less than ideal properties for imaging. It is therefore likely that the same technique will be of value for displaying the improved gated myocardial uptake images which can be obtained with the new class of Technetium-99m labelled isonitrile imaging agents.

The display of bony CT data is now a well established tool, both for evaluation of complex anatomy and for pre-operative planning. With further computing power we may expect interactive manipulation to become both quicker and more realistic. It may be possible to allow the surgeon to manipulate the object in real time, allowing several plans to be evaluated. At this stage, tactile as well as visual feedback could be introduced, so that the surgeon feels whether bone or soft tissue is being cut by the pressure needed to move the interactive device. With more realistic shading of detailed object models, the patient may be

able to inspect a realistic simulation of their post-operative appearance, a particularly desirable feature for elective procedures.

It is impossible to predict the other areas in which three dimensional display will definitely find a clinical role. Preliminary results in this work have shown that NMR data can be displayed using shaded surfaces, and other workers have shown similar results using ultrasound images. Using the experience gained in analysis of CT and ECAT data, it is likely that three dimensional display methods will prove to be useful, and that they will be applied most frequently to complex anatomical structures, where shape determination from adjacent cross-sections is a difficult task.

Appendix 1 Derivation of the Viewing Transformations

General Transformation

Let the observer be located at the point (xyz), and let the viewing direction have unnormalised direction cosines (l,m,n). If the x' axis of the observer's coordinate system is constrained to lie parallel to the xz plane then a general viewing transformation may be calculated (Figure 4.1). Stages in this calculation are as follows:-

a) Translate the coordinate system to the observer's location,

$$T_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -x & -y & -z & 1 \end{pmatrix}$$

b) Rotate about the new y-axis until the new z-axis lies in the vertical plane through the viewing direction. This corresponds to clockwise rotation through an angle equal to $\tan^{-1}(l/n)$,

$$T_2 = \begin{pmatrix} n/u & 0 & l/u & 0 \\ 0 & 1 & 0 & 0 \\ -l/u & 0 & n/u & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where $u^2 = l^2 + n^2$.

c) Rotate about the new x-axis until the new z-axis lies parallel to the viewing direction. This corresponds to anti-clockwise rotation through an angle equal to $\tan^{-1}(m/u)$,

$$T_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & u/v & m/v & 0 \\ 0 & -m/v & u/v & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where $v^2 = l^2 + m^2 + n^2$.

d) The overall matrix is obtained by combining T_1, T_2 and T_3,

$$T = (T_1 . T_2) . T_3$$

hence
$$T = \begin{pmatrix} \frac{n}{u} & \frac{-lm}{uv} & \frac{l}{v} & 0 \\ 0 & \frac{u}{v} & \frac{m}{v} & 0 \\ \frac{-l}{u} & \frac{-mn}{uv} & \frac{n}{v} & 0 \\ \frac{1}{u}(lz-nx) & \frac{m}{uv}(lx+nz)-\frac{yu}{v} & -\frac{1}{v}(lx+my+nz) & 1 \end{pmatrix}$$

Restricted Transformation

A restricted viewing transformation with only two degrees of freedom was used for interactive display. Let the coordinates of the point at the centre of the object be (C,C,C). This point has the same coordinates in the observer system, for all orientations. The observer location is specified by the parameters θ and ϕ, and the observer x' axis is constrained as before, to lie parallel to the object xz plane (Figure 4.2). Stages in the calculation are as follows:-

a) Translate the coordinate system to the fixed point at the centre of the object,

$$T_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -C & -C & -C & 1 \end{pmatrix}$$

b) Rotate about the new y-axis, anticlockwise through angle θ,

$$T_2 = \begin{pmatrix} c_t & 0 & -s_t & 0 \\ 0 & 1 & 0 & 0 \\ s_t & 0 & c_t & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where $c_t = \cos \theta$ and $s_t = \sin \theta$.

c) Rotate about the new x-axis, anticlockwise through angle $\emptyset$,

$$T_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & c_p & s_p & 0 \\ 0 & -s_p & c_p & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where $c_p = \cos \emptyset$ and $s_p = \sin \emptyset$.

d) Translate the new coordinate system so that the fixed point lies at the point (C,C,C),

$$T_4 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ C & C & C & 1 \end{pmatrix}$$

e) As before the overall transformation matrix can be obtained by combining these elementary matrices,

$$T = ((T_1 . T_2) . T_3) . T_4$$

and hence,

$$T = \begin{pmatrix} c_t & s_t s_p & -s_t c_p & 0 \\ 0 & c_p & s_p & 0 \\ s_t & -c_t s_p & c_t c_p & 0 \\ C(1-s_t-c_t) & C(1-c_p-s_t s_p+c_t s_p) & C(1-s_p+s_t c_p-c_t c_p) & 1 \end{pmatrix}$$

Appendix 2 Fitting Coronary Arteries to a Myocardial Surface

Points **n** and **a** specify the location of the arterial tree, **N** and **A** the location of the myocardium. It will be assumed that the arteries can be fitted to the myocardium using a transformation which renders **n** and **a** coincident with **N** and **A**. In the general case, matching two points is insufficient to determine this transformation uniquely. However, if we assume that the myocardium and the heart are already aligned along the vertical axis then a satisfactory transformation may be deduced. This is a reasonable assumption, since transverse tomographic Thallium-201 sections are always obtained perpendicular to the long axis of the body, and the arterial tree was similarly aligned before digitisation.

Let $\mathbf{N} = (X_n, Y_n, Z_n, 1)$ $\quad \mathbf{A} = (X_a, Y_a, Z_a, 1)$

$\mathbf{n} = (x_n, y_n, z_n, 1)$ $\quad \mathbf{a} = (x_a, y_a, z_a, 1)$

Stages in the transformation are (Figure 6.8):-

a) Rotate about the z axis until the projection of the line **n-a** on the xy plane is parallel to that of the line **N-A**.

$$\text{Let} \quad A_1 = \tan^{-1}[(Y_a - Y_n)/(X_a - X_n)]$$

$$A_2 = \tan^{-1}[(y_a - y_n)/(x_a - x_n)]$$

$$\text{and} \quad B_1 = A_2 - A_1$$

$$\text{then} \quad \mathbf{T_1} = \begin{pmatrix} \cos B_1 & -\sin B_1 & 0 & 0 \\ \sin B_1 & \cos B_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Note that this transformation moves the object, rather than the coordinate system, and is therefore the inverse of those described in Section 4.

b) Define a new coordinate system with x axis parallel to **n-a**, z axis unchanged. The new coordinates are obtained using the transformation:-

$$T_2 = \begin{pmatrix} \cos A_1 & -\sin A_1 & 0 & 0 \\ \sin A_1 & \cos A_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

c) Using the new coordinates, rotate about the y axis until the line **n-a** is parallel to the line **N-A**. Let L_{NA} be the distance from **N** to **A**, and l_{na} be the distance from **n** to **a**.

Let $A_3 = \sin^{-1}[(Z_n - Z_a)/L_{NA}]$

$A_4 = \sin^{-1}[(z_n - z_a)/l_{na}]$

and $B_2 = A_4 - A_3$

then $T_3 = \begin{pmatrix} \cos B_2 & 0 & \sin B_2 & 0 \\ 0 & 1 & 0 & 0 \\ -\sin B_2 & 0 & \cos B_2 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

d) Return to the original coordinate system using the inverse of T_2:-

$$T_4 = \begin{pmatrix} \cos A_1 & \sin A_1 & 0 & 0 \\ -\sin A_1 & \cos A_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

e) Scale the arteries isometrically until $l_{na} = L_{NA}$:-

$$T_5 = \begin{pmatrix} L_{NA}/l_{na} & 0 & 0 & 0 \\ 0 & L_{NA}/l_{na} & 0 & 0 \\ 0 & 0 & L_{NA}/l_{na} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Transformations T_1 to T_5 can be concatenated to obtain a single matrix:-

$$T_6 = (((T_1 . T_2) . T_3) . T_4) . T_5$$

This is applied to the point **n** to obtain an intermediate point m,

$$\mathbf{m} = \mathbf{n} \; . \; T_6$$

where $\mathbf{m} = (x_m, y_m, z_m, 1)$

f) The final transformation matrix applies a simple translation to make **m** coincide with **N**. The points **a** and **A** will also be coincident after this transformation:-

$$T_7 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ X_n - x_m & Y_n - y_m & Z_n - z_m & 1 \end{pmatrix}$$

In the implementation of this algorithm, a subroutine was written to calculate the overall transformation matrix numerically given points **N**, **A**, **n** and **a**.

Appendix 3 Software and Selected Listings

The basic structure of a three dimensional display program for a single object is shown below:-

Algorithm A3.1

get contiguous tomographic sections
segment into object and background
create 3-D representation
save on disk
choose viewing position ←
transform into observer coordinates
create a depth (visible voxel) array
create a shaded image and display → (back to choose viewing position)

The 3-D representation is created using interactive selection of the segmentation parameters. Once stored on disk it can be used for further display without the need for the original sections to be present. This is particularly useful for the display of objects sampled at several time intervals:-

Algorithm A3.2

get R 3-D representations of the object
k = 1
choose initial viewing position
transform the k^{th} representation into observer coordinates ←
create a depth array
create a shaded image and display
new viewing position? → **yes** → (back to transform)
k = k + 1
if k > R then k = 1 → (back to transform)

In this case the observer chooses a new viewing direction by adjusting the trackerball position. If a new position is not selected, then the display cycles through the R object representations, giving a 'movie' effect.

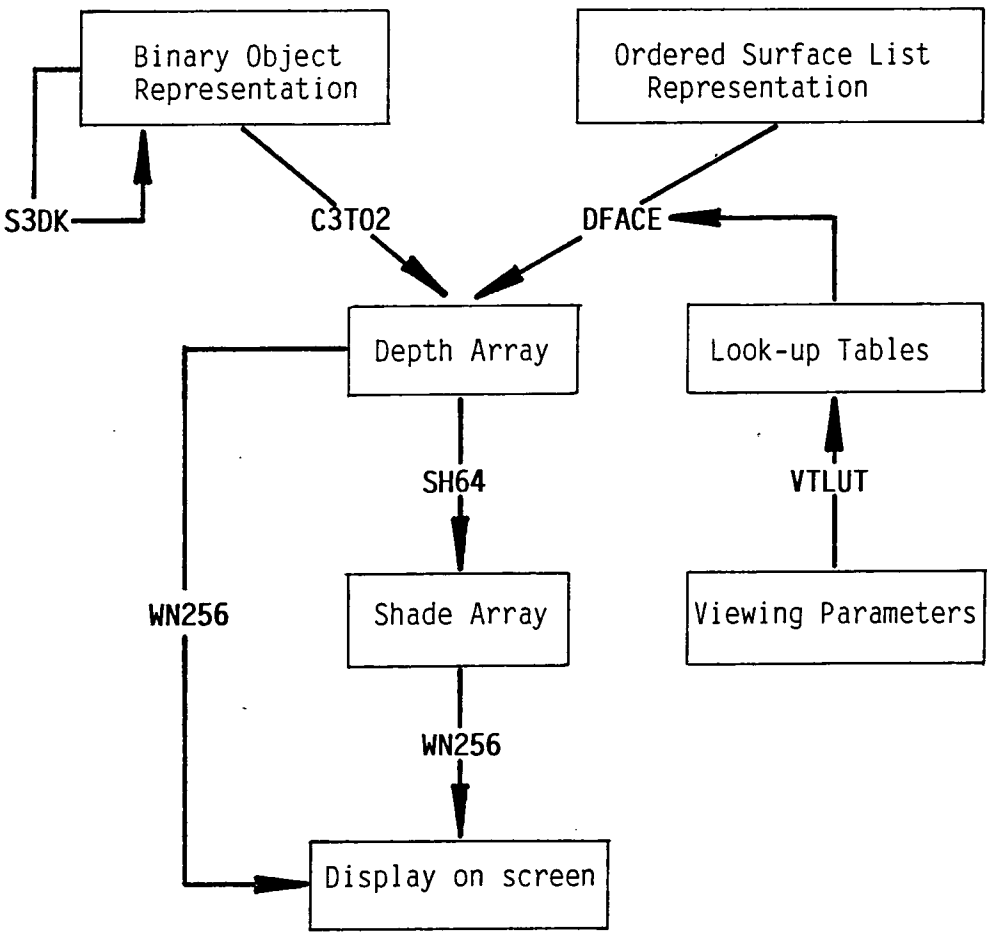
For the simultaneous display of several independent objects a slightly different algorithm is required:-

Algorithm A3.3

get R 3-D object representations
choose hues to assign to each object
choose viewing position ←
for k = 1 to R
  transform the k^{th} representation into observer coordinates
  create a depth array
  merge with existing depth array, resolving priorities
next k
create a shaded surface image using composite depth array
colour the surface using appropriate hue values for visible voxels
display final image → (back to "choose viewing position")

The merged depth array contains the nearer of the existing and the new depth values for each pixel. During the merging process, information must be retained to indicate the object which is visible at each pixel. This can be accomplished using a second array, equal in size to the depth array, containing the k values for the voxels which are still visible after each merger. When all the objects have been processed, this array contains sufficient information to enable the surface image to be coloured with the hue assigned to each object.

Six listings have been included, illustrating stages in the display procedure from calculation of a viewing transformation to display of a shaded surface image. The flow of data and the relationships between these subroutines are summarised below :-



The routines are suitable for execution on either of the systems used in this work. They are written in FORTRAN and ASSEMBLER for the Data General Corporation NOVA 4x.

a) The Binary Object hidden surface algorithm

Format : C3TO2(IBIN,IDEPTH)

Input Data : A binary object array declared as IBIN(0:63,0:63,0:3), in which the first two coordinates correspond to the x and y axes of Figure 4.1, and the third coordinate together with the bit number (0..15) corresponds to the z axis. Each element of the array contains 16 bits and therefore represents 16 object voxels. For example, the voxel with coordinates (20,20,0) is represented by to the least significant bit of IBIN(20,20,0), and the voxel at (30,40,63) by the most significant bit of IBIN(30,40,3).

Action : The routine interrogates the binary object array to obtain the depth (z coordinate) value for each pixel in the output array. Pixels for which no object is visible are assigned the value 4096.

Output Data : A depth array declared as IDEPTH(0:63,0:63), in which the pixel values are the depth coordinates of the visible voxels at the same (x,y) locations, scaled by a factor 16 to facilitate subsequent manipulation.

```
;       CGC3TO2.SR
;
;       CALL C3TO2(IBIN,IDEPTH)
;
;       CONVERTS THE 3-D BINARY IMAGE IN IBIN(0:63,0:63,0:3) TO A
;       2-D PROJECTION IN IDEPTH(0:63,0:63)
;
;       THE VALUE OF IDEPTH(I,J) IS THE SCALED K COORDINATE (0..1023)
;       OF THE FIRST SET VOXEL ALONG THAT COLUMN. IF NO VOXELS ARE SET
;       THEN IDEPTH(I,J) IS SET TO 4096.
;
;       CJG 24/8/83
;
.TITLE  C3TO2
.EXTD   .FARL,.FRET
.ENT    C3TO2
.NREL
;
        4
C3TO2:  JSR 3.FARL
;
;       SET UP ADDRESSES AND OUTER LOOP COUNTER
;
        LDA 0,-167,3    ;ADDRESS OF IBIN(0,0,0)
        STA 0,ADRA
        LDA 0,-166,3    ;   "    "  IDEPTH(0,0)
        STA 0,ADRB
        LDA 0,C4K       ;4096
        STA 0,CNT1
;
;       OUTER LOOP
;
LOOP1:  LDA 2,ADRA      ;ADDRESS OF IBIN(I,J,0)
        LDA 0,C4
        STA 0,CNT2      ;SET MIDDLE LOOP COUNTER
        LDA 1,C4K
```

```
;
;       MIDDLE LOOP
;
LOOP2:  LDA 0,0,2       ;GET IBIN(I,J,K)
        MOV# 0,0,SZR    ;SKIP IF NO BIT SET
        JMP BIT         ;BIT SET, FIND WHICH
        ADD 1,2         ;INCREMENT K
        DSZ CNT2        ;DONE 4 TIMES ?
        JMP LOOP2       ;NO
;
;       ALL WORDS AT THIS (I,J) ARE 0, SET IDEPTH(I,J) TO 4096
;
        STA 1,@ADRB
        JMP NEXT
;
;       INNER LOOP
;
BIT:    LDA 1,C16
        STA 1,CNT3      ;COUNTER FOR INNER LOOP
        SUBZL 1,1       ;1
LOOP3:  AND# 1,0,SZR    ;SKIP IF THIS BIT CLEAR
        JMP FOUND       ;BIT FOUND
        MOVZL 1,1       ;TRY NEXT MORE SIG. BIT
        DSZ CNT3        ;DONE 16 BITS ?
        JMP LOOP3       ;NO
        JMP NEXT        ;SHOULD NEVER GET HERE !
;
;       BIT FOUND, SET IDEPTH(I,J) TO K
;
FOUND:  LDA 0,CNT2
        ADDZL 0,0
        ADDZL 0,0
        LDA 1,CNT3
        ADD 1,0         ;AC0=16*CNT2+CNT3
        LDA 1,C80
        SUB 0,1         ;AC1 IS THE 'DEPTH' OF THE FIRST BIT SET
        ADDZL 1,1
        ADDZL 1,1       ;*16 FOR GREATER DEPTH RESOLUTION
        STA 1,@ADRB
NEXT:   ISZ ADRA        ;POINT TO NEXT (I,J)
        ISZ ADRB
        DSZ CNT1        ;DONE 4096 TIMES ?
        JMP LOOP1       ;NO
        JSR @.FRET
;
C4K:    4096.
C4:     4
C16:    16.
C80:    80.
CNT1:   0
CNT2:   0
CNT3:   0
ADRA:   0
ADRB:   0
.END
```

b) The Binary Object manipulation/slicing algorithm

Format : S3DK(IBIN,K)

Input Data : A binary object array declared as in Appendix 3a above, and a depth threshold value K, in the range 0 to 63.

Action : The routine slices away the proximal portion of the object by setting all bits to zero if they are closer to the observer than the chosen depth threshold value.

Output Data : The same binary object array with the nearer voxels erased.

```
;       CGS3DK.SR
;
;       CALL S3DK(IBIN,K)
;
;       SLICES THE 3-D OBJECT IN IBIN(0:63,0:63,0:3) PARALLEL
;       TO THE XY PLANE SO THAT ALL BITS FROM 0 TO K-1 ARE
;       CLEARED.
;
;       CJG 25/8/83
;
.TITLE  S3DK
.EXTD   .FARL,.FRET
.ENT    S3DK
.NREL
;
        7
S3DK:   JSR @.FARL
        LDA 0,-167,3    ;ADDRESS OF IBIN(0,0,0)
        STA 0,ADRA
        LDA 0,@-166,3   ;K
;
;       GET KK=K/16, IF NONZERO THEN SET NEAR WORDS TO 0
;
        MOVZR 0,1
        MOVZR 1,1
        MOVZR 1,1
        MOVZR 1,1,SNR   ;K/16, SKIP IF K > = 16
        JMP DOWN
;
;       SET KK*4096 WORDS TO 0
;
        STA 1,KK
        STA 1,CNT1
LOOP1:  LDA 0,C4K
        STA 0,CNT2
        SUBO 0,0
LOOP2:  STA 0,@ADRA
        ISZ ADRA
        DSZ CNT2
        JMP LOOP2
        DSZ CNT1
        JMP LOOP1
```

```
;
;       FIND THE BIT NUMBER TO CLEAR TO
;
        LDA 1,KK
        ADDZL 1,1
        ADDZL 1,1           ;16*KK
        LDA 0,@-166,3       ;K
        SUB 1,0
DOWN:   MOV# 0,0,SNR        ;SKIP IF NEED TO CLEAR SOME BITS
        JMP RET
;
;       THIS LOOP TO 'AND' IBIN WITH RELEVANT MASK
;
        LDA 2,AMSK
        ADD 0,2             ;POINTER TO MASK
        LDA 1,0,2           ;GET MASK
        LDA 0,C4K
        STA 0,CNT1
LOOP3:  LDA 0,@ADRA
        AND 1,0
        STA 0,@ADRA
        ISZ ADRA
        DSZ CNT1
        JMP LOOP3
RET:    JSR @.FRET
;
ADRA:   0
KK:     0
CNT1:   0
CNT2:   0
C4K:    4096.
AMSK:   M0          ;POINTER TO START OF TABLE
;
M0:     177777      ;THIS TABLE OF MASK VALUES IS USED
M1:     177776      ;TO CLEAR BITS WITHIN A WORD BY
M2:     177774      ;SIMPLE LOGICAL AND INSTRUCTION.
M3:     177770
M4:     177760
M5:     177740
M6:     177700
M7:     177600
M8:     177400
M9:     177000
M10:    176000
M11:    174000
M12:    170000
M13:    160000
M14:    140000
M15:    100000
;
.END
```

c) Creation of lookup tables for coordinate transformation

Format : VTLUT(THETA,PHI,N,LUT)

Input Data : Angles THETA and PHI specify the current viewing direction using the restricted viewing transformation of equation 4.12. The data is sampled at N x N x N resolution, hence object and observer coordinates are restricted to the range 0 to N-1.

Action : Calculates 9 lookup tables containing terms such as x.cos(theta) for all possible integer values of x. Each table contains N elements, and the values stored in the table are converted to a scaled integer representation.

Output Data : Array LUT(N,9) contains the lookup tables from which observer coordinates can be deduced. The table values are calculated from the Transformation Matrix for the restricted viewing transformation. Object coordinates x, y and z are used as indices to obtain table values which are summed to generate x', y' and z' :-

$$x' = LUT(x,1) + LUT(y,2) + LUT(z,3)$$
$$y' = LUT(x,4) + LUT(y,5) + LUT(z,6)$$
$$z' = LUT(x,7) + LUT(y,8) + LUT(z,9)$$

To maintain high precision during these integer additions, the values in the tables are scaled up by a factor 256, and the calculated observer coordinates must therefore be divided by the same factor.

```
C       CGVTLUT.FR
C
        SUBROUTINE VTLUT(THETA,PHI,N,LUT)
C
C       GENERATES 9 LOOK-UP TABLES IN LUT(N,9)
C       CORRESPONDING TO THE VIEWING TRANSFORM
C       SPECIFIED BY THETA AND PHI.
C
C       CJG 19/8/85
C
        DIMENSION LUT(N,9)
        CT=COS(THETA)
        ST=SIN(THETA)
        CP=COS(PHI)
        SP=SIN(PHI)
        CENTRE=FLOAT(N-1)/2.0
C
C       LUT NO.1 IS X.COS(THETA)
C
        DO 10 I=1,N
        R=FLOAT(I-1)*CT
        LUT(I,1)=INT(256.*R)
10      CONTINUE
C
C       LUT NO.2 IS ZERO FOR THIS TRANSFORMATION
C
        DO 20 I=1,N
        LUT(I,2)=0
20      CONTINUE
```

```
C
C       LUT NO.3 IS Z.SIN(THETA)+CENTRE.(1-COS(THETA)-SIN(THETA))
C
        R0=CENTRE*(1.-CT-ST)
        DO 30 I=1,N
        R=FLOAT(I-1)*ST+R0
        LUT(I,3)=INT(256.*R)
30      CONTINUE
C
C       LUT NO.4 IS X.SIN(THETA).SIN(PHI)
C
        DO 40 I=1,N
        R=FLOAT(I-1)*ST*SP
        LUT(I,4)=INT(256.*R)
40      CONTINUE
C
C       LUT NO.5 IS Y.COS(PHI)
C
        DO 50 I=1,N
        R=FLOAT(I-1)*CP
        LUT(I,5)=INT(256.*R)
50      CONTINUE
C
C       LUT NO.6 IS -Z.COS(THETA).SIN(PHI) PLUS AN OFFSET TERM GIVEN BY
C       CENTRE.(1-COS(PHI)+SIN(PHI).COS(THETA)-SIN(THETA).SIN(PHI))
C
        R0=CENTRE*(1.-CP+SP*CT-ST*SP)
        DO 60 I=1,N
        R=-FLOAT(I-1)*CT*SP+R0
        LUT(I,6)=INT(256.*R)
60      CONTINUE
C
C       LUT NO.7 IS -X.SIN(THETA).COS(PHI)
C
        DO 70 I=1,N
        R=-FLOAT(I-1)*ST*CP
        LUT(I,7)=INT(256.*R)
70      CONTINUE
C
C       LUT NO.8 IS Y.SIN(PHI)
C
        DO 80 I=1,N
        R=FLOAT(I-1)*SP
        LUT(I,8)=INT(256.*R)
80      CONTINUE
C
C       LUT NO.9 IS X.COS(THETA).COS(PHI) PLUS AN OFFSET TERM GIVEN BY
C       CENTRE.(1-SIN(PHI)-COS(PHI).COS(THETA)+COS(PHI).SIN(THETA))
C
        R0=CENTRE*(1.-SP-CP*CT+CP*ST)
        DO 90 I=1,N
        R=FLOAT(I-1)*CT*CP+R0
        LUT(I,9)=INT(256.*R)
90      CONTINUE
C
        RETURN
        END
```

d) Coordinate transformation of BTF lists

Format : DFACE(NVOX,LIST,LUT,IMAGE,MODE,N,NSHFT)

Input Data : The array LIST contains NVOX voxels, each represented by three bytes, to store the x, y and z coordinates. The routine assumes that the list is an Ordered Surface list, created by a BTF traversal of the raw data, which was sampled at N x N x N resolution. The array LUT(N,9) contains the lookup table for coordinate transformation (see VTLUT above), and the transformed points are added in sequence to array IMAGE. If MODE is 0, this array contains N x N words, if MODE is 1 then it contains N x N bytes. It is assumed that N will be either 64, 128 or 256, and parameter NSHFT must be specified as either 2, 1 or 0 respectively.

Action : Coordinate transformation is performed extremely rapidly by integer addition of the relevant elements in the 9 lookup tables. The x' and y' coordinates are then used as indices to locate the relevant pixel in the output IMAGE array in which to store the z' coordinate.

Output Data : Array IMAGE contains the z' values of all the voxels in LIST, transformed in sequential order. It is either a word mode array suitable for use as input to a gradient shading algorithm (see SH64 below), or a byte mode array suitable for direct transfer to the image memory.

```
;       CGDFACE.SR
;
;       CALL DFACE(NVOX,LIST,LUT,IMAGE,MODE,N,NSHFT)
;
;       TRANSFORMS SEQUENCE OF NVOX VOXELS IN LIST (3 BYTES EACH, X,Y,Z
;       COORDINATES) USING DATA FROM LUT(N,9).
;
;       OUTPUT IN ARRAY IMAGE :-
;
;       MODE    0       WORD MODE
;               1       BYTE MODE
;
;       CJG 22/1/86
;
.TITLE  DFACE
.EXTD   .FARL,.FRET
.ENT    DFACE
.NREL
;
LDB03=60701
STB02=62201
;
        7
DFACE:  JSR @.FARL
        STA 3,SAC3
;
;       SET UP TABLE OF POINTERS (L1,L2..L9) TO THE DATA IN LUT ARRAY
;
        LDA 0,C9
        STA 0,CNT
        LDA 2,ATBLE     ;POINTS TO START OF TABLE
        LDA 1,-165,3    ;ADDR OF LUT(1,1)
        LDA 0,@-162,3   ;N
        STA 0,N
LOOP2:  STA 1,0,2       ;SAVE ADDR IN L1,L2..L9
        ADD 0,1         ;INCREMENT BY N
        INC 2,2         ;POINT TO NEXT LOCATION IN TABLE
        DSZ CNT
        JMP LOOP2
```

```
;
;       SET UP WORD MODE STORAGE INSTRUCTION OR BYTE MODE
;
        LDA 2,AINS      ;POINTER TO INSTRUCTION TABLE
        LDA 0,@-163,3   ;MODE
        STA 0,MODE
        ADD 0,2
        LDA 1,0,2       ;GET 'STA 0,0,2' OR 'STB02'
        STA 1,@AST1     ;PUT INSTRUCTION IN RELEVANT PLACES
        STA 1,@AST2
        STA 1,@AST3
        STA 1,@AST4
        LDA 2,-164,3    ;ADDR OF IMAGE(1)
        MOV# 0,0,SZR
        MOVZL 2,2       ;IF MODE NON-ZERO, MAKE BYTE POINTER
        STA 2,AIM
        JMP DOWN
;
;       CONSTANTS AND MEMORY LOCATIONS :-
;
AST1:   ST1
AST2:   ST2
AST3:   ST3
AST4:   ST4
C2:     2.
C9:     9.
AIM:    0
I:      0
J:      0
K:      0
II:     0
JJ:     0
MASK:   177400
ATBLE:  L1
AINS:   IN1
IN1:    STA 0,0,2
IN2:    STB02
N:      0
LIMIT:  0
NSHFT:  0
SAC3:   0
CNT:    0
MODE:   0
;
;       SET UP LIMIT VALUE OF N-2, AND SHIFT VALUE NSHFT
;
DOWN:   LDA 0,N
        LDA 1,C2
        SUB 1,0
        STA 0,LIMIT
        LDA 0,@-161,3   ;NSHFT
        STA 0,NSHFT
;
;       MAIN LOOP THROUGH NVOX VOXELS
;
        LDA 0,@-167,3   ;NVOX
        STA 0,CNT
        LDA 2,-166,3    ;ADDR OF LIST(1)
        MOVZL 2,3       ;BPTR TO 1ST BYTE IN LIST
        STA 3,ALIST
        JMP LOOP3
ALIST:  0
LOOP3:  LDA 3,ALIST
        LDB03           ;GET BYTE POINTED TO BY AC3 (X COORDINATE)
        STA 0,I         ;SAVE X IN LOCATION I
        INC 3,3
        LDB03           ;GET Y
        STA 0,J         ;SAVE Y IN LOCATION J
        INC 3,3
        LDB03           ;GET Z
        STA 0,K         ;SAVE Z IN LOCATION K
        INC 3,3
        STA 3,ALIST
```

```
;
;       CALCULATE X', SKIP IF -VE OR EXCEEDS LIMIT
;
        SUBO 0,0        ;CLEAR AC0
        LDA 2,L1        ;POINTS TO START OF LUT #1
        LDA 1,I
        ADD 1,2         ;NOW POINTS TO RELEVANT ELEMENT
        LDA 1,0,2       ;GET THAT ELEMENT, LUT(X,1)
        ADD 1,0         ;AND ADD TO THE RUNNING SUM
        LDA 2,L2        ;POINTS TO START OF LUT #2
        LDA 1,J
        ADD 1,2         ;NOW POINTS TO RELEVANT ELEMENT
        LDA 1,0,2       ;GET THAT ELEMENT, LUT(Y,2)
        ADD 1,0         ;ADD TO RUNNING SUM
        LDA 2,L3        ;POINT TO START OF LUT #3
        LDA 1,K
        ADD 1,2         ;NOW POINTS TO RELEVANT ELEMENT
        LDA 1,0,2       ;GET THAT ELEMENT, LUT(Z,3)
        ADD 1,0         ;AC0=X' AT HIGH PRECISION
;
        MOVZL 0,1,SZC
        JMP NEXT        ;-VE
        LDA 1,MASK      ;HIGH BYTE
        ANDS 1,0
        LDA 1,LIMIT
        SUBZ# 0,1,SNC
        JMP NEXT        ;EXCEEDS LIMIT
        STA 0,II        ;SAVE X' IN LOCATION II
;
;       NOW DO THAT ALL AGAIN FOR Y'...
;
        SUBO 0,0
        LDA 2,L4        ;START OF LUT #4
        LDA 1,I
        ADD 1,2
        LDA 1,0,2       ;GET LUT(X,4)
        ADD 1,0
        LDA 2,L5        ;START OF LUT #5
        LDA 1,J
        ADD 1,2
        LDA 1,0,2       ;GET LUT(Y,5)
        ADD 1,0
        LDA 2,L6        ;START OF LUT #6
        LDA 1,K
        ADD 1,2
        LDA 1,0,2       ;GET LUT(Z,6)
        ADD 1,0         ;AC0=Y' AT HIGH PRECISION
;
        MOVZL 0,1,SZC
        JMP NEXT        ;-VE
        LDA 1,MASK
        ANDS 1,0
        LDA 1,LIMIT
        SUBZ# 0,1,SNC
        JMP NEXT        ;EXCEEDS LIMIT
        STA 0,JJ        ;SAVE Y' IN LOCATION JJ
;
;       ...AND YET AGAIN FOR Z'
;
        SUBO 0,0
        LDA 2,L7        ;START OF LUT #7
        LDA 1,I
        ADD 1,2
        LDA 1,0,2       ;GET LUT(X,7)
        ADD 1,0
        LDA 2,L8        ;START OF LUT #8
        LDA 1,J
        ADD 1,2
        LDA 1,0,2       ;GET LUT(Y,8)
        ADD 1,0
        LDA 2,L9        ;START OF LUT #9
        LDA 1,K
        ADD 1,2
        LDA 1,0,2       ;GET LUT(Z,9)
        ADD 1.0         ;AC0=Z' AT HIGH PRECISION
;
        LDA 1,MODE
        MOV# 1,1,SNR    ;SKIP IF MODE=1 (BYTE MODE)
        JMP PUT         ;NO, JUST OUTPUT THE DEPTH VALUE
        LDA 1,MASK
        ANDS 1,0        ;REDUCE TO LOW PRECISION
                        ;(AC0=DEPTH VALUE)
```

```
;
;       STORE THIS DEPTH VALUE AT IMAGE(II,JJ),IMAGE(II+1,JJ),IMAGE(II,JJ+1)
;       AND IMAGE(II+1,JJ+1)
;
PUT:    LDA 1,JJ
        MOVS 1,1
        LDA 2,NSHFT     ;GET NO.OF SHIFTS (0 IF N=256,1 IF N=128,2 IF N=64)
TEST:   MOV# 2,2,SNR
        JMP DONE        ;ZERO, NO MORE SHIFTS NEEDED
        MOVZR 1,1       ;DIVIDE BY 2
        NEG 2,2
        COM 2,2
        JMP TEST
DONE:   LDA 2,AIM
        ADD 1,2
        LDA 1,II
        ADD 1,2
ST1:    0               ;SAVE IN IMAGE(II,JJ)
        INC 2,2
        JSR CHECK       ;CHECK IF DONE ALREADY, IF SO SKIP NEXT INSTRUCTION
ST2:    0               ;IMAGE(II+1,JJ)
        LDA 1,N
        ADD 1,2
        JSR CHECK
ST3:    0               ;IMAGE(II+1,JJ+1)
        NEG 2,2
        COM 2,2
        JSR CHECK
ST4:    0               ;IMAGE(II,JJ+1)
;
NEXT:   DSZ CNT
        JMP LOOP3
        LDA 3,SAC3
        JSR @.FRET
;
;       SPACE TO STORE POINTERS TO EACH OF THE 9 LOOKUP TABLES
;
L1:     0
L2:     0
L3:     0
L4:     0
L5:     0
L6:     0
L7:     0
L8:     0
L9:     0
;
;       SUBROUTINE TO CHECK WHETHER PIXEL ALREADY SET
;
CHECK:  LDA 1,@AMOD
        MOV# 1,1,SZR
        JMP 0,3         ;ONLY DO THIS CHECK FOR MODE=0
        LDA 1,0,2
        MOV# 1,1,SNR
        JMP 0,3         ;STILL BLANK, GO AND PAINT IT
;
;       NON-BLANK, OVERWRITE ONLY IF OLD DEPTH-NEW DEPTH EXCEEDS
;       AN EMPIRIC THRESHOLD, OTHERWISE AVERAGE
;
        STA 3,RET
        SUB 0,1         ;AC1=OLD-NEW
        MOVZL# 1,1,SZC
        NEG 1,1         ;ABSOLUTE VALUE
        LDA 3,THRE
        SUBZ 1,3,SNC    ;SKIP IF DIFFERENCE <= THRE
        JMP @RET        ;NO, OVERWRITE WITH NEW VALUE
        LDA 1,0,2       ;GET OLD DEPTH
        MOVZR 1,1
        MOVZR 0,0       ;CAREFUL WITH POSSIBLE OVERFLOW !
        ADD 1,0         ;AVERAGE OLD WITH NEW
        JMP @RET
RET:    0
THRE:   1680.
AMOD:   MODE
.END
```

e) Polynomial shading of visible voxel data

Format : SH64(IDEPTH,L,M,N,IAMB,IDIF,KOL)

Input Data : The depth array IDEPTH(0:63,0:63) is to be shaded using the polynomial shading algorithm described in section 5.3, with ambient light intensity IAMB and directional light intensity IDIF. The orientation of the latter illumination is specified by the unnormalised direction cosines L, M and N, relative to the x', y' and z' axes of the observer's coordinate system.

Action : The surface normal is estimated at all points in the array IDEPTH using a local polynomial fitting procedure along both the x' and y' axes. This is then used to calculate the intensity of reflected light as an integer 'shade' value, which is stored in the corresponding pixel of the output KOL array.

Output Data : The array KOL(0:63,0:63) contains shade values for all the visible voxels in the depth array.

```
C       CGSH64.FR
C
        SUBROUTINE SH64(IDEPTH,L,M,N,IAMB,IDIF,KOL)
C
C       DISPLAYS SHADED 64*64 IMAGE BASED ON COSINE LAW.
C
C       IDEPTH          :       DEPTH ARRAY, VALUES * 16
C       L,M,N           :       DIRN COSINES OF LIGHT VECTOR
C       IAMB            :       WEAKEST AMBIENT LIGHT STRENGTH
C       IDIF            :       DIRECTIONAL LIGHT STRENGTH
C
C       KOL             :       RETURNED 64*64 SHADE ARRAY
C
C       CJG 17/1/85
C
        COMMON/CS64/IOFF(2),WT(5,5),KV(5,5),MASK(5),MSKL(3),KVL(3,3)
        DIMENSION IDEPTH(4096),G(2),KOL(4096)
        DATA IOFF/1,64/,MASK/174K,76K,370K,37K,760K/,MSKL/70K,34K,160K/
        DATA KVL/-1,0,1,-2,-1,0,0,1,2/
        DATA KV/-2,-1,0,1,2,-3,-2,-1,0,1,-1,0,1,2,3,-4,-3,-2,-1,0,0,1,2,3,4/
        DATA WT/-14.,-7.,0.,7.,14.,6.,-17.,-20.,-3.,34.,-34.,3.,20.,17.,-6.,
     - 26.,-27.,-40.,-13.,54.,-54.,13.,40.,27.,-26./
C
C       NORMALISE DIRECTION COSINES
C
        P=0.6
        RL=FLOAT(L)
        RM=FLOAT(M)
        RN=FLOAT(N)
        D=SQRT(RL**2+RM**2+RN**2)
        RL=RL/D
        RM=RM/D
        RN=RN/D
        AMB=FLOAT(IAMB)
        DIF=FLOAT(IDIF)
C
C       LOOP THROUGH ALL PIXELS
C
        I=0
        LIMIT=96
        DO 60 JR=0,63
        DO 50 JC=0,63
        I=I+1
        KOLOR=0
        IF(IDEPTH(I).GT.1023)GOTO 3             ;NO VISIBLE VOXEL HERE
```

```
C
C       IDIR=1, GRADIENT IN I DIRECTION; IDIR=2, GRADIENT IN J DIRECTION
C
        DO 40 IDIR=1,2
        MSUM=0
        MV=1
        IADD=IOFF(IDIR)
        II=I-4*IADD
        DO 10 K=1,9
        IF(II.LT.1.OR.II.GT.4096)GOTO 1
        IDIST=IABS(IDEPTH(I)-IDEPTH(II))
        IF(IDIST.GE.LIMIT)GOTO 1
        IF(IDEPTH(II).LT.1024)MSUM=MSUM+MV
1       MV=2*MV
        II=II+IADD
10      CONTINUE
C
C       FIND WHICH OF 5 POSSIBLE 5 POINT COMBINATIONS TO FIT QUADRATIC TO
C
        DO 20 NN=1,5
        IV=IAND(MSUM,MASK(NN))
        IF(IV.EQ.MASK(NN))GOTO 2
20      CONTINUE
C
C       CAN'T FIT QUADRATIC, TRY THREE POSSIBLE LINEAR FITS
C
        DO 25 NN=1,3
        IV=IAND(MSUM,MSKL(NN))
        IF(IV.EQ.MSKL(NN))GOTO 12
25      CONTINUE
        IF(JR.EQ.0.OR.JR.EQ.63)GOTO 3
        IF(JC.EQ.0.OR.JC.EQ.63)GOTO 3
        NN=1                ;FORCE A FIT !
12      II1=I+KVL(1,NN)*IADD
        II2=I+KVL(3,NN)*IADD
        YV1=63.-FLOAT(IDEPTH(II1))/16.
        YV2=63.-FLOAT(IDEPTH(II2))/16.
        G(IDIR)=(YV2-YV1)/2.
        GOTO 40
C
C       CALCULATE GRADIENT OF QUADRATIC AT THE CURRENT PIXEL
C
2       G(IDIR)=0.
        DO 30 MM=1,5
        II=I+KV(MM,NN)*IOFF(IDIR)
        YV=63.-FLOAT(IDEPTH(II))/16.
        G(IDIR)=G(IDIR)+YV*WT(MM,NN)
30      CONTINUE
        G(IDIR)=G(IDIR)/70.
40      CONTINUE
C
C       FROM TWO GRADIENTS CALCULATE SURFACE NORMAL, HENCE SHADE VALUE
C
        D=SQRT(G(1)**2+G(2)**2+1.)
        CT=(RL*G(1)+RM*G(2)+RN)/D
        IF(CT.LT.0.)CT=0.
        DTERM=1.-FLOAT(IDEPTH(I))/1024.
        KOLOR=INT(DTERM*(AMB+DIF*CT**P))
3       KOL(I)=KOLOR
50      CONTINUE
60      CONTINUE
        RETURN
        END
```

f) Transfer of image data to the 256 x 256 display memory

Format : WN256(IAR,IR,IC,NWD,MERGE,IBUF)

Input Data : A row of shaded image pixel values in array IAR(NWD), which contains 2*NWD pixels packed two per word. These are to be transferred to the display memory, starting at location (IR,IC). Coordinates in the image memory are restricted to the range 0 to 255. The parameter MERGE controls the mechanism of transfer, either to overwrite the display pixel with the whole byte value from IAR, or to scale that value into the range 0..15 and then overwrite either the low or high order nibble of the display memory. The buffer array IBUF(NWD) is required for such transfers.

Action : If MERGE is zero, the row of image values is transferred direct to the display memory. Otherwise the corresponding row is first read from the display memory into IBUF, then merged with IAR, before being written back to the display. If MERGE is 1 then the low order nibble is overwritten by values from IAR, if MERGE is 2 then the high order nibble is overwritten.

Output Data : Only values in the display memory are affected by this routine. It is used to display either single images, or two superimposed images in separate nibbles of the display memory.

```
;       CGWN256.SR
;
;       CALL WN256(IAR,IR,IC,NWD,MERGE,IBUF)
;
;       WRITES NWD FROM IAR INTO 256*256 DISPLAY STARTING FROM (IR,IC).
;
;               MERGE   :       0       OVERWRITE DISPLAY CONTENTS
;                       :       1       PUT NEW INTO LOW NIBBLE OF PIXEL
;                       :       2       PUT NEW INTO HIGH NIBBLE
;
;       IN BOTH CASES WHERE MERGE <> 0 THE OTHER NIBBLE IS LEFT UNCHANGED.
;
;       MAKES USE OF THE LIBRARY ROUTINE IO256, CALLED AS A FORTRAN
;       SUBROUTINE IN THE FORM :-
;
;               CALL IO256(IAR,IR,IC,NWD,KODE)
;
;       WHERE THE ARGUMENTS ARE AS FOR WN256, EXCEPT THAT KODE SHOULD
;       BE 0 TO WRITE TO THE DISPLAY MEMORY AND 1 TO READ FROM IT.
;
;       CJG 22/7/83
;
.TITLE  WN256
.EXTD   .FARL,.FRET,.FCAL
.EXTN   IO256
.ENT    WN256
.NREL
;
        17
WN256:  JSR @.FARL
        SUBO 0,0            ;0
        STA 0,-161,3        ;FORTRAN ADDRESS 17
        INC 0,0             ;1
        STA 0,-160,3        ;FORTRAN ADDRESS 20
        LDA 0,@-163,3       ;MERGE
        STA 0,MERGE
        MOV 0,0,SNR
        JMP OUT             ;0, SO JUST OUTPUT IAR
```

```
;
;       READ OLD LINE INTO ARRAY IBUF(NWD)
;
        JSR @.FCAL
        IO256
        5
        @16     ;IBUF
        @12     ;IR
        @13     ;IC
        @14     ;NWD
        20      ;1=READ
;
;       MERGE WITH IAR (LOW OR HIGH NIBBLE)
;
        LDA 0,@-164,3   ;NWD
        STA 0,CNT
        LDA 0,-167,3    ;ADDR OF IAR(1)
        STA 0,ADRA
        LDA 0,-162,3    ;ADDR OF IBUF(1)
        STA 0,ADRB
        STA 3,SAC3
        LDA 2,MASK2     ;TWO HIGH NIBBLES
        LDA 3,MASK3     ;TWO LOW NIBBLES
        LDA 0,MERGE
        MOVR# 0,0,SZC   ;SKIP IF MERGE = 2
        JMP LOW
;
HIGH:   LDA 0,@ADRA     ;IAR(I)
        LDA 1,@ADRB     ;IBUF(I)
        AND 2,0
        AND 3,1
        ADD 0,1         ;REPLACE THE TWO HIGH NIBBLES
        STA 1,@ADRA
        ISZ ADRA
        ISZ ADRB
        DSZ CNT
        JMP HIGH
        JMP DOWN
;
LOW:    LDA 0,@ADRA
        LDA 1,@ADRB
        MOVZR 0,0       ;MOVE HIGH TO LOW NIBBLES
        MOVZR 0,0
        MOVZR 0,0
        MOVZR 0,0
        AND 3,0
        AND 2,1
        ADD 0,1         ;REPLACE TWO LOW NIBBLES
        STA 1,@ADRA
        ISZ ADRA
        ISZ ADRB
        DSZ CNT
        JMP LOW
;
DOWN:   LDA 3,SAC3
OUT:    JSR @.FCAL
        IO256
        5
        @11     ;IBUF
        @12     ;IR
        @13     ;IC
        @14     ;NWD
        17      ;0=WRITE
        JSR @.FRET
;
MERGE:  0
SAC3:   0
CNT:    0
ADRA:   0
ADRB:   0
MASK2:  170360
MASK3:  7417
.END
```

References

Akins et al 1985
Importance of the imaging plane for magnetic resonance imaging of the normal left ventricle.
Akins E W, Hill J A, Fitzsimmons J R, Pepine C J and Williams C M
American Journal of Cardiology **56**:366-372 1985

Altman et al 1986
Three-dimensional CT reformation in children.
Altman N R, Altman D H, Wolfe S A and Morrison G
American Journal of Roentgenology **146**:1261-1267 1986

Anderson et al 1982
Three dimensional viewing system for tomographic scan interpretation.
Anderson R E, Baxter B S, Hitchner L E, Shorthill R W and Johnson S A
Radiology **145**:525-527 1982

Artzy et al 1981
The theory, design, implementation and evaluation of a three dimensional surface detection algorithm.
Artzy E, Frieder G and Herman G T
Computer Graphics and Image Processing **15**:1-24 1981

Axel et al 1983
Three dimensional display of NMR cardiovascular images.
Axel L, Herman G T, Udupa J K, Bottomley P A and Edelstein W A
Journal of Computer Assisted Tomography **7**:172-174 1983

Barber and Skellas 1982
Three dimensional display of radionuclide tomograms.
Barber D C and Skellas I S
p3648 in 'Proceedings of the 3rd World Congress of Nuclear Medicine and Biology',
ed. Raynaud C, (Pergamon Press; Oxford)

Bauer-Kirpes et al 1987
Display of organs and isodoses as shaded 3-D objects for 3-D therapy planning.
Bauer-Kirpes B, Schlegel W, Boesecke R and Lorenz W J
International Journal of Radiation, Oncology, Biology and Physics **13**:135-140 1987

Baxter and Hitchner 1982
Applications of a three dimensional display in diagnostic imaging.
Baxter B and Hitchner L E
p2173 in 'Proceedings of the 3rd World Congress of Nuclear Medicine and Biology',
ed. Raynaud C, (Pergamon Press; Oxford)

Blau 1985
Radiotracers for functional brain imaging.
Blau M
Seminars in Nuclear Medicine **15**:329-334 1985

Blinn and Newell 1976
Texture and reflection in computer generated images.
Blinn J F and Newell M E
Communications of the ACM **19**:542-547 1976

Bloch and Udupa 1983
Application of computerised tomography to radiation therapy and surgical planning.
Bloch P and Udupa J K
Proceedings of the IEEE **71**:351-355 1983

Block et al 1984
Coronary angiographic examination with the dynamic spatial reconstructor.
Block M, Bove A A and Ritman E L
Circulation **70**:209-216 1984

Bories et al 1985
CT in hemispheric ischaemic attacks.
Bories J, Derhy S and Chiras J
Neuroradiology **27**:468-483 1985

Borrello et al 1981
Oblique angle tomography : a restructuring algorithm for transaxial tomographic data.
Borrello J A, Clinthorne N H, Rogers W L, Thrall J H and Keyes J W
Journal of Nuclear Medicine **22**:471-473 1981

Braddick 1982
Binocular vision.
Braddick O J
p192 in 'The Senses'
ed. Barlow H B & Mollon J D (Cambridge University Press; Cambridge)

Burk et al 1985
Three dimensional computed tomography of acetabular fractures.
Burk D L, Mears D C, Kennedy W H, Cooperstein J A and Herbert D L
Radiology **155**:183-188 1985

Charkes 1971
Three dimensional radionuclide imaging : stereoscintiphotography.
Charkes N D
Radiology **98**:335-339 1971

Chamayou et al 1982
A 3-D medical graphics package;interpolation, hidden lines, shadowing.
Chamayou J M F, Schlegel W and Lorenz W J
p3444 in 'Proceedings of the 3rd World Congress of Nuclear Medicine and Biology',
ed. Raynaud C, (Pergamon Press; Oxford)

Chawla et al 1982
An interactive computer graphic system for 3-D stereoscopic reconstruction from serial sections.
Chawla S D, Glass L, Freiwald S and Proctor J W
Computers in Biology and Medicine **12**:223-232 1982

Chen et al 1984
Surface rendering in the cuberille environment.
Chen L S, Herman G T, Reynolds R A and Udupa J K
Technical Report MIPG87, University of Pennsylvania, January 1984

Chen et al 1985a
Interactive manipulation of three dimensional data via a two dimensional device.
Chen L S, Herman G T, Hung H M, Levkowitz H, Trivedi S S and Udupa J K
Optical Engineering **24**:893-900 1985

Chen et al 1985b
Surface shading in the cuberille environment.
Chen L S, Herman G T, Reynolds R A and Udupa J K
IEEE Computer Graphics and Applications **5**:33-43 1985

Christiansen and Sederberg 1978
Conversion of complex contour line definitions into polygonal element mosaics.
Christiansen H N and Sederberg T W
Computer Graphics **12**:187-192 1978

Cook et al 1981
Three dimensional reconstruction from serial sections for medical applications.
Cook P N, Batnitzky S, Lee K R, Levine E, Price H I, Preston D F, Cook L T, Fritz S L, Anderson W and Dwyer S J
Proceedings of the SPIE **283**:98-105 1981

Cook et al 1983
A three dimensional display system for diagnostic imaging applications.
Cook L T, Dwyer S J, Batnitzky S and Lee K R
IEEE Computer Graphics and Applications **3**:13-19 1983

Cutting et al 1986
Computer aided planning and evaluation of facial and orthognothic surgery.
Cutting C, Grayson B, Bookstem F, Fellingham L and McCarthy J G
Clinics in Plastic Surgery **13**:449-462 1986

DeMarino et al 1986
Three dimensional computed tomography in maxillo-facial trauma.
DeMarino D P, Steiner E, Poster R B, Katzberg R W, Hengerer A S, Herman G T, Wayne W S and Prosser D C
Archives of Otolaryngology, Head and Neck Surgery **112**:146-150 1986

DeMontebello 1977
The synthaliser for three dimensional synthesis and analysis by optical dissection.
DeMontebello R L
Procedures of the SPIE **120**:184 1977

Doctor and Torborg 1981
Display techniques for octree-encoded objects.
Doctor L J and Torborg J G
IEEE Computer Graphics and Applications **1**:29-38 1981

DuFresne et al 1987
Volumetric quantification of intracranial and ventricular volume following cranial vault remodelling.
DuFresne C R, McCarthy J G, Cutting C B, Epstein F J and Hoffman W Y
Plastic and Reconstructive Surgery **79**:24-32 1987

Farrell 1983
Color display and interactive interpretation of three dimensional data.
Farrell E J
IBM Journal of Research and Development **27**:356-366 1983

Farrell et al 1984
Color 3-D imaging of normal and pathological intracranial structures.
Farrell E J, Zappulla R and Yang W C
IEEE Computer Graphics and Applications **4**:5-17 1984

Fishman et al 1987
Volumetric rendering techniques : applications for three dimensional imaging of the hip.
Fishman E K, Drebin B, Magid D, Scott W W, Ney D R, Brooker A F, Riley L H, St Ville J A, Zerhouni E A and Siegelman S S
Radiology **163**:737-738 1987

Folland et al 1977
The radionuclide ejection fraction : a comparison of three techniques with contrast angiography.
Folland E D, Hamilton G W, Larson S M, Kennedy J W, Williams D L and Ritchie J L
Journal of Nuclear Medicine **18**:1159-1166 1977

Fram et al 1982
Three dimensional display of the heart, aorta, lungs and airway using CT.
Fram E K, Godwin J D and Putnam C E
American Journal of Roentgenology **139**:1171-1176 1982

Frieder et al 1985
Back-to-front display of voxel based objects.
Frieder G, Gordon D and Reynolds R A
IEEE Computer Graphics and Applications **5**:52-60 1985

Fuchs et al 1977
Optimal surface reconstruction from planar contours.
Fuchs H, Kedem Z M and Uselton S P
Communications of the ACM **20**:693-702 1977

Fuchs et al 1982
Adding a true 3-D display system to a raster graphic system.
Fuchs H, Pizer S M, Tsai L C, Bloomberg S H and Heinz R E
IEEE Computer Graphics and Applications **2**:73-78 1982

Fuchs et al 1983
Near real time shaded display of rigid objects.
Fuchs H, Abram G D and Grant E D
Computer Graphics **17**:65-72 1983

Garcia et al 1985
Quantification of rotational thallium-201 myocardial tomography.
Garcia E V, Van Train K, Maddahi J, Prigent F, Friedman J, Areeda J, Waxman A and Berman D S
Journal of Nuclear Medicine **26**:17-26 1985

Gibbons 1985
'Non-parametric methods for quantitative analysis'.
Gibbons J D
(American Sciences Press; Columbus)

Gibson 1983
A new method for the three dimensional display of tomographic images.
Gibson C J
Physics in Medicine and Biology **28**:1153-1157 1983

Gibson 1986a
Interactive display of three-dimensional radionuclide distributions.
Gibson C J
Nuclear Medicine Communications **7**:475-487 1986

Gibson 1986b
Real time display of three dimensional medical images.
Gibson C J
p229 in 'Proceedings of the 2nd International Conference on Image Processing and its Applications'
Conference Publication 265 (IEE; London)

Gibson 1987
Real time 3-D display of the beating heart (abs).
Gibson C J
Nuclear Medicine Communications **8**:235 1987

Gibson 1988
Real time 3-D display of gated blood pool tomograms.
Gibson C J
Physics in Medicine and Biology, in press.

Gibson et al 1985
New technique for showing the relation of tomographic myocardial perfusion images obtained with thallium-201 to the coronary arteries.
Gibson C J, Laird E E, Williams E D, Rajathurai A, Mittra B and Rankin D
British Heart Journal **54**:367-374 1985

Gillespie et al 1987
Three dimensional CT reformations in the assessment of congenital and traumatic cranio-facial deformities.
Gillespie J E, Quayle A A, Barker G and Isherwood I
British Journal of Oral and Maxillofacial Surgery **25**:171-177 1987

Goldwasser et al 1985
Physicians workstation with real-time performance.
Goldwasser S M, Reynolds R A, Bapty T, Baraff D, Summers J, Talton D A and Walsh E
IEEE Computer Graphics and Applications 5:44-57 1985

Gonzalez and Wintz 1977
'Digital image processing'.
Gonzalez R C and Wintz P
(Addison-Wesley : Massachusetts)

Gordon and Reynolds 1983
Image space shading of three dimensional objects.
Gordon D and Reynolds R A
Technical Report MIPG95, University of Pennsylvania, 1983

Greguss 1977
Holographic displays for computer assisted tomography.
Greguss P
Journal of Computer Assisted Tomography 1:184 1977

Grotch 1983
Three dimensional and stereoscopic graphics for scientific data display and analysis.
Grotch S L
IEEE Computer Graphics and Applications 3:31-43 1983

Harris 1982
Display and analysis of medical volume images.
Harris L D
p2181 in 'Proceedings of the 3rd World Congress of Nuclear Medicine and Biology',
ed. Raynaud C, (Pergamon Press; Oxford)

Harris et al 1982
Myocardial imaging with dipyridamole: comparison of the sensitivity and specificity of Tl-201 versus MUGA.
Harris D, Taylor D, Condon B, Ackery D and Conway N
European Journal of Nuclear Medicine 7:1-5 1982

Harris et al 1986
Three dimensional display and analysis of tomographic volume images utilising a varifocal mirror.
Harris L D, Camp J J, Ritman E L and Robb R A
IEEE Transactions on Medical Imaging 5:67-72 1986

Heffernan and Robb 1985
A new method for shaded surface display of biological and medical images.
Heffernan P B and Robb R A
IEEE Transactions on Medical Imaging 4:26-38 1985

Herman and Liu 1979
Three dimensional display of human organs for computed tomograms.
Herman G T and Liu H K
Computer Graphics and Image Processing **9**:1-21 1979

Herman and Udupa 1981
Display of three-dimensional discrete surfaces.
Herman G T and Udupa J K
Proceedings of the SPIE **283**:90-97 1981

Herman and Udupa 1982
Display of multiple surfaces.
Herman G T and Udupa J K
p2165 in 'Proceedings of the 3rd World Congress of Nuclear Medicine and Biology',
ed. Raynaud C, (Pergamon Press; Oxford)

Herman and Udupa 1983
Display of 3-D digital images: computational foundations and medical applications.
Herman G T and Udupa J K
IEEE Computer Graphics and Applications **3**:39-46 1983

Hesselink 1985
Non-holographic three dimensional display devices.
Hesselink L
Proceedings of the SPIE **532**:14-28 1985

Hohne and Bernstein 1986
Shading 3-D images from CT using grey scale gradients.
Hohne K H and Bernstein R
IEEE Transactions on Medical Imaging **5**:45-47 1986

Holman 1979
Thallium-201: when should we use it ?
Holman B L
Journal of Nuclear Medicine **20**:1206-1208 1979

Jackins and Tanimato 1980
Oct-trees and their use in representing three dimensional objects.
Jackins C L and Tanimato S L
Computer Graphics and Image Processing **14**:249-270 1980

Jackson et al 1983
The display of multiple images derived from ECAT.
Jackson P C, Davies E R, Goddard P R and Wilde R P H
Nuclear Medicine Communications **4**:348-353 1983

Jackson et al 1984
Three dimensional display of the cardiac blood pool.
Jackson P C, Fraser J R and Wilde R P H
Journal of Australasian Physical and Engineering Sciences in Medicine **7**:115-119 1984

Jaman et al 1985
Display of 3-D anisotropic images from limited view computed tomograms.
Jaman K A, Gordon R and Rangayyan R M
Computer Vision, Graphics and Image Processing **30**:345-361 1985

Jannson and Kosowsky 1984
Display of moving volumetric images.
Jannson D G and Kosowsky R P
Proceedings of the SPIE **507**:82-92 1984

Jimenez et al 1986
Computer graphics display method for visualising three-dimensional biological structures.
Jimenez J, Santisteban A, Carazo J M and Carrascosa J L
Science **232**:113-115 1986

Julesz 1971
'Foundations of Cyclopean Perception'.
Julesz B
(University of Chicago Press; Chicago)

Kalff et al 1984
Liver-spleen studies with the rotating gamma camera. I: Utility of the rotating display.
Kalff V, Satterlee W, Harkness B A and Keyes J W
Radiology **153**:533-536 1984

Kehtarnavaz et al 1984
A novel surface reconstruction and display method for cardiac PET imaging.
Kehtarnavaz N, Phillippe E A and DeFigueiredo R J P
IEEE Transactions on Medical Imaging **3**:108-115 1984

Kennedy and Nelson 1987
Three dimensional display from cross-sectional tomographic images: an application to magnetic resonance imaging.
Kennedy D N and Nelson A C
IEEE Transactions on Medical Imaging **6**:134-140 1987

Keppel 1975
Approximating complex surfaces by triangulation of contour lines.
Keppel E
IBM Journal of Research and Development **19**:2-11 1975

Keyes 1982
Image motion and depth perception.
Keyes J W
p2169 in 'Proceedings of the 3rd World Congress of Nuclear Medicine and Biology',
ed. Raynaud C, (Pergamon Press; Oxford)

Keyes and Rogers 1981
Non-tomographic three dimensional imaging with a rotating camera.
Keyes J W and Rogers W L
Nuclear Medicine Communications **2**:26 1981

Keyes et al 1984
Liver-spleen studies with the rotating gamma camera. II : Utility of tomography.
Keyes J W, Singer D, Satterlee W, Kalff V and Harkness B A
Radiology **153**:537-541 1984

Kursunoglu et al 1986
Three dimensional computed tomographic analysis of the normal temperomandibular joint.
Kursunoglu S, Kaplan P, Resnick D and Sartoris D J
Journal of Oral and Maxillofacial Surgery **44**:257-259 1986

Lenz et al 1986
Display of density volumes.
Lenz R, Gudmundsson B, Lindskag B and Daniellson P E
IEEE Computer Graphics and Applications **6**:20-29 1986

Liu et al 1986
Angiography for delineation of systemic-to-pulmonary shunts in congenital pulmonary atresia : evaluation with the dynamic spatial reconstructor.
Liu Y H, Mair D D, Hagler D J, Seward J B, Julorud P R and Ritman E L
Proceedings of the Mayo Clinic **61**:932-941 1986

Maravilla 1978
Computer reconstructed sagittal and coronal computed tomography head scans.
Maravilla K R
Journal of Computer Assisted Tomography **2**:120 1978

Marino et al 1980
The use of computer imaging techniques to visualise cardiac muscle cells in three dimensions.
Marino T A, Cook P N, Cook L T and Dwyer S J
The Anatomical Record **198**:537-546 1980

Marsh and Vannier 1983a
Surface imaging from computerised tomographic scans.
Marsh J L and Vannier M W
Surgery **94**:159-165 1983

Marsh and Vannier 1983b
The third dimension in craniofacial surgery.
Marsh J L and Vannier M W
Plastic and Reconstructive Surgery **13**:759-767 1983

Marsh et al 1986
Applications of computer graphics in cranio-facial surgery.
Marsh J L, Vannier M W, Bresina S and Hemmer K M
Clinics in Plastic Surgery **13**:441-448 1986

Mazziotta and Huang 1976
THREAD (three dimensional reconstruction and display) with biomedical applications in neuron ultrastructure and computerised tomography.
Mazziotta J C and Huang K H
Proceedings of the American Federation of Information Processing Societies **45**:241 1976

McClaren 1987
Technegas - a new ventilatory agent (abs).
McClaren C
Clinical Physics and Physiological Measurement 8:191 1987

Meagher 1984
Interactive solids processing for medical analysis and planning.
Meagher D J
p96 in 'Proceedings of the 5th Annual General Conference of the National Computer Graphics Association', 1984

Minato et al 1982
Three dimensional display of single photon ECT imaging.
Minato K, Mukai T, Yamamoto K, Tamaki N, Maeda H, Yonekura Y, Ito H, Isii Y, Torizuka K, Komori M, Hirakawa A, Maekawa T and Kuwahara M
p3651 in 'Proceedings of the 3rd World Congress of Nuclear Medicine and Biology',
ed. Raynaud C, (Pergamcn Press; Oxford)

Moore et al 1980
ECG gated emission tomography of the cardiac blood pool.
Moore M L, Murphy P H and Burdine J A
Radiology **134**:233-235 1980

Nakamura 1984
Three dimensional digital display of ultrasonograms.
Nakamura S
IEEE Computer Graphics and Applications **4**:36 1984

Newman and Sproull 1973
'Principles of Interactive Computer Graphics'.
Newman W M and Sproull R F
(McGraw-Hill: New York)

Olson et al 1981
Calculation of transverse ventricular sections from transaxial ECT reconstructions of the myocardium.
Olson D O, Williams D L, Ritchie J L, Harp G D, Hamilton G W, Gullberg G T, Eisner R L and Nowak D J
p167 in 'Functional Mapping of Organ Systems',
ed. Esser P D (Society of Nuclear Medicine; New York)

Pate et al 1986
Perspective: three dimensional imaging of the musculoskeletal system.
Pate D, Resnick D, Andre M, Sartoris D J, Kursunoglu S, Bielicki D, Dev P and Vassiliadis A
American Journal of Roentgenology **147**:545-551 1986

Pavel et al 1977
In vivo labelling of red blood cells with Tc-99m : a new approach to blood pool visualisation.
Pavel D G, Zimmer A M and Patterson V N
Journal of Nuclear Medicine **18**:305-308 1977

Perkins and Green 1982
Three dimensional reconstruction of biological sections.
Perkins W J and Green R J
Journal of Biomedical Engineering **4**:37-43 1982

Phong 1975
Illumination for computer generated pictures.
Phong B T
Communications of the ACM **18**:311-317 1975

Pizer et al 1982
Varifocal mirror display of organ surfaces from CT scans.
Pizer S M, Fuchs H, Heinz E R, Bloomberg S H and Tsai L C
p2177 in 'Proceedings of the 3rd World Congress of Nuclear Medicine and Biology',
ed. Raynaud C, (Pergamcn Press; Oxford)

Pohost et al 1977
Differentiation of transiently ischaemic from infarcted myocardium by serial imaging after a single dose of thallium-201.
Pohost G M, Zir L M, Moore R H, McKusick K A, Guiney T E
Beller G A
Circulation **55**:294-302 1977

Preston et al 1982
Three dimensional reconstruction and visualisation system for medical images.
Preston D F, Batnitzky S, Lee K R, Cook P N, Cook L T and Dwyer S J
p518 in 'Proceedings of the 3rd World Congress of Nuclear Medicine and Biology',
ed. Raynaud C, (Pergamon Press; Oxford)

Radermacher and Frank 1984
Representation of three-dimensionally reconstructed objects in electron microscopy by surfaces of equal density.
Radermacher M and Frank J
Journal of Microscopy **136**:77-85 1984

Ratib et al 1982
Phase analysis of radionuclide ventriculograms for the detection of coronary artery disease.
Ratib O, Henze E, Schon H and Schelbert H R
American Heart Journal **104**:1-12 1982

Ridgway and Sharp 1986
Simple display routines for NMR proton density and T1 images.
Ridgway J P and Sharp P F
Physics in Medicine and Biology **31**:171-179 1986

Ritchie et al 1982
Transaxial tomography with thallium-201 for detecting remote myocardial infarction.
Ritchie J L, Williams D, Harp G, Stratton J and Caldwell H
Americal Journal of Cardiology **50**:1236-1241 1982

Roberts et al 1984
Three dimensional imaging and display of the temperomandibular joint.
Roberts D, Pettigrew J, Udupa J K and Ram C
Oral Surgery **58**:461-474 1984

Rothman et al 1984
Computed tomography of the spine: curved coronal reformations from serial images.
Rothman S L G, Dobben G D, Rhodes M L, Glenn W V and Yu-Ming A
Radiology **150**:181-190 1984

Salyer et al 1986
Three dimensional CAT scan reconstruction - pediatric patients.
Salyer K E, Taylor D P and Billmire D E
Clinics in Plastic Surgery **13**:463-474 1986

Savitzky and Golay 1964
Smoothing and differentiation of data by simplified least squares procedures.
Savitzky A and Golay M J E
Analytical Chemistry **36**:1627-1639 1964

Siddon and Kijewski 1982
Perspective display of patient external and internal contours.
Siddon R L and Kijewski P
Computers in Biology and Medicine **12**:217-221 1982

Smye and Unsworth 1985
The use of single photon emission computed tomography in lung imaging with aerosols.
Smye S W and Unsworth J
Nuclear Medicine Communications **6**:397-404 1985

Strauss et al 1979
Of linen and laces - the eighth anniversary of the gated blood pool scan.
Strauss H W, McKusick K A, Boucher C A , Bingham J B and Pohost G M
Seminars in Nuclear Medicine **9**:296-309 1979

Sunguroff and Greenberg 1978
Computer generated images for medical applications.
Sunguroff A and Greenberg D
Computer Graphics **12**:196-202 1978

Sutherland et al 1974
A characterisation of ten hidden surface algorithms.
Sutherland I E, Sproull R F and Schumacker R A
ACM Computing Surveys **6**:1 1974

Tamaki et al 1984
Segmental analysis of stress thallium myocardial emission tomography for localisation of coronary artery disease.
Tamaki N, Yonekura Y, Mukai T, Fajita T, Nohara R, Kadota K, Kambara H
European Journal of Nuclear Medicine 9:99-105 1984

Tauxe et al 1982
Determination of organ volume by single photon emission tomography.
Tauxe W N, Soussaline F, Todd-Pokropek A, Cao A, Collard P, Richard S, Raymond C and Itti R
Journal of Nuclear Medicine **23**:984-987 1982

Totty and Vannier 1984
Complex musculo-skeletal anatomy: analysis using three dimensional surface reconstruction.
Totty W G and Vannier M W
Radiology **150**: 173-177 1984

Trivedi et al 1986
Segmentation into three classes using gradients.
Trivedi S S, Herman G T and Udupa J K
IEEE Transactions on Medical Imaging 5:116-119 1986

Tuy and Tuy 1984
Direct 2-D display of 3-D objects.
Tuy H K and Tuy L T
IEEE Computer Graphics and Applications **4**:29-33 1984

Tuy and Udupa 1982
Representation, display and manipulation of 3-D discrete scenes.
Tuy H and Udupa J K
Technical Report MIPG66, University of Pennsylvania, 1982

Udupa 1981
Determination of 3-D shape parameters from boundary information.
Udupa J K
Computer Graphics and Image Processing **17**:52-59 1981

Udupa 1982
Interactive segmentation and boundary surface formation for 3-D digital images.
Udupa J K
Computer Graphics and Image Processing **18**:213-235 1982

Udupa 1983
Display of 3-D information in 3-D scenes produced by computerised tomography.
Udupa J K
Proceedings of the IEEE **71**:420-431 1983

Udupa et al 1982
Boundary detection in multidimensions.
Udupa J K, Srihari S N and Herman G T
IEEE Transactions on Pattern Analysis and Machine Intelligence 4:41-50 1982

Vannier and Marsh 1983
Craniofacial disorders.
Vannier M W and Marsh J L
Diagnostic Imaging 5:36-43 1983

Vannier et al 1983
Three dimensional computer graphics for cranio-facial surgical planning and evaluation.
Vannier M W, Marsh J L and Warren J O
Computer Graphics 17:263-273 1983

Vannier et al 1984
Three dimensional CT reconstruction images for craniofacial surgical planning and evaluation.
Vannier M W, Marsh J L and Warren J O
Radiology 150:179-184 1984

Verbout et al 1983
A three dimensional graphic reconstruction method of the vertebral column from CT scans.
Verbout A J, Falke T H M and Tinkelenberg J
European Journal of Radiology 3:167-170 1983

Webb et al 1987
Three dimensional display of data obtained by single photon emission computed tomography.
Webb S, Ott R J, Flower M A, MacCready V R and Meller S
British Journal of Radiology 60:557-562 1987

Williams 1981
Three dimensional radionuclide images (letter).
Williams E D
Journal of Nuclear Medicine 22:193 1981

Williams 1983
Viewing multiple section SPECT images.
Williams E D
Nuclear Medicine Communications 4:215-218 1983

Wind et al 1986
Computer graphic modelling in surgery.
Wind G, Dvorak B K and Dvorak J A
Orthopaedic Clinics of North America 17:657-668 1986

Wixson 1983
Four dimensional processing tools for cardiovascular data.
Wixson S E
IEEE Computer Graphics and Applications 3:53-59 1983

Publications

Material included in this thesis has been presented at the following scientific meetings :-

A new method for three dimensional display of ECAT images
Gibson C. J.
11th Annual Meeting of the British Nuclear Medicine Society, London, April 18th-20th 1983

A new technique for displaying tomographic images of perfused myocardium obtained using Tl-201, and their relationship to the coronary arteries
Gibson C. J., Laird E. E., Williams E. D., Rajathurai A. and Mittra B.
Autumn Meeting of the British Cardiac Society, London, September 1984

Three dimensional display of radionuclide and CT images
Gibson C. J.
42nd Annual Conference of the Hospital Physicists' Association, Southampton, September 25th-28th 1985

Real time display of three dimensional medical images
Gibson C. J.
2nd International Conference on Image Processing and its Applications, organised by the Institution of Electrical Engineers, London, June 24th-26th 1986

Real time 3-D display of the beating heart
Gibson C. J.
Awarded 1st Prize at the 15th Annual Meeting of the British Nuclear Medicine Society, London, April 13th-15th 1987

Material included in this thesis has also been published in the following journals :-

A new method for the three-dimensional display of tomographic images
Gibson C. J.
Physics in Medicine and Biology, **28**, 1153-1157, 1983

Image Display
Gibson C. J.
Chapter 7 in 'An Introduction to Emission Computed Tomography', edited by E. D. Williams, Institute of Physical Sciences in Medicine Report Number 44, London 1985

New technique for showing the relation of tomographic myocardial perfusion images obtained with thallium-201 to the coronary arteries
Gibson C. J., Laird E. E., Williams E. D., Rajathurai A., Mittra B. and Rankin D.
British Heart Journal, **54**, 367-374, 1985

Interactive display of three-dimensional radionuclide distributions
Gibson C. J.
Nuclear Medicine Communications, **7**, 475-487, 1986

Real time 3-D display of gated blood pool tomograms
Gibson C. J.
Physics in Medicine and Biology, in press

Phys. Med. Biol., 1983, Vol. 28, No. 10, 1153–1157. Printed in Great Britain

Technical Note

A new method for the three-dimensional display of tomographic images

C J Gibson

Regional Medical Physics Department, Durham Unit, Dryburn Hospital, Durham, England

Received 9 March 1983, in final form 26 May 1983

1. Introduction

Many medical imaging modalities are capable of producing tomographic images showing the three-dimensional distribution of a physical parameter such as attenuation coefficient or radionuclide concentration. These images are usually displayed as a series of parallel slices, and the viewer must correlate the appearance of the cross-section of an object in any one slice with those in adjacent slices to obtain an impression of the overall shape of the object. A truly three-dimensional display would help the viewer perform this difficult conceptual task. Methods proposed so far include the use of a varifocal mirror to produce a virtual three-dimensional image (Pizer *et al* 1982), rapid sequential display of the original projection images (Williams 1981), which may be combined with stereoscopic images (Barber and Skellas 1982), and digital reprojection of the reconstructed volume with selective enhancement of features of interest (Harris 1981).

An alternative approach uses the method of surface detection and shaded surface display (Herman and Liu 1979, Herman and Natterer 1981). Shaded surface images do not include all the information present in the original data, but do allow the observer to infer the shape of the object from its visible surfaces. Most implementations of this technique have used a network of triangular patches to cover the surface of the objects (Cook *et al* 1981, Minato *et al* 1982). This method can be used to produce high quality images but is difficult to apply to objects with complex topology. It is the purpose of this note to describe a simple, rapid method for the display of three-dimensional surfaces which can be applied to complex objects, and is therefore ideally suited to the display of medical images. The method will be illustrated with reference to radionuclide tomography (ECAT) but could be applied to the data from any tomographic imaging modality.

2. Method

The original data consists of a set of volume elements (voxels) arranged in a cubic array. The contents of each voxel specify what may be loosely termed the radionuclide concentration at that point in space. It should be noted that even after attenuation correction the reconstructed counts are not directly proportional to radionuclide concentration (Flower and Parker 1980). This set is then segmented, i.e., divided into voxels considered to be part of the object(s), and those which are not. This process

is always subjective, and a simple thresholding algorithm (Srihari 1981) has been found to give satisfactory results when used interactively. More complex segmentation schemes, based on gradient searches, should give better results where the radionuclide concentration is inhomogeneous within the organ of interest. Note that the segmented array may be stored in binary form, greatly reducing the data storage required. The extraction of the set of visible voxels from a cubic array is considerably simpler than the general problem of hidden surface removal (Artzy *et al* 1981). However, maintaining the viewing direction parallel to one of the axes of the cubic array makes this step trivial. Note that the object may be viewed from different aspects by rotating the data within the 3D array prior to the determination of visible voxels.

Finally, the surface formed by the set of visible voxels is illuminated with an imaginary light source, and a shading value computed for each voxel. This represents the light intensity reflected from the surface, and using a simple model is proportional to the cosine of the angle of incidence of the surface illumination (Newman and Sproull 1973). More complex shading schemes could include surface texture (Blinn 1978) and shadows (Atherton *et al* 1978) at the cost of increased computation time.

3. Results

3.1. Hardware

The method has been applied to tomographic sections obtained with an IGE 400T rotating gamma camera and reconstructed at 64×64 resolution using a Link Dyanne nuclear medicine computer system. The three-dimensional shaded images were produced on a 256×256 display with 256 grey scale levels. Execution times for rotation of the data, visible voxel determination, shading and display were between 25 and 50 s, depending on the size of the final displayed image.

3.2. Liver and spleen scans

ECAT images of a normal liver and spleen were obtained following injection of 80 MBq of $^{99}Tc^{m}$-tin colloid. Transverse slices were reconstructed from 64 views

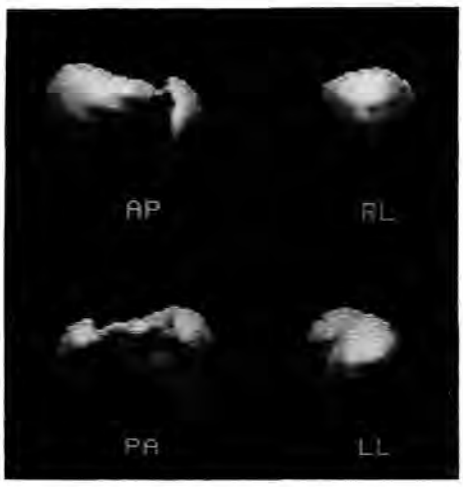


Figure 1. A normal liver and spleen seen in the AP, RL, PA and LL projections.

taken at equiangular intervals around the patient, and the binary object created using the techniques of section 2. Figure 1 shows four views of the liver and spleen from anterior, posterior and lateral aspects. The display of these external surfaces enables the observer to perceive the overall shape of the object, but conceals any internal structure.

Removal of outer layers of the object can, however, reveal voids or other internal details. An elementary and rapid method for removal of outer layers uses a modified binary object, obtained by setting all voxels to zero if they are closer to the observer than some chosen value. If this modification is applied prior to the determination of visible voxels, then the image obtained will show the original object 'sliced' at right angles to the observer's line of sight. Figure 2 shows the same liver and spleen rotated through 90° to display the superior aspect, with increasing amounts removed, showing a simulated defect in the right lobe. The shape and extent of the void are readily determined, together with its position relative to the normal liver anatomy.

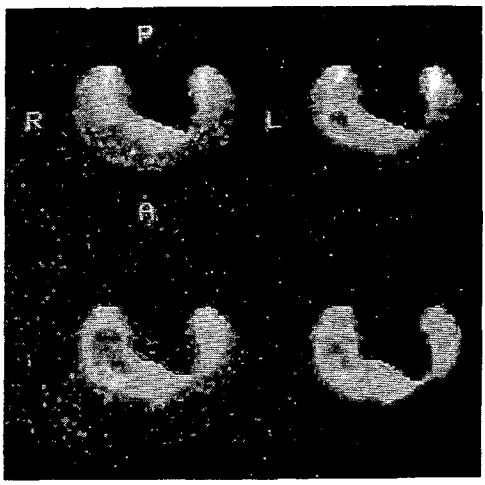


Figure 2. Superior aspect of liver and spleen with patient lying prone. As increasing amounts of the binary object are removed, the void in the right lobe of the liver becomes visible.

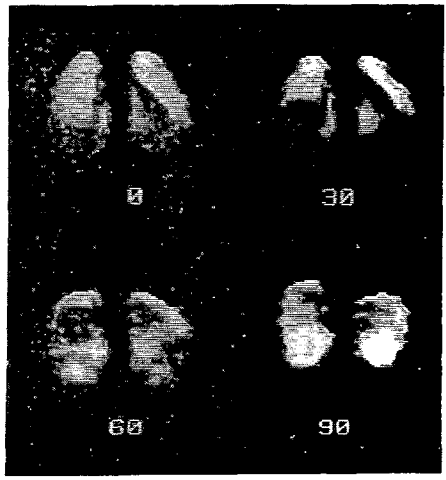


Figure 3. Normal lung perfusion scan showing the lungs in the AP projection tilted backwards through 0°, 30°, 60° and 90°. The cardiac impression is seen most clearly at 30° of tilt.

3.3. Lung scans

ECAT images of normal lungs were obtained following injection of 80 MBq of $^{99}Tc^{m}$-MAA. Following reconstruction and creation of the binary object, shaded images were produced showing the anterior, antero-inferior and inferior aspects (figure 3). The cardiac impression is clearly seen, thus indicating the size and position of the heart, as well as the distribution of perfused lung tissue.

3.4. Brain scans

ECAT images of a normal brain were obtained following injection of 400 MBq of $^{99}Tc^{m}$-eluate. Figure 4 shows a comparison between shaded images of the left and right halves of the binary object, and conventional right and left lateral projection views. The normal brain cavity is seen as a void with no radionuclide present. By

Figure 4. A comparison between shaded surface and conventional projection images of a normal brain scan, showing the RL and LL projections.

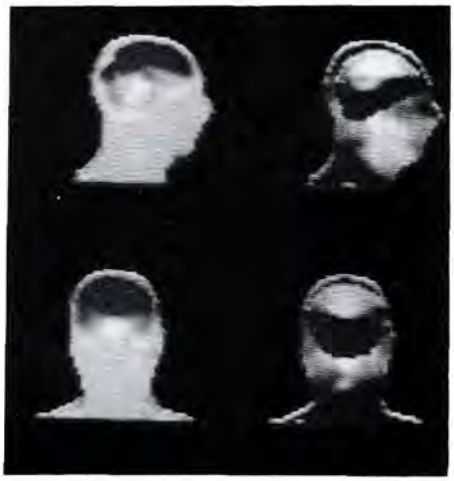


Figure 5. A comparison between shaded surface images of the original brain scan data, and of the logical complement of that data, showing the RL and AP aspects. In the case of the complemented data the brain is seen as a solid object in front of the concave distal surface of the original object.

obtaining the logical complement of the binary object, voids become solids and vice versa. Figure 5 shows the left and posterior halves of the same binary object, together with the corresponding segments of the complemented object. The latter images show the brain as a solid object in front of the concave distal surface of the binary object.

4. Discussion

Shaded surface images have the ability to display three-dimensional information in two-dimensional form. They can therefore be produced on conventional displays, and allow hard copies to be obtained. However, they do not include information about inhomogeneities beneath the displayed surface. A technique has been described for the production of shaded surface images, and simple procedures such as slicing and complementing have been shown to overcome some of the limitations of surface displays. It is suggested that this form of display may be a useful adjunct to conventional displays, especially for complex 3D distributions.

References

Artzy E, Frieder G and Herman G T 1981 *Comput. Graphics & Image Process.* **15** 1

Atherton P, Weiler K and Greenberg D 1978 *Comput. Graphics* **12** 275

Barber D C and Skellas I S 1982 in *Proc. 3rd World Congr. of Nucl. Med. and Biol. August 1982, Paris* (Oxford; Pergamon) p 3648

Blinn J F 1978 *Comput. Graphics* **12** 286

Cook P N, Batnitsky S, Lee K R, Levine E, Price H I, Preston D F, Cook L T, Fritz S L, Anderson W and Dwyer S J 1981 in *Proc. SPIE* **283** 98

Flower M A and Parker R P 1980 *Radiology* **137** 535

Harris L D 1981 in *Proc. SPIE* **283** 106

Herman G T and Liu H K 1979 *Comput. Graphics & Image Process.* **9** 1

Herman G T and Natterer F 1981 in *Mathematical Aspects of Computerized Tomography*, Lecture Notes in Medical Informatics 8 (Berlin: Springer-Verlag)

Minato K, Mukai T, Yamamato K, Tamaki N, Maeda H, Yonekura Y, Ito H, Ioii Y, Torizuka K, Komori M, Hirakawa A, Maekawa T and Kuwahara M 1982 in *Proc. 3rd World Congr. of Nucl. Med. and Biol. August 1982, Paris* (Oxford: Pergamon) p 3651

Newman W M and Sproull R F 1973 in *Principles of Interactive Computer Graphics* (McGraw-Hill: New York)

Pizer S M, Fuchs H, Heinz E R, Bloomberg S H and Tsai L C 1982 in *Proc. 3rd World Congr. of Nucl. Med. and Biol. August 1982, Paris* (Oxford: Pergamon) p 2177

Srihari S N 1981 *Comput. Surv.* **13** 399

Williams E D 1981 *J. Nucl. Med.* **22** 193

Reprinted from Br Heart J 1985; 54: 367–74

New technique for showing the relation of tomographic myocardial perfusion images obtained with thallium-201 to the coronary arteries

CHRISTOPHER J GIBSON, ELSIE E LAIRD, E DAVID WILLIAMS, A RAJATHURAI, BRAJ MITTRA, DONALD RANKIN

British Medical Association, Tavistock Square, London WC1H 9JR

Br Heart J 1985; **54**: 367–74

New technique for showing the relation of tomographic myocardial perfusion images obtained with thallium-201 to the coronary arteries

CHRISTOPHER J GIBSON,* ELSIE E LAIRD,† E DAVID WILLIAMS,†
A RAJATHURAI,‡ BRAJ MITTRA,‡ DONALD RANKIN†

*From the Regional Medical Physics Department, *Durham Unit, Dryburn Hospital, Durham; and †Sunderland Unit, Sunderland District General Hospital; and the ‡Department of Medicine, Sunderland District General Hospital, Sunderland, Tyne and Wear*

SUMMARY A new technique has been developed for presenting myocardial tomograms that allows the observer to perceive the shape of the thallium-201 distribution directly. The surface of the myocardium was found by applying an interactive thresholding technique to a set of conventional transverse slices. Computer graphics techniques were used to display a shaded image of that surface on a television screen, showing the three dimensional shape of the myocardial surface from any chosen aspect. A set of normal preserved coronary arteries was digitised, and using scaling and transformation techniques these arteries were mapped on to the myocardial tomograms and a shaded surface image produced with superimposed coronary arteries. This provided a familiar anatomical framework for locating perfusion defects. Its value in identifying various diseased vessels was confirmed by a comparison of the tomographic findings with the angiographic findings in five individual cases.

Thallium-201 perfusion scintigraphy is now a well established technique for imaging the myocardium.[1,2] Its use is advocated in patients with atypical chest pain for diagnosing possible coronary artery disease.[3–5] The interpretation of thallium-201 images is, however, difficult and requires considerable experience to obtain reproducible results, and considerable interobserver variation has been found.[6,7] This situation led to the development of semiquantitative methods for analysing thallium scans[8–10] and may also be responsible for the poor correlation reported by some authors between the site of disease as determined from scans and that of coronary artery stenoses found by angiography.[11,12] Further studies to improve the accuracy of localisation of disease have compared semiquantitative results with empirically derived normal ranges.[13,14] The scintigraphic appearances of specific coronary artery stenoses have been studied extensively and many patterns identified[14]; with experienced observers these methods gave accurate localisation of disease.[15]

Rotating gammacamera tomography has been shown to be the best imaging technique generally available for thallium-201 scintigraphy of the myocardium.[16–18] The use of tomograms is an important step forward towards the location of arterial disease, with relatively good results being obtained in one or two vessel disease.[19] We have previously established a tomographic technique using both exercise and resting scans and have shown good agreement with exercise electrocardiographic results, good agreement between qualitative and semiquantitative assessment, and an absence of equivocal results.[20] We now report a method for relating scintigraphic findings to the coronary arteries. We believe that this will be a useful step forward towards the use of thallium-201 for localising coronary artery disease.

Requests for reprints to Dr E D Williams, Regional Medical Physics Department, Sunderland District General Hospital, Kayll Road, Sunderland, Tyne and Wear SR4 7TP.

Accepted for publication 27 June 1985

Patients and methods

Patients underwent a symptom limited exercise electrocardiographic test in accordance with a Bruce protocol. Once peak stress had been achieved in the

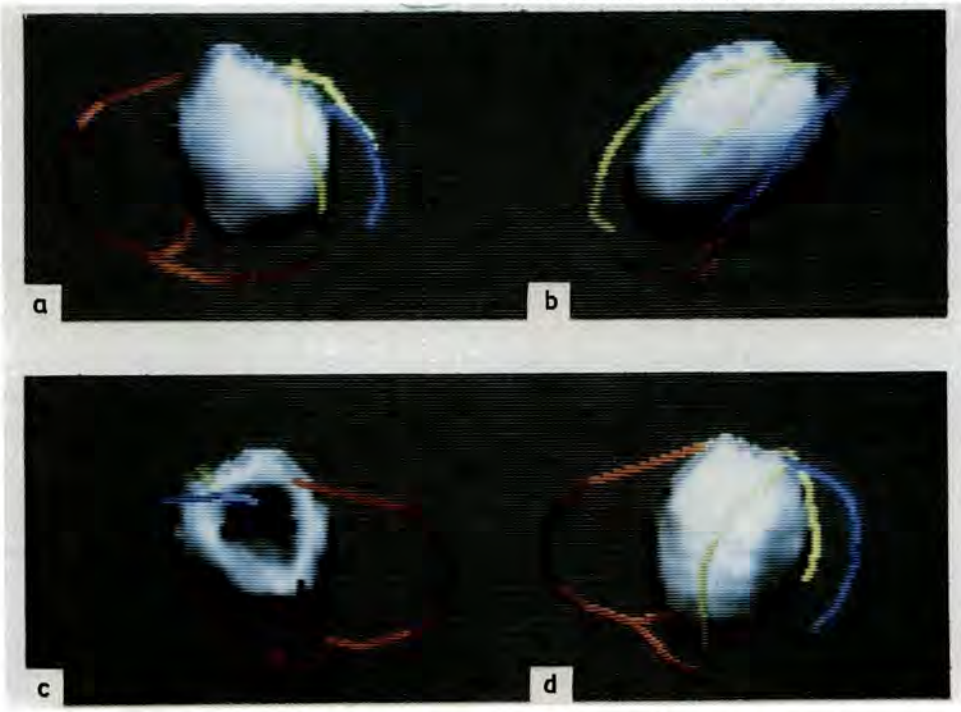


Fig. 1 *Three dimensional shaded image representations of thallium-201 uptake in a normal myocardium (case 1) viewed from (a) the anterior, (b) the left lateral, (c) the right posterior oblique, and (d) the left anterior oblique positions with the coronary arteries and their major branches superimposed (right coronary artery in red, left descending in yellow, circumflex in blue).*

exercise test 80 MBq thallium-201 thallous chloride was injected and exercise continued for a further minute. Patients were then quickly transferred for imaging, which began within five minutes of the end of exercise. An IGE 400 AT gammacamera with a DEC Gamma-11 computer system was used for tomographic imaging.[21] Data were acquired for 40 s at each of 32 equally spaced positions over a 180° arc from left posterior oblique to right anterior oblique. A low energy general purpose collimator was used and the pulse height analyser window set at 55–90 keV. Tomography was repeated after four hours rest to allow the thallium-201 to redistribute to the resting perfusion pattern. The image data were filtered with a Metz count dependent prefilter[22] and a magnified reconstruction performed by back projection (NucMed software package, Analogic Corp, Wakefield, Massachusetts, USA). The pixel size in reconstructed images was equivalent to 3 mm. This image processing was considered subjectively to give the best quality tomographic images.

Transverse section data were transferred to a Link Dyanne computer system via magnetic tape and were used to construct a three dimensional representation of the myocardial uptake. A count threshold level was selected interactively, and a computer model of the resulting myocardial distribution created by stacking adjacent transverse sections. Only pixels with count values greater than the threshold level were considered to be part of the model. From a set of up to 64 transverse sections (3 mm nominal thickness) each of 64 × 64 pixels (3 × 3 mm), the myocardium was modelled by an array of 64 × 64 × 64 bits, where each bit was set only if the corresponding pixel exceeded the threshold level. The time required to create this binary model from a set of tomograms was 1·5 s per slice or typically 45 s per study. Note that the model needed to be created once only and was then stored on disk. This model was displayed as if illuminated by a light source located above the observer's head and angled down on the model. For all visible points on the model a grey scale value was calculated from the local surface orientation. The resulting shaded surface image presents information about the shape of three dimensional objects in a two dimensional form.[23] By rotating and tilting the model the myocardial distribution may be viewed from any angle, thus allowing the observer to perceive directly the three dimensional distribution of perfused myocardium. The

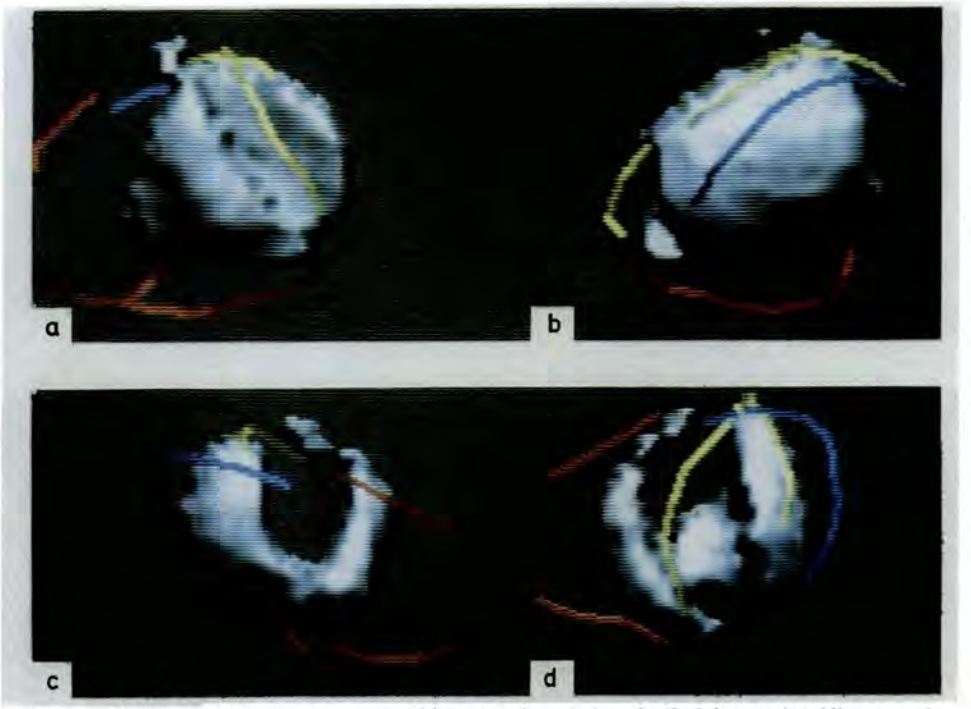


Fig. 2 *Thallium-201 images at rest viewed from (a) the anterior, (b) the left posterior oblique (c), the right posterior oblique, and (d) the left anterior oblique positions showing a large perfusion defect in the anterior wall, coincident with the left anterior descending artery and therefore indicating the site of a previous myocardial infarction (case 2). Colour coding of coronary arteries as in Fig. 1.*

time taken to rotate the model and to display a shaded surface image depended on myocardial size and ranged from 4 to 6 s to rotate or tilt the model and from 13 to 24 s to calculate and display the shaded surface image. Although faster execution would be desirable, these times are sufficiently short to allow the program to be used interactively.

To provide anatomical reference points, the locations of the coronary arteries and their major branches were added to the display. A cast of coronary arteries was obtained from a normal necropsy specimen, and the major arteries were digitised in three dimensions using a reflex metrograph (Ross Instruments). A simple fitting procedure was used to scale these arteries on to the computer model of myocardial uptake. A hidden surface algorithm ensured that only the visible portions of the coronary arteries were displayed when the myocardium was observed from different aspects.

Several methods were used for viewing the result. Arteries were distinguished one from another and from myocardium by using a different colour for each artery on a colour video monitor and superimposing the arteries on a grey shaded image of myocardial uptake. The location of arteries could be perceived by presenting multiple views of the image model from several different aspects simultaneously (Figs. 1–5). An alternative method was a rapid sequential display of views with small angular displacements so that motion parallax improved the three dimensional perception.[24] Finally, stereo pair images can be constructed by creating two images with a small angular displacement (10°) and viewing these as transparencies with an appropriate viewer. Only the first method was considered suitable for printed publication.

Results

CASE REPORTS

Case 1 (normal)—A man aged 46 was admitted to hospital after a "collapse." Initially, he manifested sinus tachycardia and later sinus bradycardia. An exercise electrocardiogram performed to rule out ischaemic heart disease was equivocal and was therefore repeated and combined with a thallium scan. During the exercise test he exercised well into stage VII with no chest pain and no electrocardiographic changes. Coronary angiography was therefore not sought. The thallium tomograms showed no evi-

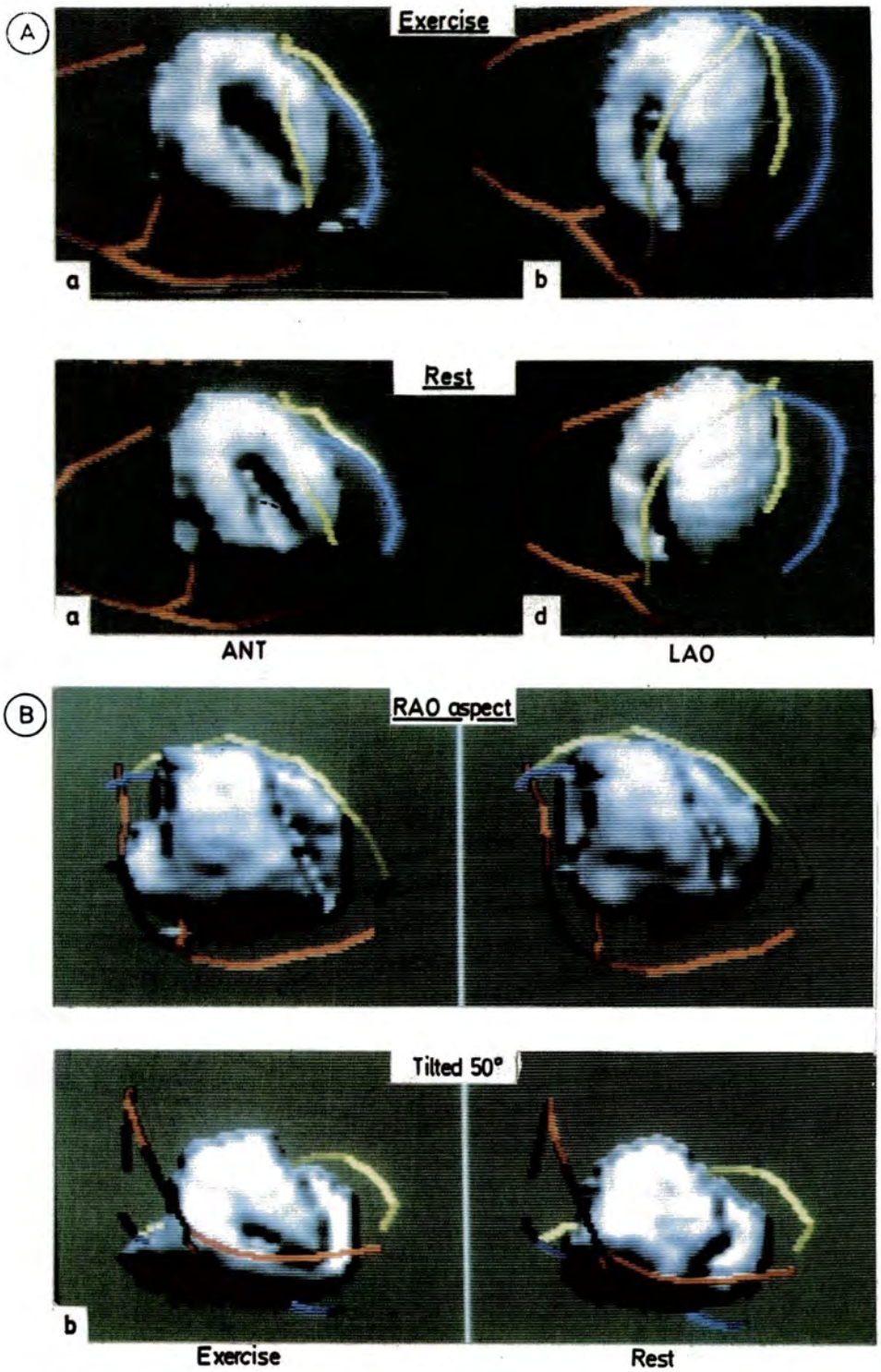


Fig. 3 *Thallium-201 images viewed from the anterior (ANT) and left anterior oblique (LAO) positions showing (A) apical defect both on exercise (a and b) and at rest (c and d) corresponding to distal left anterior descending artery. (B) The right anterior oblique (RAO) view shows a defect on exercise in the right coronary artery territory. Colour coding of coronary arteries as in Fig. 1.*

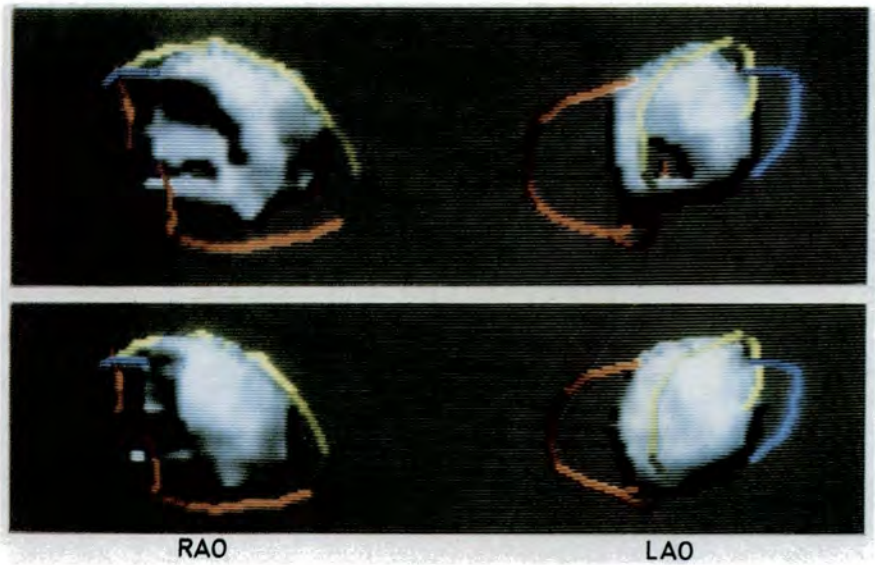


Fig. 4 *Thallium-201 images in the right anterior oblique (RAO) and left anterior oblique (LAO) positions on exercise (top) and at rest (bottom) (case 4). Two defects are seen on exercise in the septal and anterior regions, both of which are perfused at rest. The defects do not correspond directly to the major arteries, which indicates that some of their branches or distal portions are diseased. Colour coding of coronary arteries as in Fig. 1.*

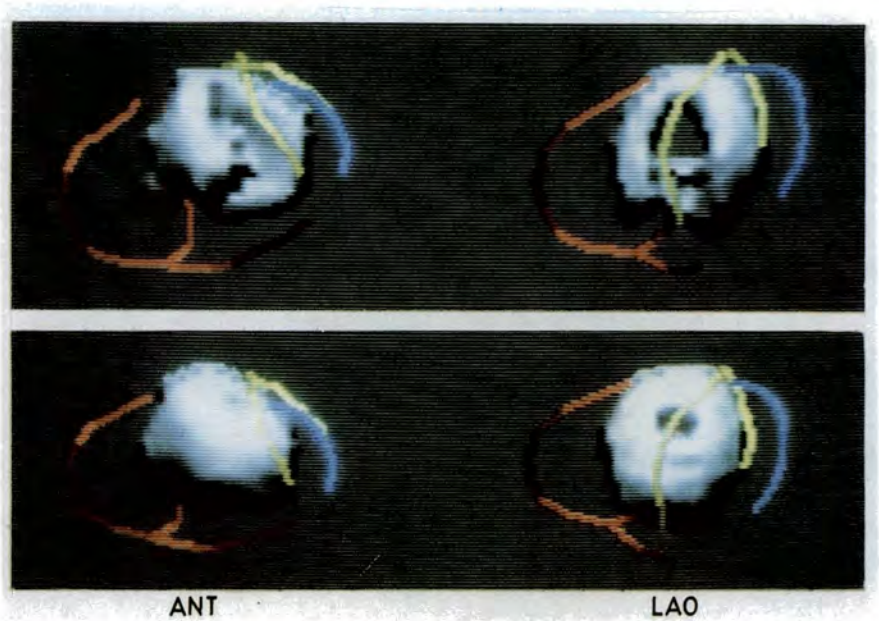


Fig. 5 *Thallium-201 images on exercise (top) and at rest (bottom) viewed from the anterior (ANT) and left anterior oblique (LAO) positions in a patient (case 5) with ischaemia. The large defect on exercise extends along the line of the left anterior descending artery and continues right under the left ventricle. There is a second small defect on the inferior surface indicating right coronary artery disease also. Colour coding of coronary arteries as in Fig. 1.*

dence of ischaemia, and three dimensional reconstruction showed no defects on exercise or resting images. Thus each of the investigations confirmed the absence of ischaemic heart disease (Fig. 1).

Case 2 (infarct)—A 38 year old man with a previous history of anteroseptal myocardial infarction presented again four months later with chest pain that could have been anginal in nature and was treated with beta blockers and nitrates. An exercise electrocardiogram was terminated at stage IV because of chest pain, but no changes were evident in the tracing. Left ventricular cineangiography showed the anteroapical region to be akinetic, and on coronary arteriography the left anterior descending artery was completely occluded just beyond the large first diagonal. The thallium scan showed a large apical defect on the exercise tomograms but a slight uptake at the apex on the resting images. This indicated an apical myocardial infarction, with a possible ischaemic area. The three dimensional display (Fig. 2) showed a large left ventricle with a major defect in the superior/apical surface, almost exactly coincident with the left anterior descending artery after the first diagonal branch. Rest and exercise images were similar, except for possible defect enlargement seen on the exercise image. The findings of the thallium scan were therefore in good agreement with the coronary angiography report.

Case 3 (infarct and coronary artery disease)—A 36 year old man had had extensive anterior wall myocardial infarction one year previously. His continuing chest pain was treated with a beta blocker (atenolol), a calcium antagonist (nifedipine), and nitrates. An exercise electrocardiogram was terminated at stage II because of chest pain and dizziness. There was ST elevation and T wave inversion in the anteroseptal leads. Left ventriculography showed apical akinesia and arteriography showed a tight stenosis in the middle third of the left anterior descending artery and its second diagonal. There was also fairly diffuse disease affecting the marginal and posterolateral branches of the circumflex and the right coronary artery. A thallium scan showed decreased uptake on both rest and exercise images in the apicoseptal region, indicating myocardial infarction there. Also an area of decreased uptake was seen in the inferoseptal region on exercise, which had normal uptake at rest, indicating ischaemia. The three dimensional display confirmed these findings by showing a defect in the inferoapical region that extended to the superior surface (Fig. 3) and was reasonably coincident with the distal portion of the left anterior descending artery and an inferoseptal defect on exercise in the territory of the right coronary artery.

Case 4 (coronary artery disease)—A man aged 50 years had mild hypertension and a four month history of angina of effort, which was gradually getting worse and was not controlled by treatment with atenolol and chlorthalidone, nitrates, and nifedipine. The exercise electrocardiogram showed ST segment depression in V5 leads at stage III. The coronary angiogram indicated a blockage in the right coronary artery beyond its right ventricular branch, but the distal part of the vessel was filled retrogradely. The circumflex artery was the dominant vessel, but this had regions of severe stenosis distal to the obtuse marginal branch and proximal to the posterior descending artery. The mid portion of the obtuse marginal branch was also diseased. The left anterior descending artery had regions of stenosis both proximal and distal to the first septal branch and proximal to the large second diagonal. The second diagonal branch showed moderate stenosis in the mid portion and the left anterior descending artery itself was blocked distal to this branch, although there was some retrograde filling. The thallium scan showed defects on exercise in the septum and inferolateral regions which had normal uptake on the later (resting) images. The three dimensional representation indicated two defects on exercise in the anterior and septal walls (Fig. 4). These could not be directly related to the main coronary arteries but appeared to be in the territories of the left anterior descending or right coronary artery or both. Neither defect was seen at rest. The defects seen on the thallium scan did not correspond directly to the main coronary arteries, and this is consistent with the fairly extensive disease found at angiography to involve the distal portions and branches of the main coronary arteries. Thus the collateral circulation shown on the coronary arteriogram to supplement that of the distal right coronary and left anterior descending arteries would appear to be adequate at rest but insufficient to meet the requirements of the stressed myocardium.

Case 5 (coronary artery disease)—A 52 year old man had had angina of effort and sometimes at rest for about one year: he had had two previous hospital admissions for chest pain. The symptoms were treated with atenolol and chlorthalidone, nitrates, and nifedipine. An exercise electrocardiogram showed ST segment depression in the inferolateral leads at stage III. He is awaiting coronary angiography. Thallium scans showed an apical defect on exercise, which had normal uptake on the later (resting) image. The three dimensional display also showed a large defect on the exercise image, extending along the line of the left anterior descending artery from approximately the second diagonal branch. This defect persisted as a very small abnor-

mality on the resting image. The defect extended right under the left ventricle, with a second small defect in the inferior surface. This suggests disease of the right coronary artery also. Although the findings have not yet been confirmed by coronary angiography, the three dimensional reconstructions unequivocally indicated the site of disease to be in the left anterior descending artery.

Discussion

Presentation has long dogged the acceptance of nuclear medicine images by doctors, both because images commonly lack anatomical reference points and because a functional rather than an anatomical image is presented. Thus recognition of image features can be difficult for the uninitiated. In nuclear cardiology further difficulties have been presented by the introduction of parametric imaging,[25] which is even further removed from common ideas of anatomy.

The method described in this paper is an attempt to reverse this process by presenting images of functioning myocardium in which the spatial relations of the different parts of myocardium, and their relative function, are clearly discernible and in which the appropriate anatomical reference points are provided by superimposing the coronary arteries. We hope that by this means we shall be able to establish more clearly the relation between myocardial perfusion defects, at rest and during exercise, and the site of coronary artery disease.

Several authors have described shaded surface techniques for displaying three dimensional information.[26-28] Applications have included surgical planning,[29] radiotherapy planning,[30] and an evaluation of complex musculoskeletal anatomy.[31] There have been only a few reports of applications using radionuclide tomography.[2332] Our technique differs from previous methods in two ways. Firstly, for the chosen viewing direction a two dimensional image array was calculated, each pixel of which contained the value of the distance from a plane of each visible point on the three dimensional model. The shading algorithm was then applied to this array to create the final image. The algorithm includes some surface smoothing and a two dimensional local polynomial fit to estimate the orientation of the surface normal at all points in the image, which is necessary before illumination by the imaginary light source. Despite the relatively coarse sampling of radionuclide tomograms compared with other modalities this procedure can produce smooth surface images rather than the irregular surfaces shown by simpler techniques. Because the shading algorithm operates on two dimensional arrays it can incorporate this level of complexity without also requiring a long execution time. Secondly, we have added anatomical information to the display in the form of superimposed coronary arteries. This not only provides a familiar anatomical framework for viewing the myocardium and may thus aid correct identification of the location of arterial disease but also enhances the perception of depth.

Two assumptions have to be made in order to create our three dimensional images. Firstly, it is necessary to assume that the distribution of thallium-201 contains definite areas of uptake that may be enclosed by a surface, corresponding to an isocount contour. In practice the uptake often varies continuously from lower to higher values. A simple thresholding technique can distinguish the areas of appreciably reduced uptake but will probably fail to detect small areas of only slightly reduced uptake. Secondly, the spatial distribution of the coronary arteries corresponds to that of one normal subject. The arteries displayed on each image have been scaled and rotated to match the individual myocardial model, but they should be interpreted as only indicating the probable site of the actual coronary arteries.

We suspect that the accuracy achievable by perfusion scintigraphy in detecting one or two vessel disease may be better than has previously been reported.[19] Localisation of coronary artery disease is important in order to determine whether patients are suitable for coronary artery bypass graft surgery.[33] In addition, the method should prove useful for the non-invasive follow up of patients after surgery who produce new symptoms and for assessing the success of coronary angioplasty.

We thank R Blair, Y Ennew, C Taylor, K Turner, and J Russell for their technical assistance, and cardiologists at the Regional Cardiothoracic Centre, Freeman Hospital, Newcastle upon Tyne, for the angiography reports.

References

1 Strauss HW, Pitt B. Evaluation of cardiac function and structure with radioactive tracer techniques. *Circulation* 1978; **57**: 645–54.
2 Wainwright RJ, Maisey MN. Cardiac imaging, Part 1, Myocardial perfusion scintigraphy. *Hospital Update* 1978; **22**: 623–38.
3 Iskandrian AS, Wasserman LA, Anderson GS, Hakki H, Segal BL, Kane S. Merits of stress thallium-201 myocardial perfusion imaging in patients with inconclusive exercise electrocardiograms: correlation with coronary arteriograms. *Am J Cardiol* 1980; **46**: 553–8.
4 Fletcher JW, Walker KE, Witztum KF, *et al*. Diagnosis of coronary artery disease with 201-Tl. *Radiology*

1978; **128:** 423–7.
5 McCarthy DM, Blood DK, Sciacca RR, Cannon PJ. Single dose myocardial perfusion imaging with thallium-201: application in patients with non-diagnostic electrocardiographic stress tests. *Am J Cardiol* 1979; **43:** 899–906.
6 Trobaugh GB, Wackers FJT, Sokole EB, DeRouen TA, Ritchie JL, Hamilton GW. Thallium-201 myocardial imaging: an interinstitutional study of observer variability. *J Nucl Med* 1978; **19:** 359–63.
7 Okada RD, Boucher CA, Kirshenbaum HK, *et al.* Improved diagnostic accuracy of thallium-201 stress test using multiple observers and criteria derived from interobserver analysis of variance. *Am J Cardiol* 1980; **46:** 619–24.
8 Francisco DA, Collins SM, Go RT, Erhardt JC, Van Kirk OC, Marcus ML. Tomographic thallium-201 myocardial perfusion scintigrams after maximal coronary artery vasodilation with intravenous dipyridamole. *Circulation* 1982; **66:** 370–9.
9 Buell U, Kleinhaus E, Seiderer M, Strauer BE. Quantitative assessment of thallium-201 images. *Cardiovasc Radiol* 1979; **2:** 183–93.
10 Berger BC, Watson DD, Taylor GJ, *et al.* Quantitative thallium-201 exercise scintigraphy for detection of coronary artery disease. *J Nucl Med* 1981; **22:** 585–93.
11 McKillop JH, Murray RG, Turner JG, Bessent RG, Lorimer AR, Greig WR. Can the extent of coronary artery disease be predicted from thallium-201 myocardial images? *J Nucl Med* 1979; **20:** 715–9.
12 Rigo P, Bailey IK, Griffith LSC, *et al.* Value and limitations of segmental analysis of stress thallium myocardial imaging for localization of coronary artery disease. *Circulation* 1980; **16:** 973–81.
13 Burow RD, Pond M, Schafer AW, Becker L. "Circumferential profiles". A new method for computer analysis of thallium-201 myocardial perfusion images. *J Nucl Med* 1979; **20:** 771–7.
14 Wainwright RJ. Scintigraphic anatomy of coronary artery disease in digital thallium-201 myocardial images. *Br Heart J* 1981; **46:** 465–77.
15 Wainwright RJ, Maisey MN, Sowton E. Segmental quantitative analysis of digital thallium-201 myocardial scintigrams in diagnosis of coronary artery disease: Comparison with rest and exercise electrocardiography and coronary arteriography. *Br Heart J* 1981; **46:** 478–85.
16 Tamaki N, Mukai T, Ishii Y, *et al.* Clinical evaluation of Thallium-201 emission myocardial tomography using a rotating gamma camera: comparison with seven-pinhole tomography. *J Nucl Med* 1981; **22:** 849–55.
17 Ritchie JL, Larsson S, Israelson A, *et al.* Single photon tomographic imaging of a standard heart phantom with 201Tl: a gamma camera based system. *Eur J Nucl Med* 1982; **7:** 254–9.
18 Myers MJ, Sokole EB, de Bakker J. A comparison of rotating slant hole collimator and rotating camera for single photon emission tomography of the heart. *Phys Med Biol* 1983; **28:** 581–8.
19 Tamaki N, Yonekura Y, Mukai T, *et al.* Segmental analysis of stress thallium myocardial emission tomography for localization of coronary artery disease. *Eur J Nucl Med* 1984; **9:** 99–105.
20 Laird EE, Rajathurai A, Williams ED, Mittra B, Rankin D. Quantitative analysis of rotating gamma camera thallium-201 scintigrams of myocardium. *Nuclear Medicine Communications* 1984; **5:** 577–86.
21 Larsson SA. Gamma camera emission tomography. *Acta Radiologica* 1980; suppl 363.
22 King MA, Schwinger RB, Doherty PW, Penney BC. Two-dimensional filtering of SPECT images using the Metz and Wiener filters. *J Nucl Med* 1984; **25:** 1234–40.
23 Gibson CJ. A new method for the three dimensional display of tomographic images. *Phys Med Biol* 1983; **28:** 1153–7.
24 Williams ED. Three-dimensional radionuclide images [Letter]. *J Nucl Med* 1981; **22:** 193.
25 Houston AS, Elliott AT, Stone DL. Factorial phase imaging: a new concept in the analysis of first-pass cardiac studies. *Phys Med Biol* 1982; **27:** 1269–77.
26 Herman GT, Liu HK. Three dimensional display of human organs from computed tomograms. *Computer Graphics and Image Processing* 1979; **9:** 1–21.
27 Cook LT, Dwyer SJ, Batnitzky S, Lee KR. A three dimensional display system for diagnostic imaging applications. *IEEE Computer Graphics and Applications* 1983; **3:** 13–9.
28 Udupa JK. Display of 3-D information in discrete 3-D scenes produced by computerised tomography. *Proceedings of the IEEE* 1983; **71:** 420–31.
29 Vannier MW, Marsh JL, Warren JO. Three dimensional CT reconstruction images for craniofacial surgical planning and evaluation. *Radiology* 1984; **150:** 179–84.
30 Bloch P, Udupa JK. Application of computerised tomography to radiation therapy and surgical planning. *Proceedings of the IEEE* 1983; **71:** 351–5.
31 Totty WG, Vannier MW. Complex musculo-skeletal anatomy: analysis using three dimensional surface reconstruction. *Radiology* 1984; **150:** 173–7.
32 Minato K, Mukai T, Yamamoto K, *et al.* Three dimensional display of single photon ECT imaging. *Proceedings of the 3rd World Congress of Nuclear Medicine and Biology. 1982.* Oxford: Pergamon Press, 1983: 3651.
33 Hampton JR. Coronary artery bypass grafting for the reduction of mortality: an analysis of the trials. *Br Med J* 1984; **289:** 1166–70.

Nuclear Medicine Communications 7, 475–487 (1986)

Interactive display of three-dimensional radionuclide distributions

C.J. GIBSON

Regional Medical Physics Department, Dryburn Hospital, Durham, DH1 5TW, UK

Received 4 December 1985

Summary

Tomographic images of three-dimensional radionuclide distributions are usually presented as a set of parallel planar slices. A technique has been developed to enable the observer to perceive directly the three-dimensional morphology of the activity distribution by displaying iso-count surfaces. Computer graphics techniques have been used to present shaded surface images using a conventional nuclear medicine computer system. Because of the relatively coarse spatial sampling of radionuclide tomograms, a smooth shading algorithm based on a local polynomial fit procedure was developed. By using a simple solid model approach for data storage the program executes interactively, allowing the observer to view the reconstructed activity distribution from any aspect. The technique has been successfully applied to radionuclide tomograms of several organs, and promises to be particularly useful for the display or cardiac blood pool data.

Introduction

Several medical imaging modalities produce tomographic images, in which the content of a pixel represents the value of the imaging parameter at the corresponding point in space. If a series of images are obtained from adjacent slices then a complete sampling of the three-dimensional distribution of the parameter has been made. Several techniques have been developed for displaying such data [1]. Each slice can be displayed as a two-dimensional image on a conventional raster graphics screen. The perception of three-dimensional information may then be enhanced by observing successive slices as a rapid movie sequence [2]. When the plane of the sections does not correspond to natural anatomical features then the data can be re-formatted into a set of planes at a different angle. This process has been used to

0143-3636/86 $03.00+.12

create coronal and sagittal planes [3] oblique planes [4], and curved coronal sections which follow the natural curvature of the spine [5].

One object of these techniques is to assist the observer in the difficult task of evaluating three-dimensional morphology from a set of two-dimensional sections. However, the technique of surface detection and display can present three-dimensional information in a readily perceivable form using conventional display systems [6]. All surface display techniques are based on the principle that three-dimensional shape information is conveyed most effectively by displaying surface morphology. The process of inferring an object's shape from its surface appearance corresponds to the way in which objects are perceived in the natural world.

In a set of transverse sections, surfaces are defined as boundaries between regions with differing values of imaging parameter. The volume enclosed within such a surface represents the organ of interest. The method used to display the surface depends largely on the format chosen to store surface information. The surface may be represented by a stack of outlines, representing the intersection of the organ surface with the planes of successive sections. This technique has been applied to histological sections [7, 8] and to organ outlines obtained from CT scans [9, 10]. Additional surface information may be displayed by 'titling' the set of outlines. The tiles are created using a minimum surface area criterion [11] or a heuristic algorithm designed to achieve a smooth outline [12]. The tiles are displayed using a smooth shading algorithm which indicates surface orientation and disguises the discrete nature of the surface [13]. This technique has been applied to the display of cardiac muscle cells [14], and to organs obtained from CT scans [15], and ECAT scans [16].

An alternative scheme for representing organ surfaces arises naturally from the discrete sampling of computed tomograms. The stack of cross-sectional images may be thought of as a three-dimensional array, in which each element or 'voxel' is representative of a small volume of tissue. If all voxels can be classified by some operator into a set belonging to the object and a set which do not, then the object distribution is determined uniquely by the set of object voxels. Such a representation has been described as a 'cuberille' approach [17]. This technique has the advantage that complex topologies can be catered for, and the simple nature of the voxel representation enables a fast hidden surface algorithm to be used for cubic voxels. Cuberille techniques have been applied to both CT images [18, 19] and to NMR data [20].

It is the purpose of this paper to describe algorithms for applying the cuberille technique to coarsely sampled tomographic data. Such data are obtained from radionuclide tomograms, where the reconstructed sections are often only available as 64×64 images. However, the algorithms described in this paper may be applied to data from any tomographic imaging modality.

Methods

Data storage

Radionuclide tomograms are usually obtained as parallel transverse slices at 64×64 resolution with 16 bits of storage per pixel. The slice spacing is almost always an integral multiple of the pixel size. Using linear interpolation it is therefore possible to create a set of sections which have a slice spacing equal to the pixel size. In the general case the input data consists of 64 slices at 64×64 resolution, forming a cubic array requiring 512 Kbytes of storage. Before a surface can be displayed the object distribution must be extracted from this set of data. Generally the object of interest corresponds to those array elements (volume elements or 'voxels') which have a high concentration of radionuclide. A simple thresholding algorithm may therefore be used to separate out object voxels. Previous implementations of the cuberille approach have stored the set of object voxels either as a list of the exterior faces [6], or as a series of directed contours [21].

We have chosen a scheme for storing the object data which represents the object as a solid model rather than a set of surfaces [22]. A binary array of 64×64×64 bits can be used to represent the thresholded object. Each bit corresponds to a voxel in the original data array, and is set only if that voxel contains a value greater than the selected threshold. The data storage required for the binary object array is only 32 Kbytes, and this array may therefore be stored in the core memory of a laboratory minicomputer.

In practice the selection of a threshold level is necessarily a subjective process. Simple thresholding of noisy data leads to the production of isolated object voxels and uneven surfaces. Smoothing prior to thresholding can produce better surface continuity at the expense of poorer resolution. For radionuclide tomograms at last one smooth within and between slices was found to be necessary. Choice of threshold level was facilitated by using an interactive program which displayed successive slices at varying thresholds until the object was delineated satisfactorily. The corresponding binary object array was then created and stored as a disk file for subsequent analysis.

Surface detection

Many algorithms have been developed to solve the general problem of hidden surface removal [23]. A binary object representation leads to an extremely rapid algorithm for visible surface detection. Suppose that the viewing direction is constrained to be parallel to one of the major axes of the array. A 64×64 element image array may then be constructed perpendicular to the viewing direction (Fig. 1). Each pixel in the image array is located opposite a column of 64 object voxels. For each image pixel the visible voxel is the nearest object voxel in the corresponding column. The output from this simple algorithm is therefore a two-dimensional array containing the 'depth' of the visible voxels. Since determining the nearest object voxel corresponds to finding the first 'set bit' in successive array elements, the algorithm can be readily implemented in assembly language, and executes very rapidly.

Different viewing directions

As described above the visible surface algorithm is only capable of determining the object surfaces as seen from one of six aspects (corresponding to the major axes of the cubic array). However, the object data may be rotated or tilted in the binary array prior to the determination of the visible voxels. Because the object data is sampled so coarsely (i.e. only 1 part in 64 along

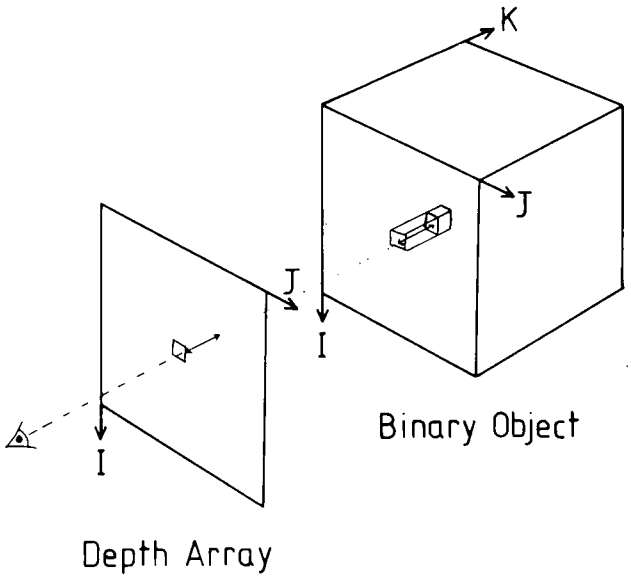


Fig. 1. Visible voxel information is stored as depth values in a two-dimensional array perpendicular to the line of sight of the observer.

each axis) some 'step' discontinuities may be created when flat surfaces are rotated through small angles. However, this undesirable effect is not significant when displaying the irregular surfaces of natural objects. Once rotated or tilted the visible surface algorithm may be used to rapidly obtain the object surfaces seen from the new aspect.

Shading

The simplest form of shading is depth shading, in which the intensity of each visible voxel is inversely related to the distance from the observer. This technique is rapid and has been used extensively for displaying cuberille models [19, 24].

However, depth shaded images provide relatively few clues to help the observer perceive surface orientation. A variety of more sophisticated shading algorithms have been developed for cuberille models [25]. We have implemented an image space shading technique [26], with the added capability of producing smooth shaded images from coarsely sampled data.

The depth array obtained by the visible surface algorithm is used as input to the shading algorithm. A simple form of surface shading based on Lamberts Law, assigns a shade to a surface using

$$I = I_0 + I_1(pl + qm + rn) \tag{1}$$

where I is the shading intensity at a point on the surface, I_0 is the intensity of ambient isotropic illumination, I_1 is the intensity of uniform directional illumination, (p, q, r) are the direction cosines of the directional light vector and $(l, m, n,)$ are the direction cosines of the surface normal. In this expression I_0, I_1, p, q and r are chosen to give the desired illumination and a suitable value for I. However, it is necessary to determine the direction cosines of the surface (l, m, n) at every visible point on the object. If the gradients of the surface along two

orthogonal directions are known then the unnormalized direction cosines of the surface normal may be estimated by

$$\left.\begin{aligned} l' &= -G_x \\ m' &= -G_y \\ n' &= -1 \end{aligned}\right\} \tag{3}$$

where G_x is the surface gradient parallel to the x-axis and G_y is the surface gradient parallel to the y-axis. Gradients of the digitally sampled surface are obtained by numerical differentiation along the two axes of the depth array. Two problems arise during this process. Firstly, the noise amplification properties of the numerical differentiation process mean that some degree of smoothing may be required. Secondly, and conversely, it is important to preserve those larger discontinuities between adjacent pixels which correspond to the edges of visible object(s). A variable shape local polynomial fitting procedure may be used to satisfy both requirements. The algorithm is essentially one-dimensional and is applied independently

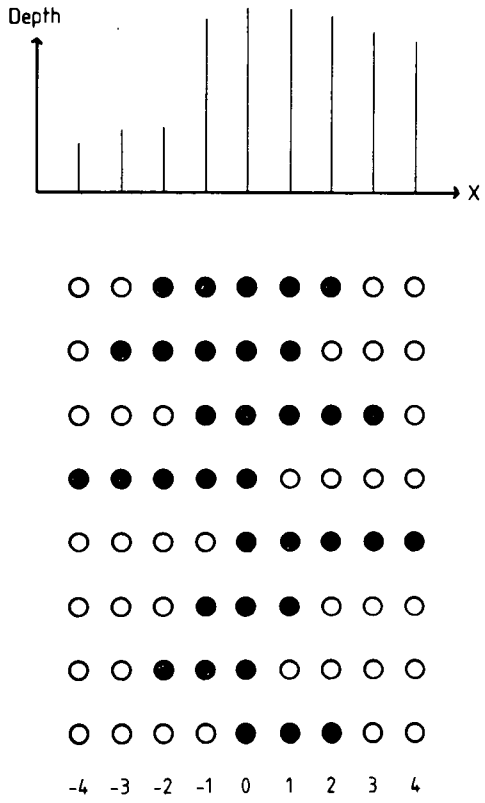


Fig. 2. A profile through the depth array showing a typical surface discontinuity. The shading algorithm attempts to fit a quadratic or a linear function, to the points indicated, starting with the central block of five points. The gradient of the fitted function is used to estimate the surface normal at the central point.

along the x and y directions. For each pixel in the depth array consider a block of nine linearly adjacent pixels centred on the pixel of interest. If the depths of the visible voxels at each location differ from the central depth by less than some chosen value (currently six voxels) then they may be considered as part of a locally continuous surface. The algorithm then attempts to fit a quadratic function to any five adjacent points out of the nine, which form a connected locally continuous surface. If the surface is highly complex then no such set of points may exist. In this case the algorithm considers the central block of five pixels only and attempts to fit a linear function to any three adjacent points satisfying the connectivity criterion. If this proves impossible then the surface is clearly so complex that no simple surface normal can be defined and depth shading only is used. When a quadratic or linear fit is obtained the gradient may be readily calculated by multiplying the observed depth values by the relevant weighting factors. Fig. 2 shows the pixels considered by the algorithm, and the sequence in which successive fits are attempted.

Interpolation and display

From an initial binary object array containing $64\times64\times64$ voxels the visible surface and shading algorithms produce a 64×64 array containing shade values in the range

$$I_0 \leq I \leq I_1+I_0 \tag{3}$$

Typical values of I_0 and I_1 are 50 and 200, giving values of I which can be stored in successive bytes of a standard raster type display memory. When a larger image is required the shade array may be interpolated to 128×128 or 256×256 pixels using a modified linear interpolation algorithm. Simple linear interpolation will blur the edges of objects. This can be avoided using an algorithm which only interpolates between shade values corresponding to visible object elements. At the edges of the object, adjacent shade elements correspond to values determined by the object orientation and values representing background (i.e. no object visible). The interpolated values for all such edge elements are set equal to the value chosen to represent the background usually zero.

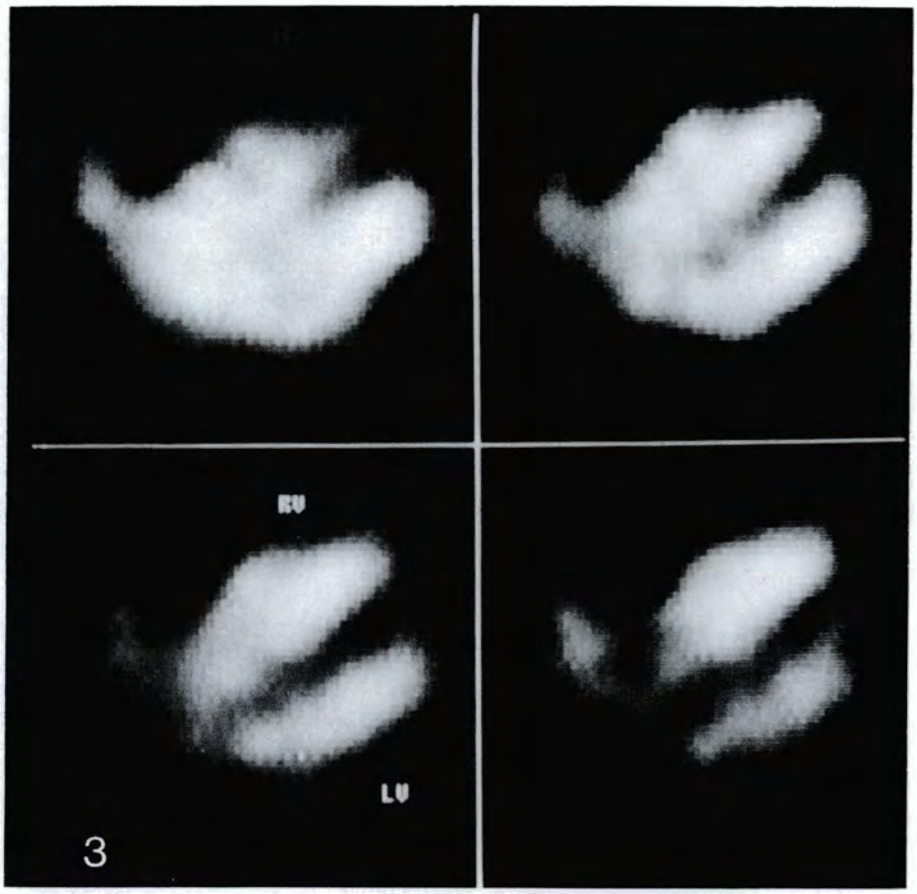
Results

Tomograms

Emission computed axial tomographic images (ECAT) were obtained using a standard rotating gamma camera. In all cases the administered activity was less than or equal to the ARSAC recommended maximum. Transverse tomographic images were reconstructed from either 64 images obtained at equiangular intervals over 360°, or from 32 images obtained over 180° for cardiac tomography. Slices were reconstructed using a conventional nuclear medicine processor which is based on a Data General NOVA 4x. Subsequent image processing was performed on the same computer using software written in Fortran and Assembler. All timing values quoted were real time, not CPU time.

Fig. 3. Four transverse sections through the cardiac blood pool displayed in the conventional tomographic orientation.

Fig. 4. The same slices as shown in Fig. 3 after smoothing and thresholding. The binary object array is created by stacking the complete set of thresholded images.


RV
LV
3


4

Thresholding

Fig. 3 shows four transverse slices through the cardiac blood pool obtained using ungated tomographic imaging. By smoothing and thresholding a binary object was created which corresponded to the distribution of activity within the cardiac chambers. Fig. 4 shows four smoothed and thresholded slices. Creation of the binary object from a set of slices required approximately 1.5 s per slice. Note that this stage was performed only once as a pre-processing technique, and the binary object stored in a disk file.

Visible voxels

Determining the visible voxel array was performed extremely rapidly using the algorithm described above. For typical objects the whole binary array was processed in 0.2 s. If depth shading only was required then an image could be produced at this stage. Figs. 5a and b show the depth shaded image of the cardiac blood pool obtained in the anterior and left lateral projections.

Shading

The surface orientation shading algorithm was applied to the depth image, and the corresponding anterior and left lateral views are shown in Figs. 5c and d; note the additional realism and contrast when compared to the depth shaded images. Execution times for this algorithm depended on the complexity and size of the object, and ranged from 19 to 30 s.

Object rotation

To observe the object from different aspects the data in the binary object array were rotated or tilted prior to determining the visible voxels. Fig. 6 shows images of the same cardiac blood pool from several aspects. Execution times for rotation or tilting were also dependent on the size of the object, and ranged from 4.2 to 8.7 s.

Object modification

The same technique has been successfully applied to radionuclide tomograms of other organs. One advantage of the solid model approach described here is the ability to reveal interior detail by simple modification of the binary object. Fig. 7 shows several views of a radionuclide brain scan seen from the left lateral and vertex aspects, with part of the binary object 'sliced off' to reveal the normal empty brain cavity. This slicing technique was performed by setting all voxels nearer to the observer than some chosen depth equal to zero. Typically the execution time was 0.1 s for processing the whole object. Fig. 8 shows an abnormal liver seen from the anterior aspect. As increasing amounts of the object are removed the full extent of the abnormality is revealed.

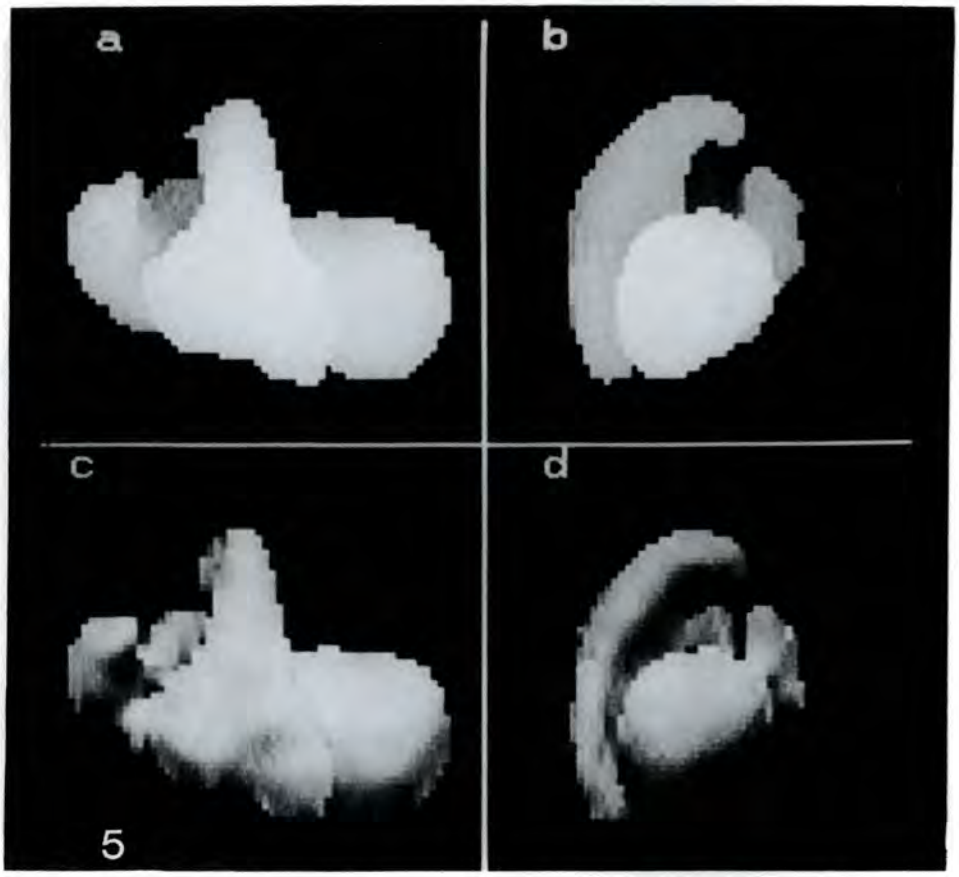


Fig. 5. (a) Depth shaded image of the cardiac blood pool seen from the anterior aspect. (b) Depth shaded image seen from the left lateral aspect. (c) The same data as Fig. 5a after applying the surface orientation shading algorithm. (d) The same data as Fig. 5b with surface shading.

Discussion

The display of shaded surface images is now an established tool in the field of surgical planning [27] and is proving useful for radiotherapy planning [18], and the evaluation of complex anatomy [24]. Application of these techniques in nuclear medicine has been restricted by their complexity, by the poor quality of surfaces obtained from noisy data, and by the fact that iso-count surfaces cannot show interior details. The algorithms described in this paper represent a possible solution to two of these problems. The relatively coarse nature of radionuclide tomographic data leads to a compact cuberille representation, which may be processed rapidly.

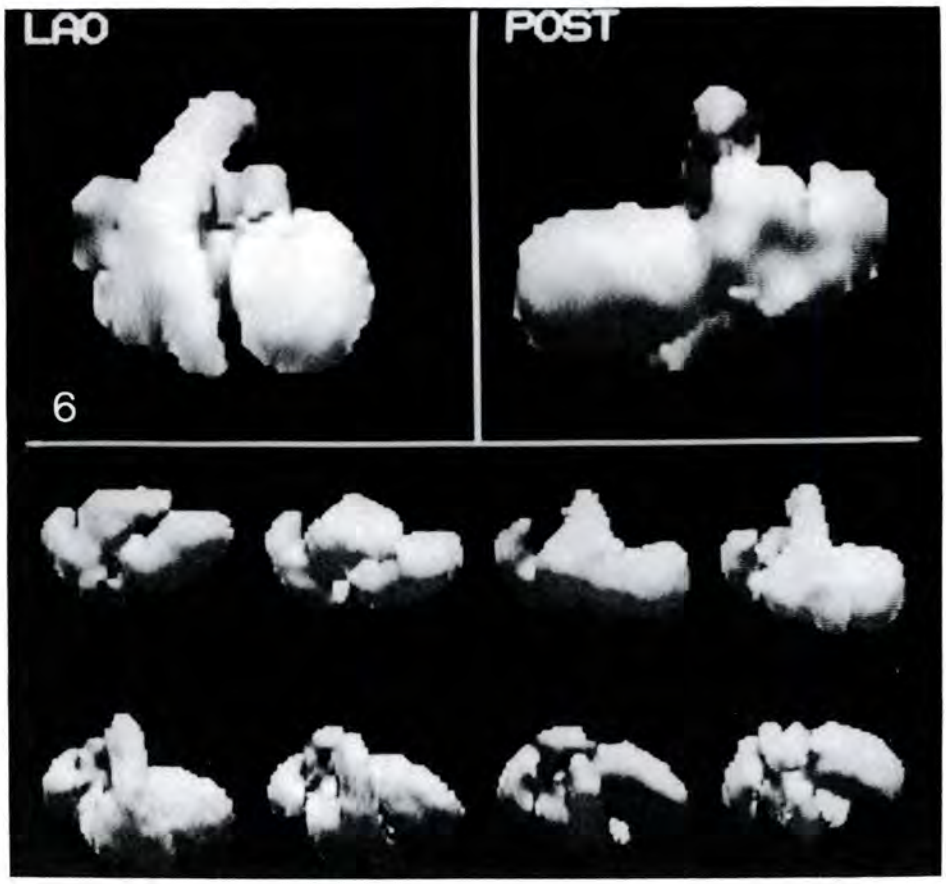


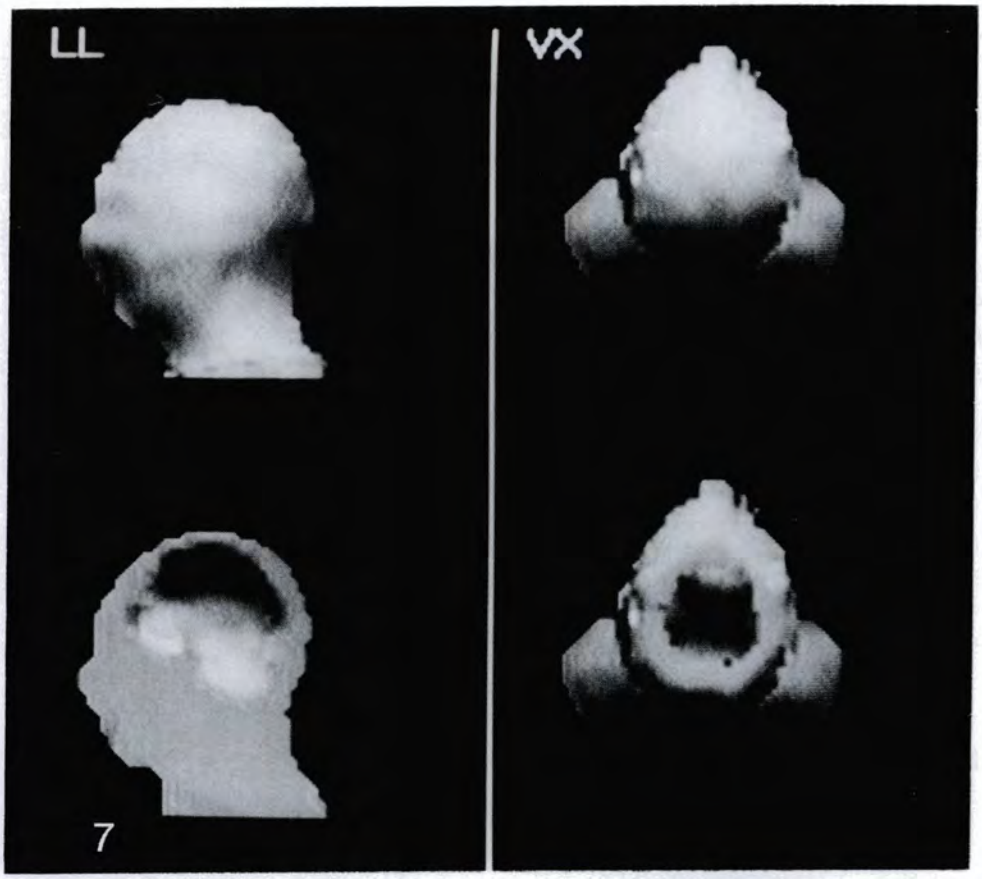
Fig. 6. Shaded surface images showing the cardiac blood pool from several aspects. The lower images show the object tilted sequentially to display the inferior, anterior and superior aspects.

Overall execution times for the selection of a new viewing aspect, determination of the visible surfaces, and display of the shaded image were typically in the range 30 to 45 s for the smooth shading algorithm, and in the range 10 to 12 s for depth shading only.

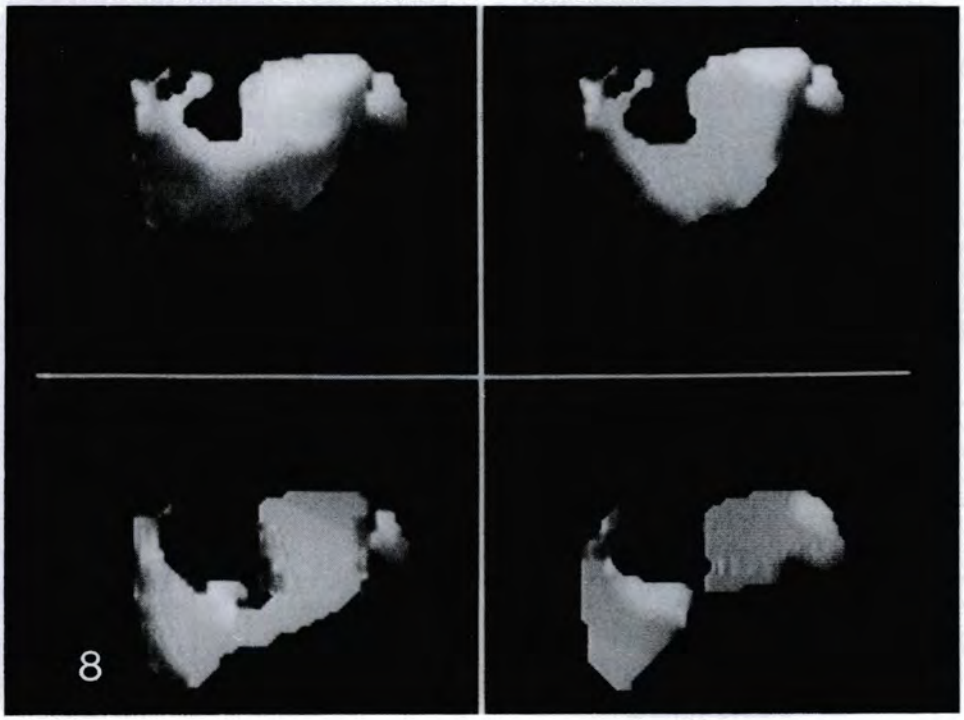
This is sufficiently fast to allow the program to be used interactively. The variable shape local polynomial fit procedure has the necessary flexibility to produce smooth

Fig. 7. A normal radionuclide brain scan showing the empty brain cavity, with uptake above the selected threshold in the scalp and facial tissues.

Fig. 8. An abnormal liver scan seen from the anterior aspect. As the binary object is removed interactively the extent of the abnormality can be determined.



7



8

surfaces from coarsely sampled data without blurring the edges of objects where true discontinuities occur. However, it is essential to retain as much information about interior structure as possible, since the exterior surface formed by an iso-contour gives no information about the variation of radionuclide uptake within the surface. The binary solid object representation described here can be combined with simple slicing and manipulative techniques [22] to interactively explore the three-dimensional activity distribution. Nevertheless the shaded surface image represents only the spatial distribution within an organ of areas where the radionuclide uptake is above the chosen threshold. These images are therefore combinations of anatomical and functional information, and should be interpreted accordingly. It is interesting to note that the activity distribution in the cardiac blood pool contains little or no interior detail since the concentration of radionuclide in the blood is essentially constant. Hence a simple thresholded object is an excellent representation of blood pool morphology. The results obtained using ungated tomograms suggest that the shaded surface display technique may be of particular value for the display of cardiac blood pool tomograms. The extension of these techniques to gated blood pool tomograms will allow abnormalities of wall motion and cardiac function to be assessed from any aspect.

References

1. Udupa JK. Display of 3-D information in discrete 3-D scenes produced by computerized tomography. *Proc. IEEE* 1983; **71:** 420.
2. Williams ED. Viewing multiple section SPECT images. *Nucl Med Commun* 1983; **4:** 215.
3. Maravilla KR. Computer reconstructed sagittal and coronal computed tomography head scans. *J Computer Assist Tomogr* 1978; **2:** 120.
4. Olson DO, Williams DL, Ritchie JL, *et al.* Calculation of transverse ventricular sections from transaxial ECT reconstructions of the myocardium. In: Functional mapping of organ systems. New York: Society of Nuclear Medicine, 1981, p. 167.
5. Rothman SLG, Dobben GD, Rhodes ML, *et al.* Computed tomography of the spine, curved coronal reformations from serial images. *Radiology* 1984; **150:** 185.
6. Artzy E, Frieder G, Herman GT. The theory, design, implementation and evaluation of a three-dimensional surface detection algorithm. *Comput Graph Image Process.* 1981; **15:** 1.
7. Chawla SD, Glass L, Freiwald S, *et al.* An interactive computer graphic system for 3-D stereoscopic reconstruction from serial sections. *Comput Biol Med* 1982; **12:** 223.
8. Perkins WF, Green RJ. Three-dimensional reconstruction of biological sections. *J Biomed Eng* 1982; **4:** 37.
9. Sidden RI, Kijewski P. Perspective display of patient external and internal contours. *Comput Biol Med* 1982; **12:** 217.
10. Fram EK, Godwin JD, Putnam CE. Three-dimensional display of the heart, aorta, lungs and airway using CT. *Am J Roent* 1982; **139:** 1171.
11. Fuchs H, Kedem ZM, Useton SP. Optimal surface reconstruction from planar contours. *Commun ACM* 1977; **20:** 693.
12. Cook PN, Batnitzky S, Lee KR. Three-dimensional reconstruction from serial sections for medical applications. *Proc. SPIE* 1981; **283:** 98.

13. Phong B-T. Illumination for computer generated pictures. *Commun ACM* 1975; **18:** 311.
14. Marino TA, Cook PN, Cook LT, *et al.* The use of computer imaging techniques to visualise cardiac muscle cells in three dimensions. *Anatomical Record* 1980; **198:** 537.
15. Preston DF, Batnitzky S, Lee KR, *et al.* Three-dimensional reconstruction and visualisation system for medical images. In: Proceedings of the 3rd World Congress of Nuclear Medicine and Biology. Paris: Pergamon, 1982, p. 518.
16. Minato K, Mukai T, Yamamoto K, *et al.* Three-dimensional display in single photon ECT imaging. In: Proceedings of the 3rd World Congress of Nuclear Medicine and Biology. Paris: Pergamon, 1982, p. 3651.
17. Herman GT, Liu MK. Three-dimensional display of human organs from computed tomograms. *Comput Graph Image Process.* 1979; **9:** 1.
18. Bloch P, Udupa JK. Application of computerised tomography to radiation therapy and surgical planning. *Proc IEEE* 1983; **71:** 351.
19. Marsh JL, Vannier MW. The 'third' dimension in craniofacial surgery. *Plastic Recons. Surgery* 1983; **71:** 759.
20. Axel L, Herman GT, Udupa JK, *et al.* Three-dimensional display of NMR cardiovascular images. *J Comput Assist Tomogr* 1983; **7:** 172.
21. Udupa JK. Interactive segmentation and boundary surface formation for 3-D digital images. *Comput Graph Image Process* 1982; **18:** 213.
22. Gibson CJ. A new method for the three-dimensional display of tomographic images. *Phys Med Biol* 1983; **28:** 1153.
23. Sutherland IE, Sproull RF, Schumacker RA. A characterisation of ten hidden-surface algorithms. *Comput Surveys* 1974; **6:** 1.
24. Totty WG, Vannier MW. Complex musculo-skeletal anatomy, analysis using three-dimensional surface reconstruction. *Radiology* 1984; **150:** 173.
25. Chen LS, Herman GT, Reynolds RA, *et al.* Surface rendering in the cuberille environment. Technical Report MIPG 87, Department of Radiology, University of Pennsylvania, 1984.
26. Gordon D, Reynolds RA. Image space shading of three-dimensional objects. Technical Report MIPG 85, Department of Radiology, University of Pennsylvania, 1983.
27. Vannier MW, Marsh JL. Craniofacial disorders. *Diagnostic Imaging* 1983; **5:** 36.