Improving 3D visualisation interactivity by using octrees

R.E. Loke, R. Lam, F. Rotaru and J.M.H. du Buf
Vision Laboratory, Dept. of Electronics and Computer Science
University of Algarve, 8000 Faro, Portugal
email: {loke,rlam,dubuf}@ualg.pt
URL: http://w3.ualg.pt/∼dubuf/vision.html

***Abstract* – O presente artigo descreve um método interactivo para visualização de dados tomográficos tridimensionais com enfase na segmentação e construção de superfícies. O método emprega a representação dos dados em multiresolução, permitindo assim uma rápida obtenção de resultados a baixas resoluções. Esta característica permite ao utilizador uma rápida interacção, inclusivé para grandes volumes de dados, a rápida selecção da região de detalhe, dos dados, permite evitar os altos custos computacionais inerentes à alta resolução. A desvantagem associada ao modelo de multiresolução é a perda parcial da conectividade, dos elementos de fronteira, a baixa resolução. Esta perda é recuperada usando técnicas de refinamento que melhoram a conecção a altas resoluções, através da aplicação de filtros espaciais nas fronteiras dos objectos.**

This paper describes an interactive method for the visualisation of 3D tomographic data with an emphasis on segmentation and surface construction. In the method multiresolution data representations are employed which have the advantage that results at low spatial resolutions are quickly obtained. This facilitates a fast user interaction, even for huge data volumes, and allows a fast selection of a ROI in the data by avoiding computationally-expensive processing at high spatial resolutions. The disadvantage of the multiresolution processing is that the spatial connectivity of the boundary is partly lost at low resolutions. This is recovered by an accurate refinement technique which improves the connectivity at high resolutions by performing a spatial filtering on the object boundaries.

***Keywords* – segmentation, surface extraction, 3D visualisation, octrees, interactivity, MRI data**

I. INTRODUCTION

Volumetric data processing extends traditional 2D image processing and facilitates the visualisation and analysis of the data in 3D, which highly increases the analytical accuracy. In the last years a visualisation pipeline has been developed for the volumetric processing of sonar subbottom data [1]. Because 3D data analysis becomes more important in medical applications, we also study the application of the developed segmentation and surface construction methods to tomographic medical data. Segmentation is used to extract relevant features out of data for their classification into homogeneous regions. We have developed an accurate 3D method which is very robust to noise in the data; it is based on the 2D method of Schroeter and Bigün [3], [4]. The surface construction method has only recently been developed. Unlike commonly used methods, e.g. [2], it determines accurate and efficient polygon meshes for boundary data which are, but not necessarily, completely connected. Both methods have a high level of user interactivity because they employ a multiresolution processing in an hierarchical data pyramid or octree. Because the tree construction and the processing at low resolutions are relatively fast, coarse processed data can be quickly visualised.

In this paper we focus on the integration of these methods in order to enable a fast interactive visualisation of the surfaces between the segmented regions in the data. **This integration of segmentation and surface construction enables a fast selection of a ROI in the data and it facilitates a fast processing of the selected data in very large datasets.** The structure of the paper is as follows. First we describe the integrated visualisation method. Then results obtained on MRI data are given and the work is discussed.

II. INTERACTIVE 3D VISUALISATION

The visualisation method consists of 3 processing steps: First, an octree is constructed in which low resolution data at high tree levels are determined by spatially smoothing the data at low tree levels. This increases the SNRs between the different regions in the data facilitating a clustering and a classification of the data at a tree level where the noise has been sufficiently reduced. Because the spatial connectivity of the boundary voxels is not necessarily retained in the smoothing process, a clustering technique must be used which does not employ any connectivity constraints. Therefore, after the classification, the voxels in each region are not necessarily spatially connected.

Second, surfaces are constructed for the boundaries of all regions in the segmented data, and these surfaces are visualised in VRML (Virtual Reality Modeling Language). VRML allows for a highly interactive analysis by "flying" through and around the data. Furthermore, it is fast if compared to volumetric visualisation. In order to construct the surfaces another octree is built by upsampling the segmented boundary data: boundaries which are not completely connected at a high resolution get connected at a lower resolution. This allows the construction of surfaces for regions in the segmented data which voxels are not spatially connected.

Third, starting at the tree level where the clustering was

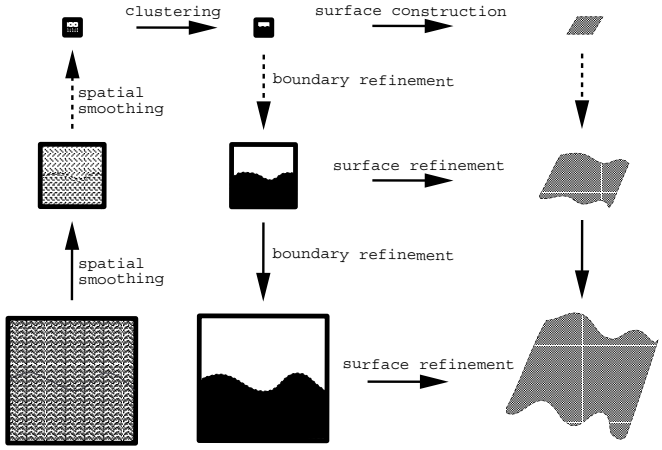


Figure 1 - Processing scheme of the 3D data visualisation. On the left and in the centre corresponding octree slices before and after segmentation are shown. On the right we can see the surfaces between the segmented regions. In order to illustrate the processing, artificial data which consist of 2 regions separated by a simple sinusoidal boundary obscured by noise are used.

performed and ending at the lowest tree level, at each tree level the boundaries in the segmentation are refined and the corresponding surfaces are constructed in order to update the visualisation of the data at the lower resolution. A boundary refinement technique is used in which the spatial connectivity of the boundary voxels is recovered by performing a spatial filtering *on* the boundary data.

Important is that after the second processing step has been performed, i.e. after the data have been visualised for the first time, the processing can be stopped at any moment in order to select a new ROI. The processing proceeds only if, according to the user, the ROI has been correctly set. If the ROI is incorrectly set, the processing is stopped and an octree is built for the new ROI. Below these 3 steps are described in detail. See Fig.1 for the processing scheme.

A. Spatial smoothing and classification

In the octree smoothing, voxels at level l ($l > 0$) are calculated by averaging non-overlapping voxel blocks of size $2\times2\times2$ at level $l-1$. The resulting tree contains $l_{max}+1$ levels in which level 0 represents the original data. At a level l_c ($0 < l_c \leq l_{max}$) where the noise in the feature space has been sufficiently reduced, a clustering algorithm is applied in order to determine the data clusters. Both supervised and unsupervised clustering algorithms have been implemented. In the supervised algorithm, the user can interactively select the clusters in the histogram of the data values. At the moment, local-centroid clustering [4] is used as the unsupervised algorithm. In this algorithm the histogram is iteratively changed by calculating, for each greyvalue x, a sum of histogram values in a local mask. The sum is weighted depending on the distance to x. This weighted sum indicates another greyvalue to which the histogram value at x is added. The algorithm stops when there are no more changes in the histogram. Then the histogram directly yields the clusters. The segmentation at level l_c is obtained by labeling each voxel with the label of its nearest cluster (minimum distance classification).

B. Surface construction

The basic idea of the surface construction is that by sampling the data in an octree the spatial continuity of the boundary voxels increases at higher tree levels. With some restrictions, high-curvature areas can be connected at high and low-curvature ones at low resolutions. Furthermore, separated voxels at a high resolution can be connected at low resolutions; see Fig. 2. In this way it is possible to construct large polygons covering flat boundaries and smaller ones in curved boundaries and at intersections. Below the processing steps are described.

B.1 Octree construction:

Let the original data of size $N_x\times N_y\times N_z$ consist of all boundary voxels of a segmented region denoted by $I1$ and background voxels denoted by $I0$. This is octree level 0. Starting at level 0, the following 4 processing steps are performed for each level n. First, if $n > 0$, a higher level is constructed by performing for each voxel a top-down search in the $2\times2\times2$ non-overlapping block of child voxels at $n-1$. If at least 1 of the 8 children is equal to $I1$ or is marked (see below), the parent is set to $I1$. Second, for each $I1$ voxel the spatial continuity in its neighborhood is determined. A simple model is used which only takes into account the 6 closest voxels: $x0$, $x1$, $y0$, $y1$, $z0$ and $z1$. Let $j01$ denote $j0 \neq I0 \wedge j1 \neq I0$, and X, Y, Z, XY, XZ and YZ denote $x01, y01, z01, x01 \wedge y01, x01 \wedge z01$ and $y01 \wedge z01$, respectively. Hence, X denotes a voxel with continuity in x ($I1$ in $x0$ and $x1$), and XY denotes a voxel with continuity in x and y (diagonal). Then, continuities are marked differently, depending on the boolean values X, Y, Z, XY, XZ and YZ; if $X \wedge Y \wedge Z$, it is left unmarked because we want only to construct polygons for region surfaces, and not for region interiors. This facilitates the 2D polygon construction. Third, if $n > 0$, for each marked voxel it is determined whether it is also valid at higher resolutions. A marked voxel is valid if it is equal to or higher than ($XY > X, XY > Y, XZ > X, XZ > Z, YZ > Y$ and $YZ > Z$), and not contradicting (according to 3 orthogonal groups: $\{X, Y, XY\}$, $\{X, Z, XZ\}$ and $\{Y, Z, YZ\}$) its marked child voxels at all lower levels. Fourth, it is determined whether a higher tree level must be added. Only when $N_x \geq 6$, $N_y \geq 6$ or $N_z \geq 6$ an extra level is built, because at a higher level at least 3 voxels are required in at least 1 spatial dimension for any continuity marking. If $N_x < 2$, $N_y < 2$ or $N_z < 2$ the tree is ready.

B.2 Polygon construction:

Let the constructed octree contain n_{max} levels. Then n_{max} surfaces can be built, independently of each other, at the different resolutions. Let s ($0 \leq s < n_{max}$) be the desired resolution level. Then, for each level l, starting at level $n_{max}-1$ and ending at level s, a polygon construction scheme is performed for each marked voxel. If $l > s$, it is performed only for voxels which are valid at s. In this scheme, vertices are determined for the relevant voxels by performing a top-down selection in the tree. Thereafter, a polygon is constructed by connecting the vertices in a, for XY, XZ and YZ differently defined, clockwise order.

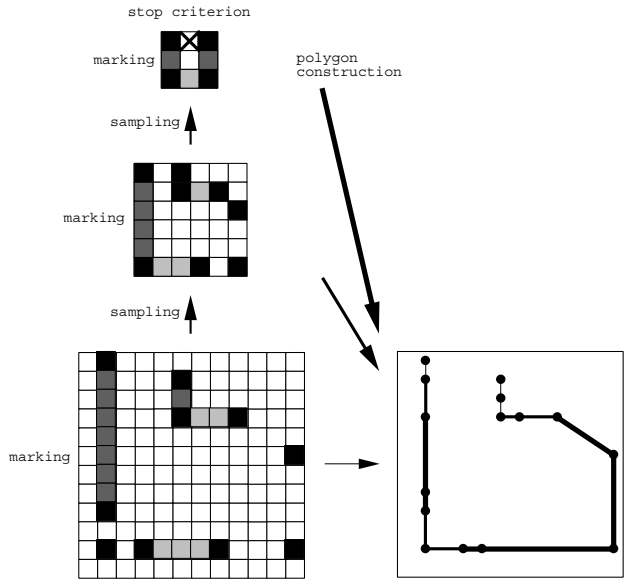


Figure 2 - 2D scheme of the processing steps in the octree which guides the surface construction and approximation. On the left an octree is shown and on the right a corresponding continuous space. Squares represent voxels and dots vertices. Filled squares denote boundary voxels, white squares background voxels. A grey square indicates that the spatial continuity of the voxel has been marked. The cross indicates that the marked voxel is not valid at level 0. Lines denote edges. The thicker the line, the higher in the tree it has been constructed.

C. Refinement

In order to construct and visualise the surfaces between the regions at a higher resolution, the boundaries in the segmentation must first be refined. The boundary refinement is performed for each boundary voxel and consists of calculations performed at 2 adjacent tree levels. First, at the higher level l which has already been segmented, the boundary orientation is estimated in the neighbourhood of the voxel. Second, at the lower level $l-1$ which is to be segmented, the original "noisy" data around each child of the boundary voxel are filtered taking the determined orientation into account. Filtering with a 2D planar filter on the boundary leads to very accurate boundaries, even for very low SNRs. Furthermore, it recovers the spatial connectivity of boundary voxels which are separated but which belong to the same region. The cluster which is allocated to the child voxel is determined by using a minimum distance criterion between the filter response and the cluster values. After this boundary refinement the corresponding surfaces are constructed as described before. See Fig. 3 for the processing scheme.

III. Results

The methods have been applied to MRI data of a head and a knee. We note that interactive visualisations of the surfaces and volumes with realtime rotations, colors and transparencies etc on a computer screen are much more realistic. More experimental results, of tomographic medical data as well as sonar data, including movies, VRML and OpenGL worlds, and raytraced animations, can be interactively visualised at our webpages (http://w3.ualg.pt/isacs/results/results.html).

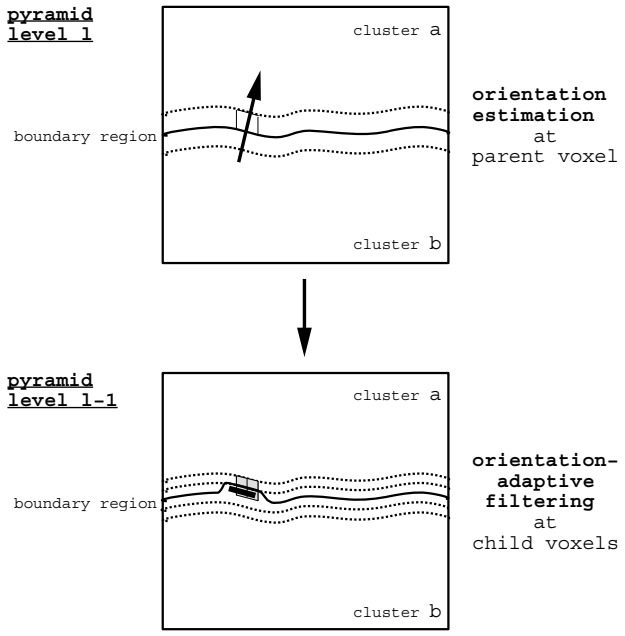


Figure 3 - 2D scheme of the 3D filter technique used for the boundary refinement. It shows 2 corresponding cross-sections at tree levels l and $l-1$. Level l has been expanded. At level l, the boundary region consists of 2 voxel layers; at level $l-1$, of 4. At level l the arrow indicates the orientation perpendicular to the boundary. At level $l-1$ the dark fat line indicates a planar filter positioned in the neighbourhood of a boundary child voxel.

A. Head

The original head data of size $256\times256\times109$ can be seen in Fig. 4a. The segmentation and surface construction methods have been applied to a selected ROI of size $160\times128\times96$ in this volume; see Fig. 4b. In both volumes, the presence of noise can be clearly observed. See Fig. 5 for results obtained on the data in this ROI. In the segmentation, an octree has been built which consists of only 2 levels. This is because the noise in the original data is relatively small; see Fig. 5d. Segmentation results at the different resolutions can be seen in Figs. 5b and e. The boundaries are very smooth and the spatial connectivity of separated voxels at level 1 has been recovered by the boundary refinement procedure. Figures 5c and f show polygons which have been built for 2 small symmetric regions in the segmented volume. The shapes of the regions at both resolutions are similar, showing large polygons in low-curvature areas and smaller ones in high-curvature areas. Figure 6 shows the surfaces of one region in the segmented MRI volume, the *cerebral cortex*, obtained by using a simple surface construction method. The surfaces are very coherent and there exist almost no small isolated regions. The cortex' characteristic shape and its 2 hemispheres can clearly be seen.

B. Knee

The segmentation and surface construction methods have been applied to a selected ROI of size $128\times140\times88$ in the original knee data; see Fig. 7. Segmentation results at the different resolutions can be seen in Figs. 8b and d. Figure 9 shows the surface of one region in the segmented MRI volume (the elongated bright one at the right in Fig. 8) which was obtained using the Marching Cubes algorithm and ap-

plying Gouraud shading. This region is called the *supraatellar bursa*. The surface is very coherent and there exist only few small isolated regions. Note that a gap exists in the constructed surface indicating a pathology. By careful inspection of the volume slices using movies it was verified that this gap exists in the original MRI data and, therefore, it is not an artifact introduced by the segmentation method.

IV. Discussion

The surfaces constructed by our newly developed methods still contain many gaps; this is due to the simple spatial continuity model used. We are developing a more elaborate model which incorporates all the voxels in the local $3\times3\times3$ neighborhood, and not only the closest. In the future we intend to develop a better OpenGL interface, which will allow to perform interactive 3D measurements, e.g., the calculation of distances and angles. Furthermore we intend to use datasets which contain higher noise levels as well as data obtained with other modalities, e.g. CT scans, and we intend to collaborate with medical imaging specialists in order to validate the visualisation methods.

V. Acknowledgements

The data visualised in this study are part of the Chapel Hill Volume Rendering Test Data Set (ftp://cs.unc.edu/pub/projects/image) of SoftLab Software Systems Laboratory, Department of Computer Science, University of North Carolina, Chapel Hill, NC, USA.

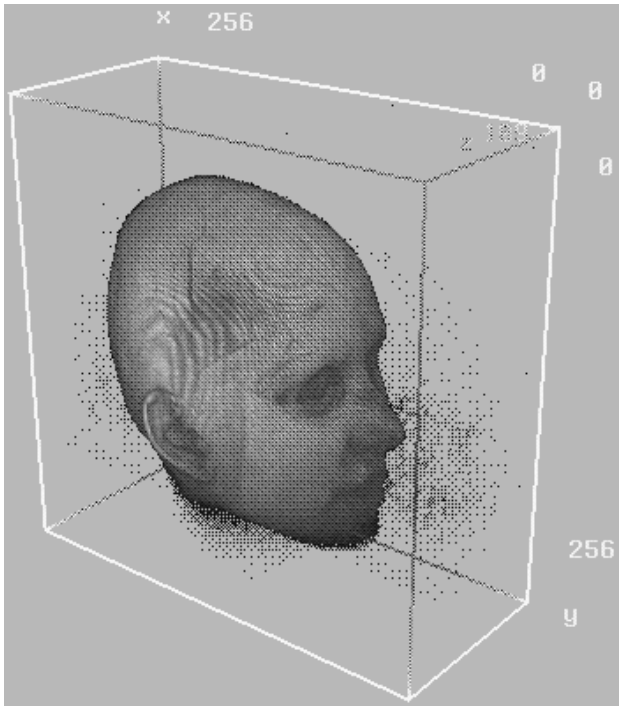
References

[1] R.E. Loke and J.M.H. du Buf. Volumetric processing of Topas underwater acoustic data. In *Proc. OCEANS'98 IEEE/OES Conf.*, pp. 742-746, Nice (France), 28 Sept.-1 Oct., 1998.

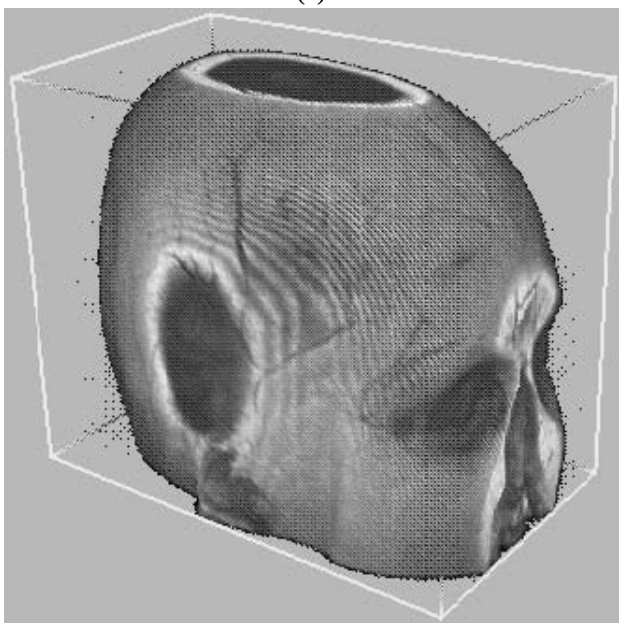
[2] W.E. Lorensen and H.E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *Computer Graphics*, 21 (4):163-169, 1987.

[3] P. Schroeter and J. Bigün. Hierarchical image segmentation by multi-dimensional clustering and orientation-adaptive boundary refinement. *Pattern Recognition*, 28 (5):695-709, 1995.

[4] R. Wilson and M. Spann. *Image Segmentation and Uncertainty.* Research Studies Press Ltd., Letchworth, England, 1988.



(a)



(b)

Figure 4 - Volumetric visualisation (in Bob) of the original head data (a) and a ROI (b).

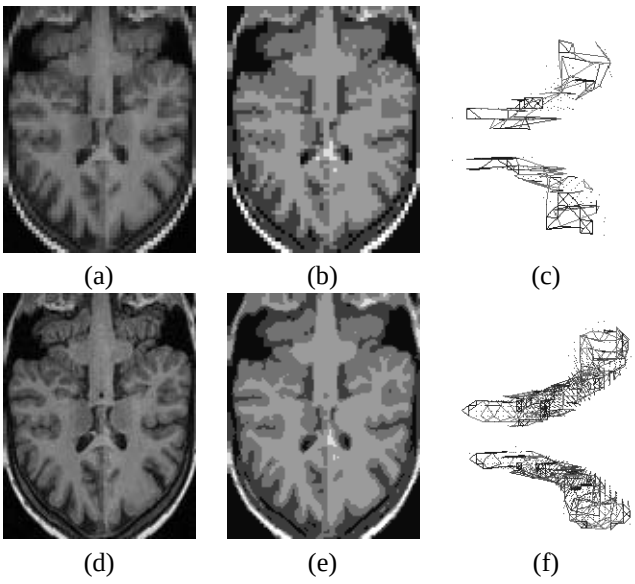


Figure 5 - Results of the pyramid smoothing, clustering, boundary refinement and polygon construction, obtained for the ROI in Fig. 4b. The left and centre column show corresponding slices of the MRI volume at octree level 0 (d, e) and level 1 (a, b). (a) and (d) are before, (b) and (e) are after segmentation. In the right column, polygons of 2 small segmented regions at level 1 (c) and level 0 (f) are visualised in VRML. The data at level 1 have been expanded.

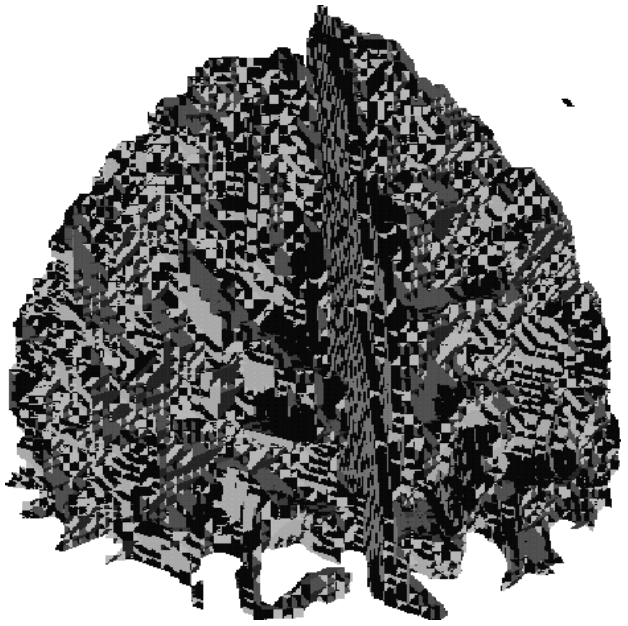


Figure 6 - Surface visualisation in VRML of the cortex found in Fig. 4b.

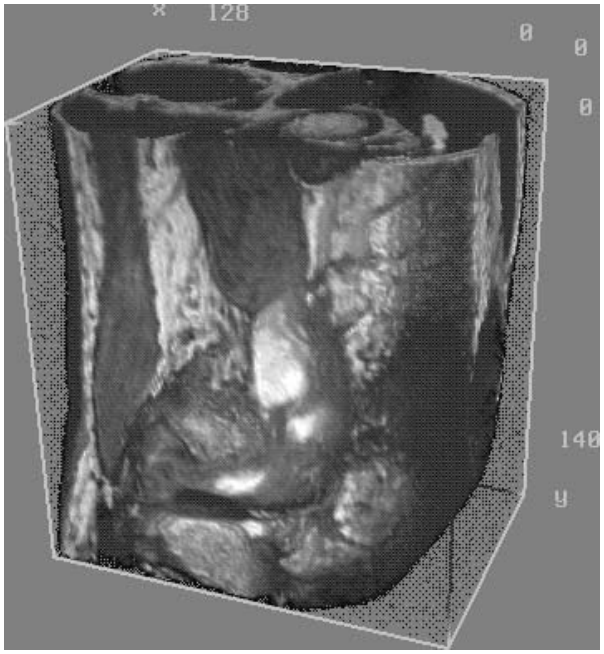


Figure 7 - Volumetric visualisation (in Bob) of a ROI in the original knee data.

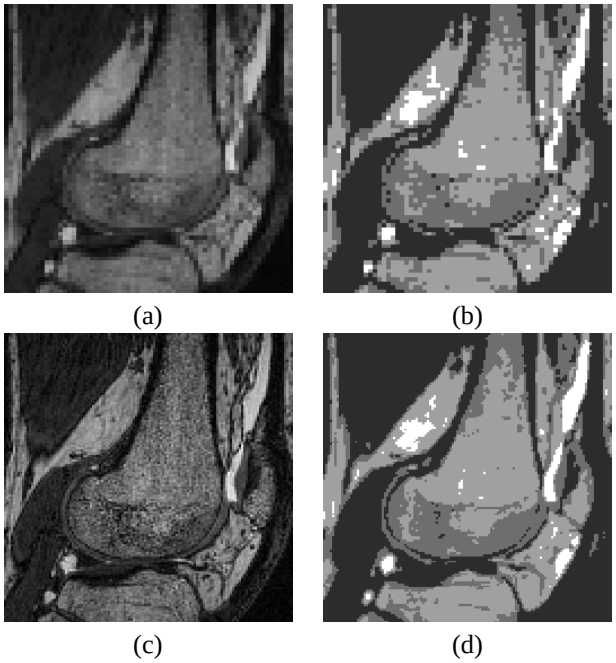


Figure 8 - Results of the pyramid smoothing, clustering and boundary refinement, obtained for the ROI in Fig. 7. The two columns show corresponding slices of the MRI volume at octree level 0 (c, d) and level 1 (a, b). (a) and (c) are before, (b) and (d) are after segmentation.

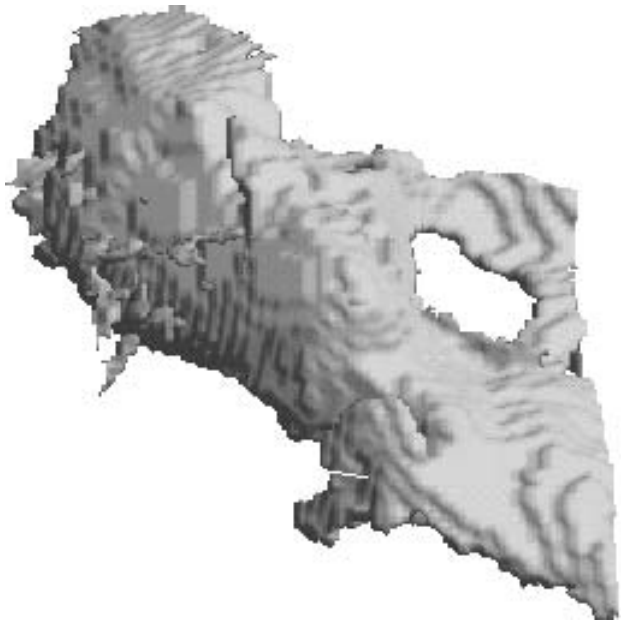


Figure 9 - Surface visualisation with Gouraud shading in OpenGL of the suprapatellar bursa found in Fig. 7.