Calculating CT data from matched geometries

B. ILLERHAUS, Berlin, Germany. (bernhard.illerhaus@bam.de)
J. L. THOMPSON, Las Vegas, USA (thompsoj@nevada.edu)

Abstract

Geometry data can be calculated from CT data [1] by defining isosurfaces. In some cases the total object has been measured in several parts, or the object has been divided by image processing in independent new parts. In rejoining or rearranging these parts in virtual space a control of the positioning is desirable. Therefore we set up an image processing module (implemented for use with AVS [2] software) which uses the displacement vector of the geometries to generate a joined CT data set from the original ones.

From geometries to voxels

These vectors and matrices can be used to position the original or the partial CT data sets into one common voxel environment. To save computer memory each CT data set is read in slice by slice and immediately processed: shifted and turned and interpolated onto voxels in the common volume. To simplify, a small error is accepted, as the interpolation is done to nearest neighbour distances, but not in projection direction (see fig.1).

Fig.1: Transposing and interpolation diagram and transposing formula (right)

Each generated isosurface can be set to have its geometric origin to the first read-in voxel of the CT data. As the size of the voxels is known, each voxel has a fitting
spatial vector. The moving and turning of the isosurfaces is done either in respect to the first or the centre voxel, which can be described by an origin vector for each isosurface and a matrix. A second displacement vector and turning matrix for compound objects may be added.

The result is a new 3D CT data set, joining the information of the original data sets and the geometrical transformations yielded from the positioning of the surfaces. All know image processing algorithms can now be used again to characterise the joined image. Furthermore, the positioning of the different geometric objects can thus be exactly controlled. Especially the matching of hidden surfaces when joining geometries can be proved: In case of a second contribution to the same output voxel the content is added, thus showing unusual high material density in overlapping parts.

Calculating the brain volume of Le Moustier 1

This procedure was used while processing the virtual reconstruction of the Le Moustier 1 skull [3,4]. After having done the reconstruction on computer screen, the different CT data sets were joined. Fig.2 shows a slice in the new common voxel space.

In a next step the inner surface of the skull can be visualised and the total volume was calculated. The "invader" module was used: from a given starting point all adjoining points in a given range of grey levels in a limited region are marked, processing slice by slice in a loop until no more changes are registered (first used in [5]). To prevent the searching algorithm from escaping the inner volume of the skull all openings, e.g. missing bones or natural openings, have to be blocked. This introduces a certain error when counting all marked voxels for volume measurement. But as the blocking was done slice by slice and was reshaped in orthogonal slices, the error in calculating the brain volume of the virtual skull reconstruction is small and estimated to be much less then 1% by the change between the first try and first orthogonal correction. (Detailed results to be published.)

Fig. 3 gives the surface of the brain. This is generated by a logical "and" of the marked voxels with the density voxels and a density inversion: The brain surface bears all features of inner side of the skull.

Figure 2: Slice in the compound voxel space.

Figure 3: Surface of the reconstructed brain of Le Moustier 1 skull
Figure 4: The new reconstruction of the Le Moustier 1 skull

Figure 5: Current state of Le Moustier 1 skull

Literature


