3D Object Modeling and Visualization Software for Surgery Preoperative Plan

Paul-Nicolae Ancuta
INCDMF Bucharest,
6-8 Pantelimon Street, Sector 2,021631,Bucharest
apaul@cefin.ro

ABSTRACT

Shape modeling using sampled data has been extensively studied in the last years. A classic issue of 3d reconstruction may consist of modeling the object using medical images or scattered points lying on the object surface. The article presents some aspects of 3D modeling and visualization techniques and the author's achievements in modeling the human femoral bone. Several software packages for 3D reconstruction and visualization were used providing mesh conversion in stl format for further CAD processing: VTK, ScanTo3D and Technodigit SDK.

INTRODUCTION

The preoperative planning in surgery needs an exact geometric and topologic description of the anatomic structures involved. The first step in anatomic 3D modeling is getting a CT or MRI stack of images, which contain voxels. The image processing should extract the areas with specific information and label them, operation called segmentation. Surface modeling implies obtaining adjacent triangles which create a surface mesh. Computer aided design software uses this mesh for further processing such as solid conversion, properties assignment, FEA assembly studies a.s.o.

The reconstruction methods may be organized as follows:
1. Surface reconstruction using planar cross sections. This is the preferred method used in medical imaging, sustained by CT, MRI or PET investigation, which obtains a series of 2D images of the patient. 3D reconstruction should be able to create the virtual image of the anatomy part.
2. Reconstruction from scattered points obtained from the surface of the object by 3D scanning, radar, etc. This type of reconstruction makes no assumption about object topology thus points could be freely acquired.

SOFTWARE FOR APPLICATION DEVELOPMENT-VTK PARADIGM

The Visualization Toolkit (VTK) [1] consists of a general purpose collection of open source, object oriented and portable software. It is widely used in universities, research centers and commercial software packages. Our survey has pointed out that VTK could be found in medical imaging software, Finite Element Analysis software, 3D reconstruction, Level of Details, air flow modeling and general purpose visualization. The
VTK implementation is carried out in C++ language. VTK provides Tcl, Phyton and Java connection. Also, one could use C++ for specific software applications. VTK contains two major components (Figure 1): a compiled kernel, written in C++, which provides speed and efficiency and an interpreted layer, which sustains Tcl, Java, Phyton programming languages and offers flexibility for applications.

Data structures, algorithms and time critical system functions have been implemented in C++ kernel. Modeling with VTK is founded on a pipeline processing paradigm. Software modules are connected in a processing network which reacts on events such as data requests or user events. Such approach provides a clear border between system modules and user modules, giving also patterns for procedural extensions.

VTK contains two types of objects:
- Processing objects which process data
- Data objects which navigates through the network.

The process objects contains three types of subsidiary objects: source objects, filter objects and mapper objects. Source objects initiates the network and generates one ore more data objects. Data processing is obtained using filters. The input of Filter objects consists of one or more data objects and produces one ore more data objects. The Mapper objects are network ends and transform the processed data in abstract graphical data for storing in files with different formats, graphical, binary, ASCII and / or display purposes.

We can obtain various processing networks by interconnecting the enumerated types of objects (Figure 2).
The network may contain branches and loops. Data could be obtained from two sources:
1. Reader objects data files
2. Procedural object computed data

CASE STUDIES

3D representation of an isosurface using VTK’s Marching Cubes Implementation
We used a stack of 2D images contained in DICOM files (Digital Imaging and Communication in Medicine). We used VTK implementation of Marching Cubes Algorithm [2] which creates and displays an isosurface made up of triangles which interpolate the points with the same Hounsfield value. This value must be somewhat provided. The obtained results using the author’s software are shown in Figure 3.

![Figure 3: Marching Cubes for Femour Surface Generation](image)

Surface generation from points using VTK’s Surface algorithm
VTK software development system contains an implementation of Hoppe’s algorithm [3] for 3D reconstruction from scattered points. For our purposes, we used a testing data set as follows: A virtual femur bone model (a SolidWorks part design) has been “cut” with parallel plan with a distance of 1 mm. The result consists of 466 slices. We use semiautomatic procedures for contour point picking and mesh reconstruction. The mesh has been converted to STL files for further processing with CAD software (Figure 4).

![Figure 4: VTK and STL Surface Generation](image)
The first step of the algorithm is getting the value of signed distance from the entry points set (nominated as X) to the real surface of the object (nominated as U). The sign distance of an arbitrary point \( p \in \mathbb{R}^3 \) to a known surface U is defined as:

\[
d_U(p) = s(p) \cdot d(p, U)
\]

where \( s(p) = \pm 1 \) according to point \( p \) position next to surface U.

If one can determine that function, the U surface is implicitly determined, because U is defined by the Zero Set function:

\[
Z(d_U) = \{ p : d_U(p) = 0 \}.
\]

The cubes where \( d_U(p) < \text{threshold} \) contribute to the surface representation.

**Surface generation from pointcloud using Technodigit SDK Kit**

![Figure 5. Pointcloud, STL and VTK visualization](image)

Figure 5 contains three images: left side contains the pointcloud visualization, middle contains the STL file obtained by processing the pointcloud with Technodigit SDK Library [4] and the right side contains the same representation as a mesh (VTK proprietary format). All the pictures were obtained using a proprietary software designed by the author.

**Solid modeling using ScanTo3D module**

![Figure 6. Stages in solid modeling using ScanTo3D](image)
ScanTo3D [5] is a powerful tool for creating solid models from pointclouds (Figure 6). Now it is integrated in SolidWorks Office Premium software. For a given pointcloud contained in an ASCII file as points coordinates, the software tries to obtain a closed surface as a starting point for solid modeling. The solid object can further be processed with solid tools provided by SolidWorks, it may be assigned material properties for FEA processing, it may become a part in an assembly and others.

CONCLUSIONS
All VTK mesh models can be exported in STL files for further CAD processing.
Marching cubes implementation provided by VTK is fast, but may produce artefacts. It creates an impressive file in native VTK format.
The implementation of Hoppe's algorithm provides satisfactory results if the conditions imposed by theory are fulfilled, such as point sampling and points distance. The user can set a threshold which guides the algorithm about the interaction of the neighbourhood points on the current point's normal plane. For large set of points, the implementation is slow (tens of seconds) when the software runs on a regular PC. Also the generated mesh may have holes.
The Technodigit SDK Library provides a robust and fast implementation of pointcloud processing. The user has multiple choices for parameters selection, but not any combination of parameters leads to success.
ScanTo3D module offers a lot of processing possibilities for a pointcloud set. One of the most important feature is solid modeling in a native SolidWorks format.

REFERENCES
[4] www.3dReshaper.com\en1\En_library.htm