A Meshed Atlas Visualization Toolkit in a Java-based CAD Platform

Avijit Roy[1,2], Francis E. H. Tay[1], Wieslaw L. Nowinski[2]

[1] Dept. of Mechanical Engineering, National University of Singapore, Singapore 119260
[2] Biomedical Imaging Lab, 30 Biopolis Street, #07-01, Matrix, Singapore 138671
E-mail: avijit@nus.edu.sg, avijitroy@sbic.a-star.edu.sg

Abstract—*In this paper we have demonstrated a Meshed Atlas Visualization Toolkit (MAVT), an automatic mesh generator, which is a set of comprehensive libraries as well as a number of interactive tools in Java Platform. It covers a wide range of fundamental functions from image conversion and transformation, image segmentation and analysis, to geometric model generation and manipulation, all the way up to 3D visualization and interactive simulation. The toolkit successfully visualized a multi-structured 3D biomechanical human brain CAD model (3D meshed atlas) using the Cerefy Brain Atlas database. The scope of the tool has also been expanded to the collaborative CAD environment through a virtual design studio (VDS) that facilitates synchronous dynamic collaboration between geometrically dispersed users. The developed tool is expected to contribute in neurosurgical simulation and other clinical and web based application development. We have validated our model by geometric matching 2D cross sections and in-vitro indentation experiment on pig brain to compare tissue deformation with the result.*

Keywords—*Brain Atlas, mesh generator, soft tissue deformation, mathematical modeling.*

1. Introduction

As the experimentation with real human brain sample is impossible, and study on corpse or even animal brain has been restricted with a lot of ethical issues today, there has been a movement of taking all design, engineering, bioengineering and biomedical applications from a physical to virtual since 1950 [1]. Today virtual reality has been advanced in a dramatic speed opening numerous possibilities which can be used to assist, replace or improve the quality of existing medical services. Especially in medical education, one of the important skills to the students is their anatomy understanding and anatomy reasoning skill. In medical educational institutes, it has been hypothesized that the ability to localize a cross section correctly in body is measure of one's anatomical expertise [2]. Medical students are required to have a lot of practice in autopsy to acquire sufficient human body anatomy knowledge. With the help of 3D models presentation, students can get a sense of spatial relationship of anatomy structures, which is very important for clinical problems requiring anatomical understanding [3]. By manipulating 3D models in a visualization or virtual reality interface, students can explore anatomy inside the body without physically cutting open a subject. Students can study and explore the anatomical structures over and over again without any cost on the actual specimens.

Not only just in training and education, the virtual reality is becoming an essential part in medical surgery simulation too. The advances in medical imaging beginning from the discovery of X-rays for more than 100 years ago open nowadays a new perspective for improvement of computer assisted surgery planning (CASP). As surgery become more and more complicated, the need for specific training has increased at a phenomenal rate. Meanwhile modern medical imaging techniques, such as computer tomography (CT) and magnetic resonance imaging (MRI), are widely used for diagnostic and visualization purposes. After FEA software became popular in last few decades, various efforts were given to develop realistic biomechanical models of brain to predict intra operative deformation or the brain shift phenomenon.
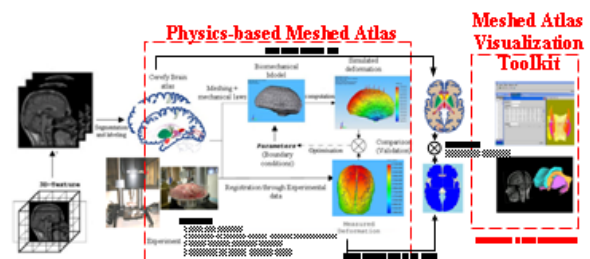

Figure 1: A physics-based atlas with the proposed mesh generator.

These physics based modes not only enable the derivation of 3D models of human anatomy, but also offers us to simulate physical interactions with virtual bodies. In such situations the realistic simulation of soft tissue deformations under the impact of external forces is of crucial importance.

The nearer a physical model approaches the properties of a living tissue, the more realistic the simulation results can be obtained. Such models were proved to very useful to update the pre-operative surgical planning and help the surgeon to avoid interfering with critical organs, by balancing the deformation effects of the structures visible on the post-operative MRI. Many researchers have proposed verities of biomechanical volumetric model which could explain the measured deformations [4, 5, 6, 7]. In addition, visualization in virtual platform offers new surgeons, students and trainees much more chances to learn and practice in simulation environment to gain necessary skill and expertise. In most of the renowned medical institutions today, the virtual anatomic models play one of the most critical roles. The significance and effectiveness of the simulation and education result largely depend on the virtual anatomy models. We have developed a multi-structured 3D biomechanical virtual anatomy CAD model using the *Cerefy* Brain Atlas database [8, 9]. The detectable interior structures are first extracted from atlas, forming feature points. The feature points of the individual structures are used to form point clouds in a CAD platform where the entire surface and volumetric model is built (Figure 1). The data has been uploaded in Meshed Atlas Visualization Toolkit (MAVT) for the visualization in Java Platform. In this paper our main focus will be on MAVT, the newly developed automated mesh generator and visualizing toolkit.
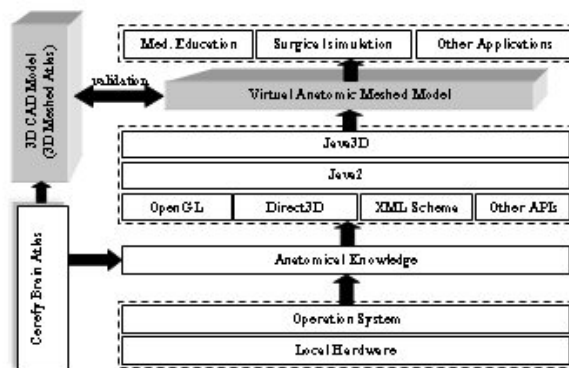

Figure 2: The architecture of the virtual anatomic modeling environment.

Virtual anatomic models are quite complex and difficult to be presented by most of the existing virtual 3D modeling languages, such as VRML. One of the shortcoming of the existing virtual reality modeling languages are unable to store and present the anatomic information, as well as special anatomic structures, for example, tubular anatomical structures. They are also not flexible enough to meet the requirement of advanced education and simulation requirement in terms of visualization and interaction. Hence our objective is to build a framework which defines a suitable data schema for presenting virtual anatomy models and meanwhile builds an interactive modeling environment for direct manipulation of the virtual anatomic models. Figure 2. shows the architecture diagram of this framework. This environment will services two purposes, first is to be used for interactively building virtual anatomic models; and second is to be used as prototypes of medical educational or simulation applications.

Depending on the application methodology of the anatomy models, the organization of anatomy models may vary quite significantly. For example, some applications may dissection from the systematic anatomy first then regional anatomy, but some applications may start from regional dissection then systematic. During the procedure of simulation or education, the organization might be required to change from time to time. Therefore, the function to interactively re-organizing and re-structuring is useful and important. Both the foundation data structure and the GUI take into account this requirement. On the other hand, depending on the objective and interaction conducted in simulations, models of different modality may be invoked simultaneously or individually and modified now and then to get a better understanding of the anatomy relationship and realistic simulation. That means the visibility and appearance property of any sub-models of individual structures or different modalities need to be manipulated by users as their pleasures.

In short, our work is aimed to build a framework to visualize virtual anatomic (brain) models that can be used in Computer-Aided Design platform as well as in Java platform. This framework includes the foundation data structure design as well as an interactive environment for manipulating the anatomical models. The foundation data structure is designed for presenting structural and topology anatomy with multi-modal data models. Anatomy knowledge has been implemented in the model schema design especially building intracranial structures. The interactive environment is designed to manipulate the virtual anatomy in the way that medical education and simulation required.

2. Anatomic 3D Meshed Atlas

Most of the previous researches were motivated to build various kinds of brain models capturing anatomy [8, 9, 10, 11], function [12, 13], and vasculature [10, 14]. Recently Roy, Nowinski and Tay [4, 5], Miller, Taylor and Nowinski [15] have

constructed a physics-based atlas model using The *Cerefy* Brain Atlas is an electronic data base that contains an extended and enhanced electronic version of the Talairach-Tournoux (TT) brain atlas. It has been derived from classic brain atlases edited by Thieme [8]. The electronic atlas images were fully segmented and labeled including subcortical structures and cortical regions. The feature points for our model have been extracted from the 2D structures of this complementary atlas digitized in 27 plates. Because the *Cerefy* Brain Atlas is fully colored and labeled, the *feature points* (a set of point database describing the outlines or surface features of an object) of each structure were easily extracted. For example, the RGB value of the *corpus callosum* has been denoted by number: (130, 75, 130); *putamen*: (0, 135, 91); *hippocampus*: (179, 147, 179); *caudate nucleus:* (255, 239, 0) etc. The surface features of 43 structures were extracted from the images (bitmaps) of 27 axial plates separated about 2-5 mm vertically from each other. The feature points of the individual structures have been extracted to form *point clouds* (a set of three-dimensional points in 3D CAD). The surface models were constructed in a CAD platform from these point clouds. The model was then filled with solid tetrahedrons and prepared for Finite Element Analysis. The typical flowchart is as follows: *Bitmaps → 2D feature points → Point clouds → Surface mesh → Volumetric mesh.* Detailed steps of construction of meshed atlas will not be discussed in this paper. For detailed discussion kindly refer to the work of Roy, Nowinski and Tay [4], [5]; Miller, Taylor and Nowinski [15].

The state of art approach for the geometrical modeling is the generation of surface models and is known as *triangulation*. The triangulated surfaces represent the boundaries between the different tissue regions. Surface models substantially reduce the original data amount and enable a compact representation of complex geometrical structures. In this work the creation of surface models with correct topology and optimized triangular shape from the point clouds of atlas data is carried out automatically with the help of marching cube algorithm used in *Macromedia Director MX v10.0.* Adaptive refinement and coarsing of surface meshes is used to achieve an optimal resolution of curved regions (for example *corpus callosum* or *caudate nucleus*) and at the same time to keep total numbers of elements as low as possible. For numerical simulation of continuum mechanics problems via finite element method, a volumetric meshed model is required. Such meshes can be categorized as structured and unstructured [16]. Structured meshes exhibit uniform topological

structure that unstructured mesh lack. For this project an unstructured tetrahedral mesh has been generated on the basis of the non-manifold surface triangulation used. Two key points were considered while meshing. First, geometric features must not prevent the mesh from being created and must also contain surfaces of consistent size and shape ratios to prevent forcing high *aspect ratio* and/or transitions between edges that may comprise accuracy. Generally a good rule of thumb for minimizing occurrence of high *aspect ratio* elements is to limit transitions of 2:1 or less, if geometry is broken into patches [17]. Secondly, simplification or manipulation of features in an attempt to clean up geometry would not reduce structural integrity of the part. In the early stages of design a larger element size was specified for a faster solution. Later, for a more accurate solution, a smaller element size was chosen. The volumetric meshed atlas model has been created in *COSMOSWorks™*. The model is then loaded in *Ansys* for FE analysis. A weighted smoothing algorithm in combination with an aspect ratio and Jacobean checking process was applied iteratively to guarantee a good quality volumetric mesh. Then the model is made ready for further verification and FEA analyses. The entire brain model (surface area: 82170.56 mm^2) consists of 15452 elements with 23461 numbers of nodes (element size: 6.1691mm, tolerance: .03085). Figure 3 shows the surface and volumetric mesh of the constructed model.
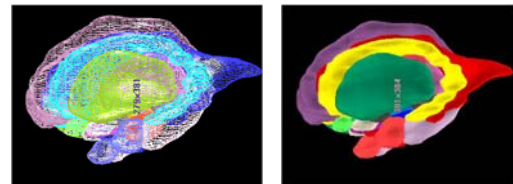

Figure 3: 3D model of brain : (a) surface mesh, (b) volumetric mesh.

3. Visualization in CAD Platform

Computer Aided Design (CAD) system has been viewed as a tool only for mechanical or civil engineers. Early CAD/CAM systems focused only on improving the productivity of draftsman [21]. But from last three or four decades, they are compelled to focus on modeling objects. *SolidWorks™* has become the standard CAD software for medical device manufacturers since long. Recently *SolidWorks* has offered many new features to design complex parts including the positron emission tomography (PET) scanners, a new breed of medical technology that detects heart disease, cancer, and brain damage earlier than CAT scans, X-rays, or MRIs etc. However, to construct a

complete brain model using SolidWorks was a major challenge for us. Even though literatures do not show any sign of construction of 3D brain model using SolidWorks, we decided to develop the model using Solidwoks as its ease of use and intuitive nature that allow to design rapidly and modify parts and assemblies. The model was created mainly using a tool called *loft. Loft* uses cross sections to extrapolate along a curve. It lets user to create complex 3D shapes by interpolating multiple 2D cross-sections of various size [19]. Loft connectors define how models profiles align. Construction of 3D model of some interior structures such as *putamen* or *Hippocampus* appeared to be very straight forward and easy, while constructing some other parts such as *corpus callosum* or *caudate nucleus* were never been straight. As Main difficulty was to maintain the connectivity in the multiple divisions in two successive slides, we had to construct those structures breaking into several steps, piece by piece. In addition to the *loft*, several other operations have been performed for the construction of the model; four important operations are stated below:

a. **Intersection:** The purpose of the operation is to obtain the intersection part of two input shapes. For example shape A intersects with shape B to obtain shape C is denoted as : $C = A \cap B$

b. **Cut:** The purpose of the cut is to obtain the subtraction part of the original shape subtracted by the destination shape. For example, shape A is subtracted by shape B to obtain shape C is denoted as $C = A - B$

c. **Difference:** The purpose of difference is to get the resulting shape that does not intersect with any of two shapes. For example, when shape A differences B to obtain shape C, the result denoted as $C = A \cup B - A \cap B$

d. **Combination:** The purpose of combination is to obtain the sum of two shapes, which can be denoted as: $C = A \cup B$

The following figure (Figure 4) shows the construction of 3D model from atlas plates and visualization in *SolidWorks* Platform.
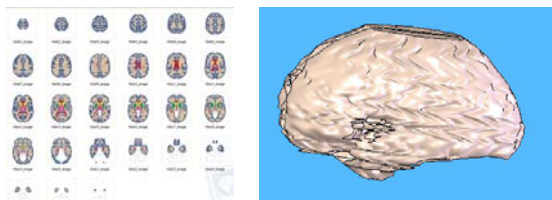

Figure 4: (a) 27 plates of the Cerefy Brain Atlas, (b) 3D model from the atlas data.

4. Building Meshed Atlas Visualization Toolkit on Java Platform

Meshed Atlas Visualization Toolkit (MAVT) is developed based on JAVA™ and its 3D API JAVA3D™. Java3D has a rich set of APIs, which provides powerful and flexible interactive functions useful for building educational and simulation systems. In addition, the other advantages of using Java are multitudinous such as -object oriented mechanism, platform independency, Interactivity of Module Behavior etc. VRML, OpenInventer, etc were also other contestants that can be used for the visualization purposes. For e.g, VRML is the first international standard for the description of 3D scene data [18]. VRML is a pure modeling language which leaves the visualization functions to the VRML browsers. It is easier to be used by model designers in certain sense. However it is not flexible enough in terms of defining new model paradigms. Another shortcoming of VRML as we mentioned before, is its inability to store anatomic information. In addition, it has less control on its rendering effects and it is not very convenient in describing advanced interactions. Java3D and OpenInventer are programming library packages for building visualization applications. Solid software design and programming background are required to use these packages. They are more flexible in building specific interaction functions. They also provide more prosperous rendering functions.

In the case of constructing MAVT, we used OpenGL, Direct3D, XML, Macromedia director with Java3D has been used to build static and dynamic properties including highlights, mapping textures, lighting, shading, motion constraints, logical judging and animation effects, etc. However, it is worth to mention that professional knowledge on visualization is always required to make use of the advanced features of Java3D to build models and interactions. Some comprehensive and abstract objects need to be design and developed in order to present the multi-modal anatomy model. Data structure for building foundation of virtual anatomy models for biomedical simulation and education has been designed as an enhancement of the scene graph, which gives a more abstraction and simplification in general biomedical information models. The data structure is an enhancement of the scene graph concept. It is an abstraction of general anatomical models.

The foundation class is the basic class that acts as the interface between MAVT and the Java and Java 3D API. The most important foundation class is *SimulatorApp* that extends *App* class and contains *SimulatorUniverse* inside. Since the picture rendered with Java3D are called *scenes*, MAVT core framework is based on a scene-graph programming model. The *SimulatorUniverse* can be broken into following components: geometry, GUI, behavior, mathematics, transform, model and appearance.
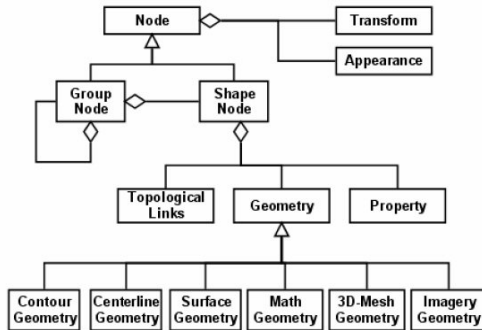

Figure 5: The UML model of the foundation data structure designed for virtual anatomy models.

A comprehensive data structure significantly simplify scene models used for medical education and simulation purposes, and hence also simplify and enhanced the operations for construction of virtual anatomic models. At the center stage of MAVT framework is the SceneGrapgh object, which is encapsulated inside *Simulator*. Each section can be divided into several subsections. For example, Geometry is further subdivided into contour geometry, surface geometry, mesh geometry etc. Mesh geometry includes an important public class called MeshMaker that extends basic Object class and responsible for creation of meshed model.

Figure 5 shows the foundation data structure of the multi-modal anatomy model in UML (Unified Modeling Language) diagram. Where the Group node are similar to Java3D scene graph transform group node, except it always attached a switch group, which allows the visibility of sub-models being manipulated easily. The geometry nodes are composted by the geometry node in Java3D with some functions needed by more complex modal data.

5. Collaboration in Virtual Design Studio

Development of an Atlas Visualization Toolkit in a virtual design studio (VDS) facilitates successful collaboration. Computer supported collaborative work is a field of research that is concerned with understanding the way people work in groups and the enabling technologies of computer hardware, software and networks. MAVT aims to develop a collaborative virtual environment to overcome geographical constraints, shorten product development time and cost through the Internet. The design activities have been studied and Java™ RMI and Java™ Media Framework has been implemented in the virtual design studio to enable effective and efficient collaborative design. As mentioned before, brain is very complex in structures; it might be a good idea to distribute the entire design work to various users depending upon their domain knowledge. It is possible that some structures of the brain are designed in a computer residing Japan, where as some other parts are developed in Singapore. Finally they all are combined in a user's computer residing USA. Such aspect of collaboration is becoming more practical day after day. The main purpose is to address existing group design processes and the methods to utilize computer technologies to distribute them. Thus a design environment must be established to make full use of all the related domain expertise, technologies and resources available in the world to achieve the common goal. The roles of MAVT in the VDS and the way it facilitates to interact with other parties participating in a particular design activity needs to be described.

In this section we will present how MAVT can implement *Synchronous* and *Asynchronous* collaborative design and virtual environment over the Internet according to the various design perspectives. *Synchronous* designing occurs when the computer and the designed objects are used as interactive tools by more than one designer. At the lowest end of the complexity spectrum is the electronic white-board where the designers draw on a common electronic document. At a higher level of complexity, designers could be working on a 3D model simultaneously in real-time. This becomes real time multi-user designing. *Asynchronous* designing occurs when the computer is used by a group of designers that do not see each other's changes instantaneously and are not constrained by other's actions. If a design starts from a common state and proceeds asynchronously, then multiple versions are produced. When all the versions are brought together, then discrete actions (i.e. changes) are serialized and made into a unified model. This is serialized or multi-version modeling.

In order to enable *Synchronous* and *Asynchronous* collaborative design in VDS, MAVT

uses a Uni-Server to meet the requirements of design activities. By Uni-server it means that no matter the designer's choice for design, he has to connect to the same server. This is basically an advancement of our previous work [22] where we were able to transmit 2D and 3D CAD data over the internet to facilitate collaborative design implementing Java™ Remote Method Invocation (RMI) and Java™ Media Framework (JMF). Such distributed application can be extended to MAVT too to facilitate effective design and collaboration between geographically dispersed users. Distributed applications developed for this module are mainly focused on collaborative design in virtual environment. In order to overcome geographical and time constraints, one fast and easy solution is the utilization of the Internet. Right from the concept analysis stage, integrated networking capability was considered as a major feature of such application. An integrated networking capability also strongly supports the cause of platform independence because it does not rely on the operating system network communication module.

The various concepts of groupware that are reviewed and analyzed meticulously and later attempted to implement for developing VDS for synchronous collaboration through a mapping window (same time–different place collaboration) and for asynchronous collaboration by saving an image files in the local system (for same place–different time collaboration) and then transmitting the image to remote users using electronic mailing system (different time–different place collaboration). The success in this issue opens a new arena in future collaborative groupware research.

The basic structure of a Uni-Server of VDS built for basic collaboration purpose. The Uni-Server holds the MAVT application and it has a storage system and a fixed IP address for clients to access. One thread of application is representative of one project. Multi-threads are allowed for the parallel CAD environment. Parallel projects can be performed concurrently. Designers can connect to server as clients and a particular designer can access different data storages depending on the project group that he is logged into. Once the designer is logged onto the server, the current server, Simulation-Universe, will display on the client's machine the relevant design components.

The designer can freely obtain the design components from the Server Simulation-Universe to Local Simulation-Universe to do the modification using the Drag & Drop functionality. As a security measure, VDS also enables the designer to lock the particular design on the server side when he is in the process of modifying it. Upon completion of the modifications (or creations), the designer can then update the design component from the client to the server side. The finished component is then stored on the server awaiting feedback from other designers, the project manager and other involved parties (such as marketing and manufacturing). In short, design creation and modification is carried out on the local client side while storage and display is carried out on the server side.
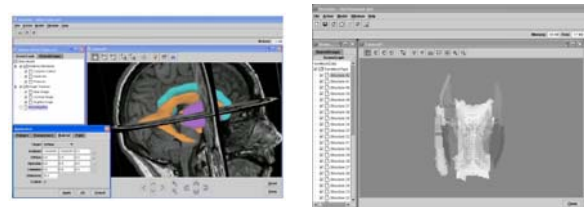

Figure 6: The GUI of MAVT.

A centralized system keeps a single copy of the shared application on a central server and the GUI part is shared among several clients over the network. The application used direct manipulation GUI which has been proved to be suitable for education and simulation [23]. The GUI components are designed to achieve the direct manipulation functionalities including direct manipulation of visual effects (such as lighting, model appearance etc.), functions for association of topology and anatomy information, manipulation of anatomy structure with multi-modal data models, simulation of geometric modeling and physical modeling etc. The GUI of virtual anatomy model in MAVT has been shown in Figure 6.

When large volume of information is to be shared, a centralized architecture may very efficient results. On the other hand, a replicated architecture is based on executing a copy of the application for each user. MAVT uses this replicated structure because, if a large amount of user transaction is necessary, which is the MAVT case, a replicated architecture will be more beneficial. Similarly, there are several kinds of protocol to resolve interpersonal conflicts [21]. MAVT team used a protocol called controller/viewer protocol, which is actually a combination of both the master–slave protocol and token-based protocol [21, 22]. The controller/viewer protocol enables the multiple users to work in a parallel way and thus speed up the entire design process. The network connection was maintained through TCP/IP.

When a collaborative design project uses VDS as the design tool, the first validated login design

partner will create the Virtual Universe. This Virtual Universe remains active in the server until it is closed. Using a user ID and password as registration data, any valid designer is able to login to that Virtual Universe to collaborate with other designers. After logging into the system, an observer may open a Server-Socket (on the server side) that is bound to a specific port number if he wishes to help another person in design. The server just waits, "listening" to the socket for a client to make a connection request. On the client-side, the client controller has to know the hostname of the server and the port number to which the server is connected. To make a connection request, the client has to rendezvous with the server on the server's machine and port. For a successful connection, the server has to first accept the connection. Upon acceptance, a new socket bound to a different port is enabled on the server side. This new socket (consequently a different port number) is needed so that the server can continue to listen to the original socket while tending to the needs of the connected client. On the client side, if the connection is accepted, a socket is successfully created and the client can use the socket to communicate with the server. Note that the socket on the client side is not bound to the port number used to rendezvous with the server. Rather, the client is assigned a port number local to the machine on which the client is running. The client and server can now communicate by writing to or reading from their sockets. Figure 7 gives the flowchart of communication between server and client in VDS.
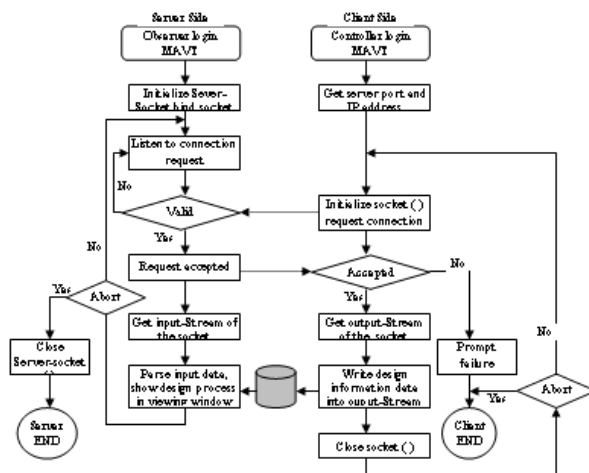

Figure 7: The flowchart of communication between server and client in VDS.

6. Computational Results and Validation

We have developed a framework to build a few multi-modal virtual anatomy models, and used these models to demonstrate their usage in medical education and simulation. Figure 8 shows a multi-modal virtual brain anatomy model. Figure 8(a) and (b) show the brain model consisting of triplanar images and surface models organized together and grouped by their modalities. The grouping operations can be done interactively using the browser window shown in Figure 6. Users can group anatomy models by systematic anatomy or regional anatomy or modality on the fly whenever they like. This will be very useful in medical education for the students to gain anatomy knowledge.
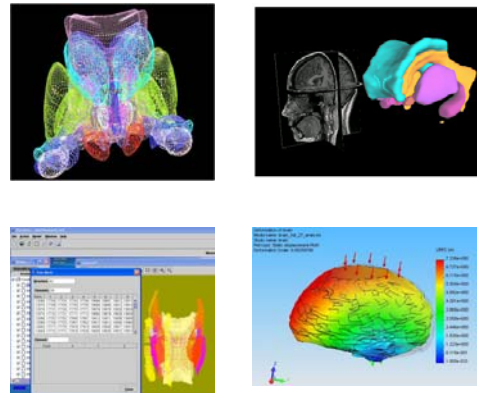

Figure 8: Visualization of MAVT.

Figure 8 (c) shows the FEM meshed model in MAVT. Figure 8 (d) shows a FEM analysis of a brain using the meshed atlas data. Medical students, radiologists, bioengineers and surgeons will be able to manipulate such models for learning the spatial position of the anatomy structures, and for simulating and planning complex surgeries. The direct manipulation GUI allows users to manipulate and control the virtual model easily and efficiently. By manipulating individual objects or individual modality data in these models or by manipulating the combination of these data models, this environment can be used as prototypes of various visualization based or virtual reality based medical applications. These models greatly reduce functional development complexity enabling the user to comfortably collaborate with distant participants.
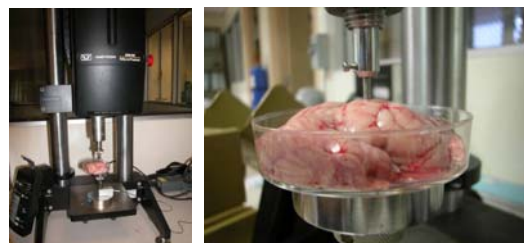

Figure 9. Experiment setup for indention test of porcine brain tissue.

In the current research of CAD modeling, Finite Element modeling is the primary tool used to predict the mechanisms of brain deformation. However, the material properties used in these computational models are crucial for accurately predicting the circumstances that cause deformation. This extensive determination of the mechanical properties of brain tissue at large strains, including the effects of region, direction, age and species, represents a major step forward in our understanding of the response of the brain during traumatic events. As the existing material properties testing were mostly for research on the impact analysis of the human brain where the strain rate is much higher than for surgery applications, very few data were available for surgical biomechanical modeling. As such, we propose to compare and analyze the limited data available in the literature, and furthermore carry out our own experiment to account for the particular neurosurgery situations. The in-vitro indentation experiment was conducted on five adult porcine (pig) brains using the facility in Biomechanics Lab of National University of Singapore (NUS), in accordance with ethical guidelines on animal experiments (Figure 9). The experiment confirmed tissue nonlinearity and inhomogeneity. Our FEM analysis shows human brain is about 32-35% stiffer compared to porcine which agrees with the result (29% stiffer in various gray matter samples) of other researchers [26].
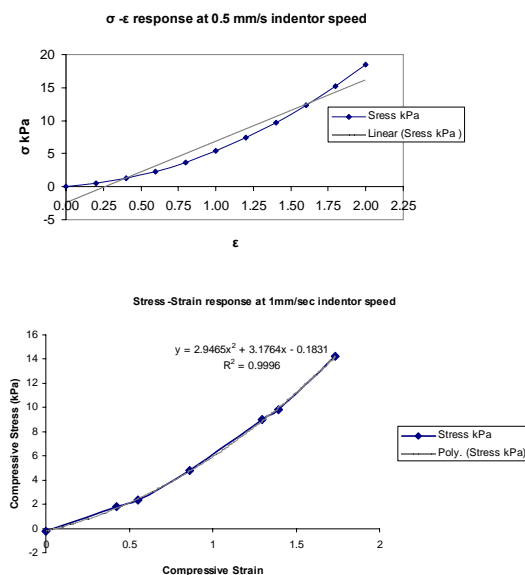

Figure 10: Stress-Srrain relationship for 0.5 mm/s and 1 mm/s indentation speed and 6 mm indentation diameter.

The general behavior of the brain samples under compression showed the general non-linear features typical of the soft tissues. Deformation of brain tissue has been simulated in a scenario where uniform load at 0.05, 0.5 and 1 mm per second indentor speed on the top surface of the brain is applied.

The force predicted by the finite element model based on quasi-liner, hyper-viscoelastic constitutive equation [4, 5] for brain tissue, at the maximum indentation of 4 mm has been compared. The analysis showed that hyper viscoelastic constitutive model was best suited for brain tissue deformation. It has also been observed that brain tissue is best suited with analytical result for the ratio between two hyperelastic polynomial coefficients C_{10} and C_{01} = 0.9: 1. Figure 10 shows the results of the uniaxial compression tests at 0.5 and 1 mm per second indentor speed on the pig brain tissue; it represents the plots of nominal stress versus nominal strain.

7. Discussion and Conclusion

In this paper we proposed a framework for developing a meshed atlas visualization toolkit that can be used for medical study, simulation purposes and other virtual reality applications. A multi-purpose GUI has been designed and developed which can be used for constructing virtual anatomy models that facilitates successful coloration between geographically dispersed users.

Object-oriented methodology has been proven to be a powerful way of organizing and developing software. It mainly involves the organization of a program as a set of components called objects. These objects exist independently of each other and they have rules for communicating with other objects. OOP capability is a major advantage of Java Platform especially in internet programming and development of distributed applications. It has been also proven useful and appropriate in developing computer graphics applications [24]. We choose Java3D as our development platform because it is well adapted to OO concepts. MAVT not only inherits the object-oriented concepts from Java but also makes more flexible and expandable. Meanwhile the Java 3D scene graph hierarchy structure is also convenient in mapping to the hierarchical anatomy structures.

To enable *Synchronous & Asynchronous* collaborative design over a network, platform independence is an important feature. Platform independence is the capability of the same program to work on different operating systems. This is also one of the major reasons for choosing JAVA as the developing platform. as Java is completely platform

independent. JAVA enables "write once, run everywhere", as quoted from JavaSoft[TM] President Alan Baratz "Anything that feels, smells, walks, or talks like it have a processor--we'd like the Java platform to live on it."[25]. There are Java interpreters that can run programs for Microsoft Windows 95/98/2000/XP and NT, Apple Macintosh 7.5, SPARC Solaris 2.3 or higher, Intel x86 Solaris and Lunix which basically covers almost all common design platforms.

MAVT is a distributed application focusing on collaborative design in virtual environment. In order to overcome geographical and time constraints, one fast and easy solution is the utilization of the Internet. We choose XML to present virtual anatomic model definition because of its exceedingly extensibility. We created an XML schema that complies with the data definition as shown in Figure 5. The XML schema is designed to handle more complex anatomy models, for example, centerline tubular model data, etc. that cannot be easily handled by other visualization modeling languages. The implementation of the system uses the MVC (model-view-controller) paradigm to increase the reusability of the code developed. The appropriate design patterns [16] are used as much as possible wherever they are applicable. Java's rich network classes also enable remote file transfer, remote drag and drop and other real time interactions within MAVT so that distant participators can collaborate comfortably with each other. To validate the model, in-vitro indentation experiment has been done and its results analyzed using a sophisticated highly nonlinear, three-dimensional finite element CAD model. The experimental result suggests brain tissue accounts for strong non-linear stress-strain relationship and hyper viscoelastic modeling approach is best suited for such analysis.

8. Acknowledgements

9. References

[1] N. C. Romano, Jr. J. F. Numamaker and R. O Brigg: "User Driven Design of a Web-Based Group Support System", System Sciences, in *Proc. 30th Hawaii Int. Conf. on System Sciences (HICSS)*, vol. 2, 1997, pp. 366 -375

[2] P Dev, C. Friedman, B. Dafoe and R. Felciano, "Testing spatial understanding of anatomy", in *Proc. Annu Symp Comput Appl Med Care*, 1992, pp. 804-805.

[3] P.A. Warrick and W.R.J. Funnell, "VRML: A tool for visualizing anatomy in medical education", in *Proc. 22nd Can. Med. & Biol. Eng. Conf.*, 1996, Charlottetown, pp. 18-19

[4] A Roy, W.L. Nowinski and F. E. H Tay, "An Atlas-Based Approach for Prediction and Analysis of the Brain Deformation", in *Proc. IASTED Int. Conf. on Biomech.*, Honolulu, Hawaii, USA 2004, 162-166.

[5] A. Roy, W.L. Nowinski and F. E. H Tay, "Finite Element Analysis of Brain for Neurosurgical Procedure", *Proc. IBEC2004,1[st] Int. Bioeng. Conf.*, Singapore 2004, 247-250.

[6] K. Miller, "Constitutive model of brain tissue suitable for finite element analysis of surgical procedures", *J Biomech.,* vol. 32, no. 5, 1999, pp. 531- 537.

[7] M. Ferrant, "Physics-based Deformable Modeling of Volumes and Surface for Medical Image Registration, Segmentation and Visualization", Ph.D. dissertation, Louvain, Belgium: Universite' Catholique de Louvain, 2001.

[8] W. L. Nowinski, A. Thirunavuukarasuu, and A.L. Benabid, "The Cerefy Clinical Brain Atlas (on CD ROM)", Thieme Medical Publishers, Inc. 2004 (EBAL specification is available at www.cerefy.com)

[9] W.L. Nowinski, "From research to clinical practice: a Cerefy brain atlas story", *Int. Congress Series*, 1256, CARS 2003, pp. 75-81

[10] W.L. Nowinski, A. Thirunavuukarasuu, I. Volkau, R. Baimuratov, Q. Hu, A. Aziz and S. Huang, "Three-dimensional brain atlas of anatomy and vasculature". *Radiographics*, vol. 25, no. 1, 2005, pp. 263-271.

[11] J. Talairach and P. Tournoux, "Referentially Oriented Cerebral MRI Anatomy. Atlas of Stereotaxic Anatomical Correlations for Gray and White Matter", *Georg Thieme Verlag/Thieme Medical Publishers*, Stuttgart - New York, 1993

[12] W.L. Nowinski and A.L. Benabid, "New directions in atlas-assisted stereotactic functional neurosurgery", In: Advanced Techniques in Image-Guided Brain and Spine Surgery (ed Germano IM), Thieme, New York, 2001.

[13] W.L. Nowinski, D. Belov and A.L. Benabid, "An algorithm for rapid calculation of a probabilistic functional atlas of subcortical structures from electrophysiological data collected during functional neurosurgery procedures", *NeuroImage,* vol. 18, no. 1, 2003, 143-155.

[14] G. Szikla, G. Bouvier, T. Hori and V. Petrov, "Angiography of the Human Brain Cortex: Atlas of Vascular Patterns and Stereotactic Localization", Springer-Verlag, Berlin 1977.

[15] K. Miller, Z. Taylo and W. L. Nowinski, "Towards computing brain deformations for diagnosis, prognosis and neurosurgical simulation", *J. Mech and Med and Biol,* vol. 5, no. 1, 2005, pp 105–121.

[16] V. D. Liseikin, *Grid Generation Methods*, Springer, Berlin   ,   1999.

[17] V. Adams and A. Askenazi, *Building Better Products with Finite Element Analysis,* High Mountain Press, 1998.

[18] E.S Jang, "3D animation coding: its history and framework", in *Proc. IEEE Int. Conf. on Multimedia and Expo*, 2000, pp. 1119–1122.

[19] "SolidWorks 2005 Online User Guide", SolidWorks ® 2005.

[20] "Introducing CosmosWorks: Design analysis made easy" Doc. No. CWMISENG0703, COSMOS™ 2004.

[21] A. Roy, "CyberCAD - A Java™- based synchronized collaborative CAD system", M.Eng dissertation, Dept. Mech. Eng, National University of Singapore, Singapore, 2000.

[22] F. E. H. Tay and A. Roy, "CyberCAD: a collaborative approach in 3D-CAD technology in a multimedia-supported environment", *Comp in Industry,* vol. 52, 2003, 127–145.

[23] B. Schneiderman, "Designing the user interface : strategies for effective human-computer-interaction, Reading", MA: Addison-Wesley, 1998

[24] R. Veltkamp and E. Blake, "Object–Oriented Programming for Graphics" *2nd Eurographics Workshop on Object-oriented Graphics*, Rep. Springer-Verlag, 1995.

[25] Java look and Feel Design Guidelines: http://java.sun.com/products/jlf/dg/index.htm

[26] M. T Prange and S. S. Margulies, Regional, directional and age-dependent properties of brain undergoing large deformation. *ASME, Transl. J. Biomech Eng*, vol. 124, 2002, pp. 244-252.

Avijit Roy received his Master of Eng. degree in Mechanical Engineering from National University of Singapore (NUS) in 2000 and B.Sc from Bangladesh University of Engineering and Technology (BUET) in 1997.

He is now pursuing his PhD at National University of Singapore. He has also been working in the Bioengineering field at Biomedical Imaging Lab, ASTAR in Singapore, and conducting research as a doctoral research fellow. This work includes the development and implementation of computational models of soft tissues, the development of a physics-based finite element human brain model (3D), simulation of the deformation of the brain and its sub-cortical structures, prediction of tumor growth and model of Deep Brain Stimulation (DBS) for surgical treatment of Parkinson's disease and CAD Collaboration. During his Master's project at NUS, he developed CyberCAD - the world's first collaborative CAD software that implemented Java™, Java3D™ and JMF. The software won the first prize in the *One Challenge* competition in Singapore. He has also worked as a Software Engineer for different IT firms.. His first Bengali book *Alo hate choliyache Adharer Jatri* (Publisher: Ankur Prakashani, April: 2005, ISBN : 984 464 124 1) on the creation of the universe from an astrophysics perspective was published in Bangladesh earlier last year. He has several publications in top-tier journals and has presented at many international conferences.

Francis Eng-Hock Tay is an Associate Professor in the Department of Mechanical Engineering of NUS and a Fellow of Institute of Physics (UK) and a Council Member of Institute of Engineering (Singapore).

He received his Ph.D. degree in 1996 from MIT, USA and M. Eng degree from National University of Singapore (NUS) in 1992. In 1998, he was among the Top Ten Finalist in achieving the Outstanding Young Person Award in Singapore. His area of research is MEMS, Design Methods and Design Automation.

Wieslaw L. Nowinski, DSc, PhD is principal scientist and director of the Biomedical Imaging Lab, Agency for Science, Technology and Research, Singapore (ASTAR). He has developed twelve brain atlas products used world-wide in neurosurgery, brain mapping, neuroradiology, and neuroeducation. His research interests include neuroinformatics, brain atlases, computer-aided diagnosis and treatment, virtual reality, modeling, segmentation, registration, and future directions in computer-aided surgery. He is also an affiliate professor in Department of Radiology, University of Washington Medical Center, Seattle, WA; visiting professor and doctoral advisor at the Harbin Institute of Technology, China; editor-in-chief of a book series on Health Informatics; and chairman of the Scientific Advisory Board of a stroke-related, US-based company.