AFRL-RH-FS-TR-2014-0026

EFFICIENT MODEL POSING AND MORPHING SOFTWARE

Stephen Aylward
Julien Finet

Kitware, Inc.
28 Corporate Drive
Clifton Park, NY 12065-8662

Final Report

April 2014

Air Force Research Laboratory
711th Human Performance Wing
Human Effectiveness Directorate
Bioeffects Division
Radio Frequency Bioeffects Branch
4141 Petroleum Road
JBSA Fort Sam Houston, Texas 78234

NOTICE AND SIGNATURE PAGE

AFRL-RH-BR-TR-2014-0026, "Efficient Model Posing and Morphing Software", HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

___//signed//_________________________________
Program Manager
Radio Frequency Bioeffects Branch

___//signed//_________________________________
GARRETT D. POLHAMUS, DR-IV, DAF
Chief, Bioeffects Division

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
03-04-2014

2. REPORT TYPE
Final Technical Report

3. DATES COVERED *(From - To)*
July 1, 2013 - April 3, 2014

4. TITLE AND SUBTITLE

Efficient Model Posing and Morphing Software

5a. CONTRACT NUMBER
FA8650-13-M-6444

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
62202F

6. AUTHOR(S)

Aylward, Stephen; Finet, Julien

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
H0F7

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Kitware, Inc.
28 Corporate Drive
Clifton Park, NY 12065-8662

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Materiel Command, Air Force Research Laboratory
711th Human Performance Wing, Human Effectiveness Directorate,
Bioeffects Division, Radio Frequency Bioeffects Branch
4141 Petroleum Road
JBSA Fort Sam Houston, Texas 78234-2644

10. SPONSOR/MONITOR'S ACRONYM(S)

AFRL-RH-FS-TR-2014-0026

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
0001AD

12. DISTRIBUTION / AVAILABILITY STATEMENT

Distribution Statement A: Approved for public release, distribution unlimited. Other requests for this document shall be referred to 711 HPW/RHD, Jason Payne, 4141 Petroleum Road, JBSA Fort Sam Houston, Texas 78234-2644.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The absorption of electromagnetic energy within human tissue depends upon anatomical posture and body type, e.g., body-mass index. Creating a large number of anatomical models with various body postures is problematic however, given medical scanner costs and acquisition requirements, e.g., the subject must be lying down for CT and MRI acquisitions.

We propose to deliver software, documentation, and examples relating to our research aimed at modifying the anatomical pose and body morphology (e.g., body-mass indexes) of volumetric, voxelized, anatomical models. In particular, we propose to adapt real-time surgical simulation methods to serve as the underlying methods in changing the pose and the fat/muscle composition of voxel models. This approach has the key benefit of being able to generate morphed voxelized anatomical models in less than 10 minutes. Intuitive software applications that incorporate these algorithms are already being prototyped and will be extended, evaluated, and delivered in Phase I.

This project builds upon significant prior work at Kitware and makes use of several open-source, image-processing toolkits. The product will be offered as open-source software and used to attract additional consulting clients to Kitware.

15. SUBJECT TERMS

Cross Platform Chemistry Application Unifying Structure Manipulation, External Tools, Databases and Visualization

16. SECURITY CLASSIFICATION OF: Unclassified			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** J. Payne
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** *(include area code)*
U	U	U	UU	21	

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

Distribution A: Approved for public release, distribution unlimited, TRSL-PA-2014-0075, August 20, 2014.

LIST OF FIGURES

1.0 SUMMARY

This report details the work accomplished during a Phase I SBIR effort titled "Efficient Model Posing and Morphing Software." During this SBIR effort, Kitware researched algorithms and modified and extended existing software to enable efficient posing and morphing of digital anatomical models. These models are typically used within high fidelity electromagnetic (EM) and thermal simulation software to characterize the absorption of radio frequency (RF) energy within tissues and the associated temperature response. The results from these simulations may vary across anatomical poses and body types, e.g. for different body-mass indexes. Acquiring anatomical body models in various postures and for various body types can be problematic due to medical scanner costs, post-processing labor efforts and acquisition requirements. Typically, subjects must be lying down during x-ray computed tomography (CT) and magnetic resonance imaging (MRI) acquisitions, and therefore anatomical image datasets are mostly constrained to this posture. During this Phase I effort, we developed techniques to extend Bender, our existing anatomic model manipulation software, in order to mitigate the need for new models by enabling model posing and morphing through software.

The long-term goal of this project is to develop and deliver software, documentation, and working examples for (a) manipulating the pose of and (b) simulating anthropomorphic changes (e.g., body-mass index (BMI) changes) in voxelized anatomical models. In particular, we intend to significantly shorten the computational runtime required to repose a voxel model. Specifically, given a re-posed skeleton or a new target BMI value, the corresponding resampled voxel model should be computed in 10 minutes or less.

The pose manipulation method we chose is based on controlling the relative rotations of bones in an anatomical model. To improve our existing voxel model manipulation software ("Bender"), the following long-term requirements needed to be addressed:

- Provide an intuitive interface for anatomic model repositioning that incorporates standard joint types (i.e., ball and socket, elbow, gliding, hinge, hip, and saddle joints) and associated ranges of motion
- Integrate multiple data sources of varying fidelity such that models acquired at different resolutions can be merged into a single model
- Make use of existing, freely available, massive libraries of pose and motion data

The method we selected for manipulating anthropomorphic measures**,** such as BMI, is based on prescribing regional changes in fat, organ, and muscle volumes. This method should also be incorporated into our "Bender" software with the following requirements:

- An organ (e.g., liver) can be selected and assigned a new volume, and it and its surrounding soft tissues will then be scaled and morphed accordingly
- Regional fat and/or muscle volumes can be increased or decreased, and they and the surrounding soft tissues will be scaled and morphed accordingly

Finally, the voxelized model resampling method is driven by the user-specified changes in pose and anthropomorphic measures. Leveraging the graphics process unit (GPU), it has the following requirements:

- Use real-time, GPU-based surgical simulation methods to resample a high-fidelity voxelized anatomical model into a new pose and anthropomorphic morphology in less than 10 minutes.
- Provide anatomically valid models even when large movements are made

• Include hand-editing tools that can be used to correct invalid anatomic configurations that may arise after motion simulation.

2.0 INTRODUCTION

The goal of the Phase I effort was to research and begin development of methodologies to address the requirements listed above. In particular, we sought to improve the rigging manipulation within Bender and to incorporate a surgical simulation toolkit to improve the resampling of voxelized anatomical models. The tasks we pursued were as follows:

1. Make use of existing, freely available, massive libraries of pose and motion data.
2. Support the specification of changes in the regional fat and/or muscle volumes of a model so that the BMI of the model can be changed.
3. Integrate surgical simulation methods to resample voxelized anatomical models.

Bender is a freely-available, open-source software for the manipulation of voxelized anatomical models. Bender allows an operator to specify a rigging that represents the anatomical pose of an existing voxel model, manipulate that rigging into a different anatomical pose, and then generate a new voxelized model that represents the original model resampled into that new position. Bender is built on top of 3D Slicer, a free open-source software package for visualization and image analysis. Slicer covers a large range of modality imaging, provides solutions for multiple organs, offers bidirectional interface for devices and is expandable and interfaced to multiple toolkits such as the Insight Segmentation and Registration Toolkit (ITK) or the Visualization Toolkit (VTK).

Bender has the following workflow:

1. Rigging the voxelized model: This involves specifying a skeleton ("Rigs") that represents the linear sections and joints of the body, by which the body will be repositioned.
2. Painting regions of interest (ROIs) per Rig: This is a 3D painting process in which the bones, soft tissues, and skin that should be moved with each rig section are explicitly associated with each section.
3. Rig manipulation: The rigging is bent and rotated at its joints to define the target repositioning of the body.
4. Resampling: The bones, soft tissues, and skin in the voxelized model are resampled (via "Skinning") onto the repositioned rigging to create a new voxelized model.

3.0 INCORPORATING EXISTING MOTION-CAPTURE DATA LIBRARIES

3.1. Method

User interaction is the largest consumer of time and the main source of error in re-posing voxelized models. Probably the most challenging user interaction task is specifying a pose that is mechanically possible and that looks natural.

The solution we developed during Phase I was to use databases of defined poses and actions that have been published on the Internet. The poses in these databases are typically specified using the Biovision hierarchy (BVH) file format. Intended for use in computer graphics applications, this format defines a rig that represents the skeletal structure of the model and defines poses and sequences of poses as rotations at each joint in the rig.

The solution we implemented added the ability to read BVH files in Bender, fit the rig in a BVH file to the voxelized model, and skin the entire voxelized model (not only its surface) to that rigging. As a result, any BVH file that uses that same rigging to specify a new pose can be used to re-pose the voxelized model.

Carnegie Mellon University and others have released over 2500 BVH files that correspond to a wide range of human motions; many of these were created using motion capture devices. In order to leverage the available motion capture database content to ease the posing step, the following tasks have been performed:

- Task 1.a: Load a standard BVH pose file
- Task 1.b: Map standard BVH rig skeleton segments and joints in a Bender rig.
- Task 1.c: Move a Bender rig based on the movement of a loaded BVH rig

Note that Bender rigs are typically more detailed than those in a BVH pose file (e.g., Bender models have arbitrarily detailed rigging while BVH models have pre-defined, modest-fidelity rigs).

3.2. Results

In order to load BVH motion capture files into Bender, we first added the ability to read the file. After analyzing the file format, we implemented a C++ reader that parses and converts BVH files into a VTK compatible structure for rigs and animation (Task 1.a) (see Figure 1.)

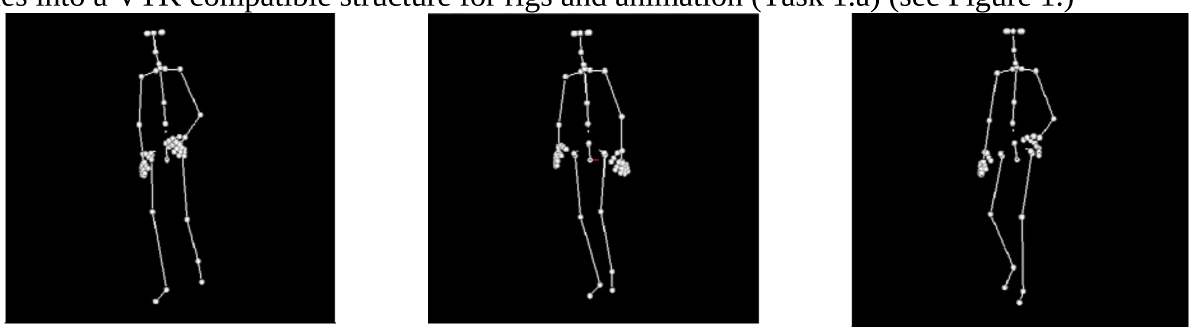


Figure 1. Three steps of a walking BVH animation file read by our BVH reader and visualized with VTK

We then added support into Bender to apply a pose to a rig without changing the topology of the rig. We also created custom dialogs to select a specific frame from the BVH file. The translations and rotations at each joint are then applied to the matching armature rig joints (Task 1.c). The user workflow is presented in Figure 2.

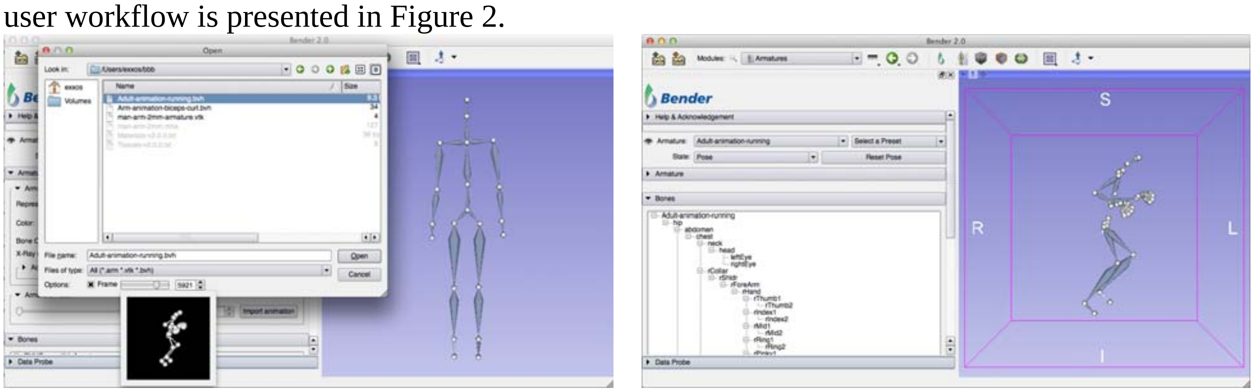


Figure 2. A motion-capture file is being loaded in Bender.

Once an armature has been rigged onto a subject, a BVH file containing a sequence of poses can be loaded. A specific frame is selected and applied to the armature rig

Importing a BVH pose into an existing rig currently works only if the number of bones is the same between the pose and the rig. Future proposed work will add a rig registration technique that will allow importing poses with different topology. Semi-automated methods controlled by the user would associate the joints and connectors in one rig with those in another. Many-to-one and one-to-many mappings would then be supported.

4.0 MORPHING OF ANATOMICAL MODELS

4.1. Method

In order to enable the creation of a representative variety of body types, Kitware plans to provide semi-automatic tools for adjusting the size of organs and the portion of fat in a voxel model. To support these tools, the user interface of Bender required the following changes, which were implemented during Phase I:

- Task 2.a: Allow the user to specify a volume-of-interest in a voxelized model
- Task 2.b: Compute the current volume of fat and muscle in that volume of interest.

As stated in the Phase I proposal, the manipulation of the muscle and fat content in a model was not to be implemented in Phase I. The manipulation task would be implemented in Phase II, after we proved that the finite element method (FEM) technique is a viable solution during Phase I (see Task 3). The Phase II plan is to achieve a user-specified fat/muscle ratio for the chosen volume of interest by placing expansion/contraction forces on the FEM models of the muscle and fat and then running the voxelized model deformation algorithms until the target volumes are achieved.

4.2. Results

The first task consisted of defining a region of interest (ROI) in the voxelized model (Task 2.a). We adapted a region of interest tool from the Slicer library so that the user can place a ROI box within a 3D view of the anatomy. This box may then be modified by pulling the handles of the box in any view (Figure 3.)

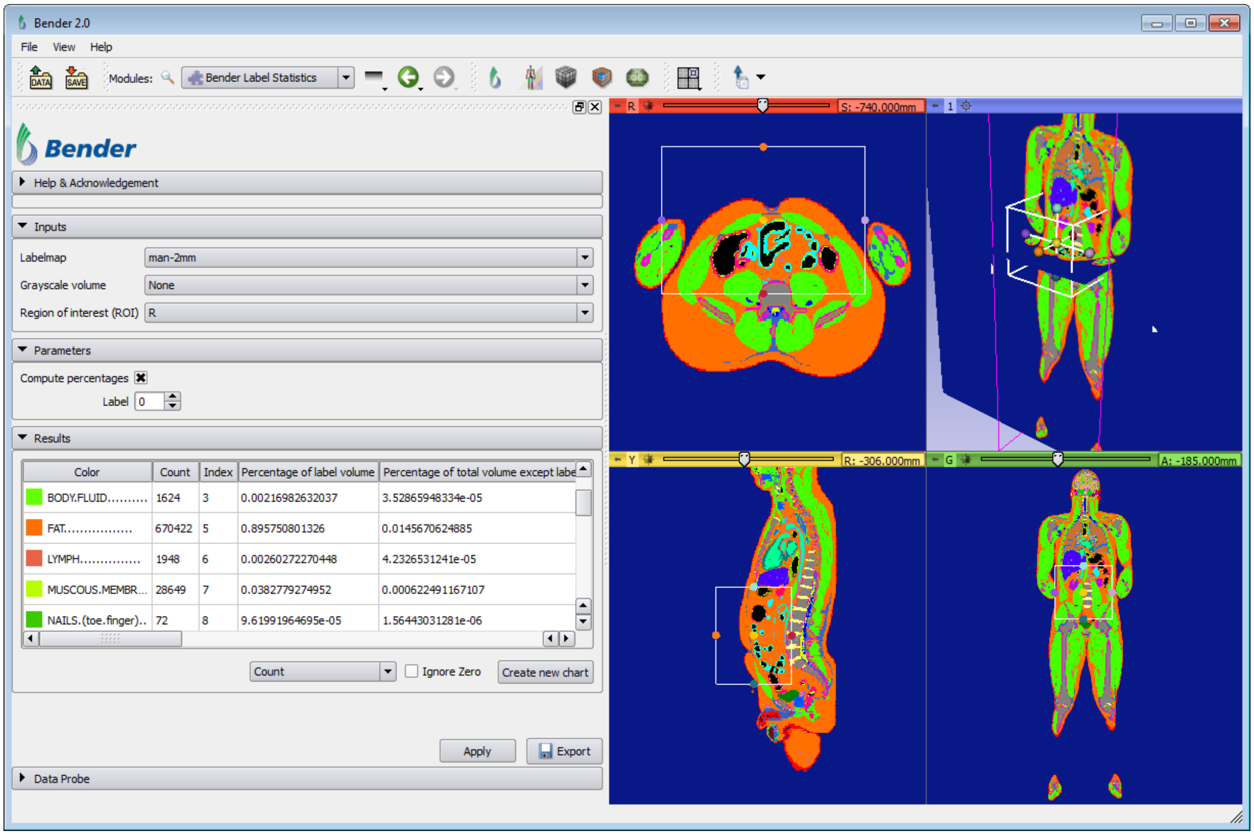


Figure 3. Extended label statistics module in Bender.
For each label, the module counts how many voxels are found in a user specified ROI and computes the ratio over the total fat voxel count

The second task was the computation of label statistics within a previously defined ROI (Task 2.b). We extended the Label Statistics module from Slicer to constrain the computation of the statistics within the ROI. We also exposed widgets in the module panel to conveniently compute the ratio between two labels. With these changes, it is now possible to compute the fat/muscle ratio in the belly area of a voxelized anatomical model (see Figure 3). We have also developed the interface to specify target fat / muscle ratios in a volume of interest; however, the underlying algorithms are proposed for Phase II development.

5.0 ADVANCED ANATOMICAL MODEL RESAMPLING TECHNIQUE

5.1. Method

Bender incorporates advanced methods for resampling voxelized models that are based on rigging and skinning techniques. Rigging and skinning were originally developed for computer graphics applications involving skins/surfaces. We previously adapted this rigging and skinning technique to work on 3D voxelized models. However, the realism of a deformed model depends on the accurate modeling of tissue deformation properties.

In prior work, we modified Bender to use Dual Quaternion (DQ) instead of linear interpolation techniques for tissue deformation approximation. Indeed, linear interpolation of transformed points does not preserve distances whereas Quaternion Spherical Linear Interpolation (Slerp) ensures constant-speed motion and increases the realism of the estimated deformation (see Figure 4.)

However, when using the Dual Quaternion technique alone, unrealistic tissue deformations may still result. For example, the arm and forearm may self-intersect when the arm is bent sharply (see Figure 4.)

To overcome the limitations in the prior approach, during Phase I we devised and implemented a Finite Element Method (FEM) with tissue-specific deformation properties to improve the realism of simulated deformations. FEM is a numerical technique for finding approximate solutions to boundary value problems for differential equations. Just as many tiny straight lines can approximate a larger circle, FEM combines many simple equations over many small subdomains to approximate a more complex equation over a larger domain. The divided approach has several advantages. It can accurately represent complex geometry, include dissimilar material properties, simply represent the total solution and capture local effects.

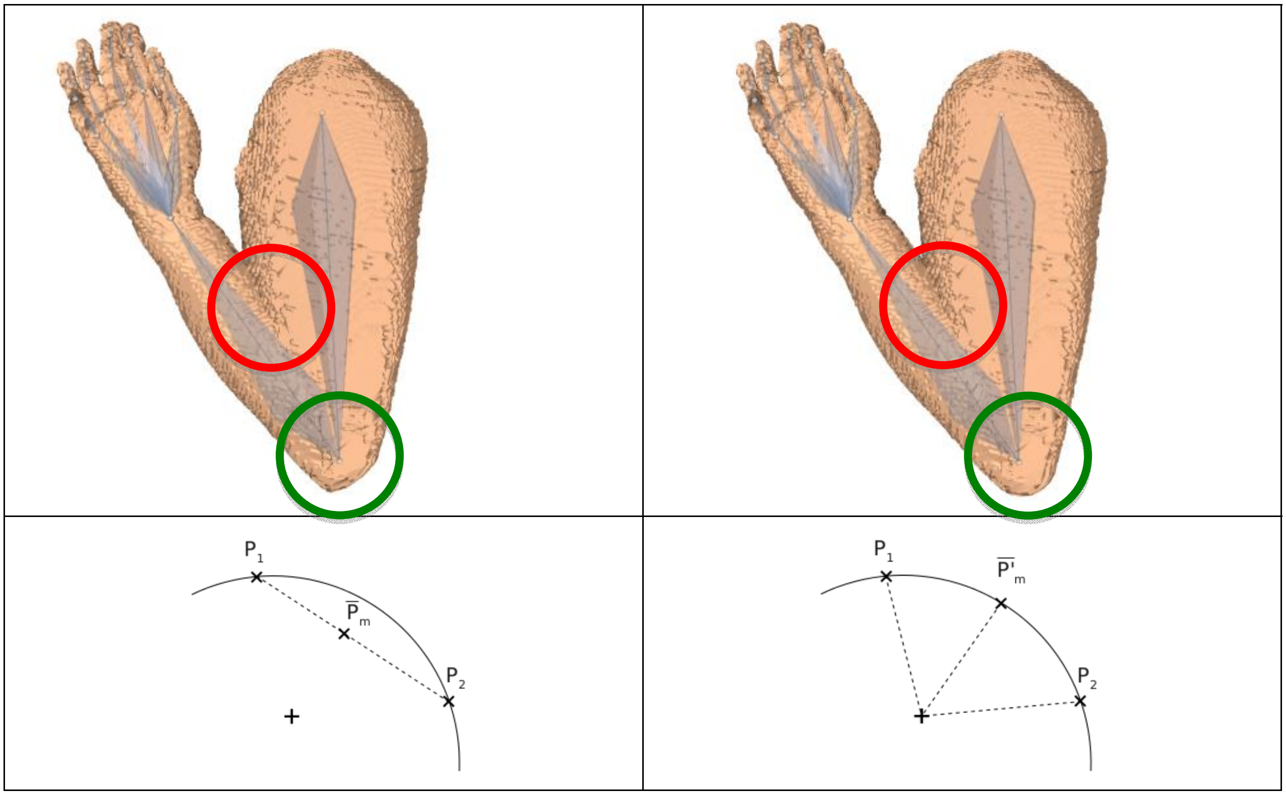


Figure 4: Comparison between linear (left) and dual quaternion (right) interpolation techniques. While volume is preserved in the elbow area with DQ interpolation, volume is lost in both cases in the forearm-arm junction due to lack of collision detection and force response

For Phase I, the FEM-based solution was implemented by integrating the Simulation Open Framework Architecture (SOFA) toolkit[3] into Bender. SOFA is a very mature and open-source framework that is supported by a large community. It consists of more than 250,000 lines of code and it has been downloaded more than 20,000 times. SOFA offers a full FEM support with a complete physics engine. Notably, it is being used for surgery simulation (see Figure 5).

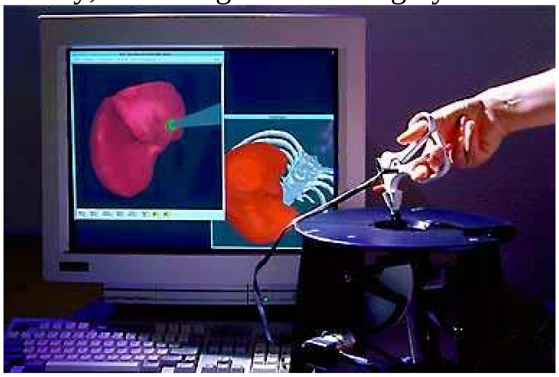


Figure 5. SOFA is used to drive interactive manipulations of 3D anatomical models for surgical simulation using force-feedback devices

The challenge with FEM models resides in the large number of parameters they potentially introduce. Various physical properties of each tissue type must be defined. While others have conducted numerous studies to determine these values, a significantly time consuming task remains to fine-tune those parameters to improve the realism of the results. During Phase I, we focused on incorporating the FEM technique into Bender, leaving further parameter refinements for the proposed Phase II effort.

In order to integrate real-time, surgical simulation methods into Bender, the following sub-tasks have been identified and implemented:

- Task 3.a: Integrate SOFA with Bender
- Task 3.b: Investigate FEM mesh generation methods in SOFA for representing anatomy in voxelized anatomic models
- Task 3.c: Assign stiffness properties to tissues in the FEM
  Task 3.d: Implement rigging movement as forces on the FEM

5.2. Results

The FEM physics engine we selected is the SOFA toolkit. In order to integrate the toolkit with Bender (Task 3.a), we used the External Project feature of the CMake build system to download, configure and compile the library as part of the Bender build process. During that process, we solved compilation and configuration errors in SOFA, and we contributed those fixes back to the SOFA developers, to the benefit of the large number of SOFA users around the world. We were then able to use all of the SOFA capabilities using its C++ API.

SOFA, and more particularly the FEM method it employs, requires a tetrahedral mesh to run physical simulations (Task 3.b). The voxelized anatomical model must therefore be converted into a tetrahedral mesh. To do so, we used the open-source Cleaver library to perform the conversion. Cleaver[4] is a small C++ software library that generates conforming tetrahedral meshes of multi-material volumetric data. Similarly to SOFA, we added the Cleaver library as part of the Bender build process.

For meshing simplification and for faster FEM simulation, the voxelized model should be made of a limited number of materials. Due to the similar tissue properties of most of the human organs, we selected 4 materials by default: air, bone, skin and muscle. A preprocessing step merges the organs into the 4 categories of organs (see Figure 6).

Moreover, due to the memory usage of the Cleaver library and the complexity of the FEM simulation, the input voxelized model must be resampled to a coarser model. We used a voting-based resampling technique to keep the important organ topology during the resampling interpolation.

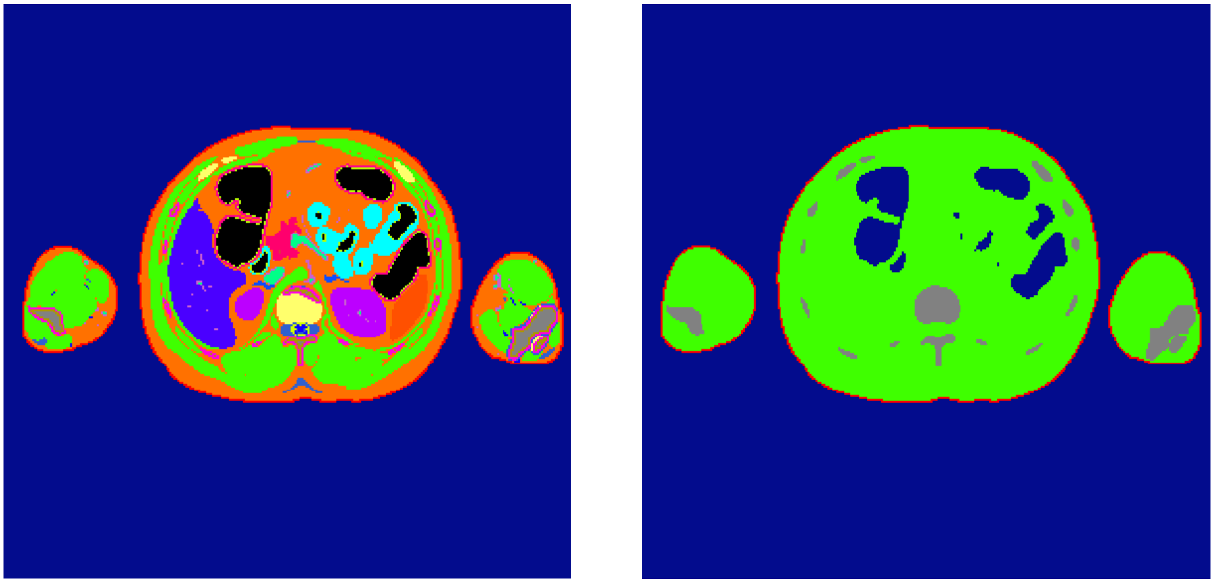


Figure 6: Input (to the left) and output (to the right) of the preprocessing filter that merges multiple organs into 4 materials: air, bone, skin and muscle

Once the anatomical voxelized model is simplified, a Cleaver based command line interface program is executed within Bender to generate the tetrahedral mesh (see Figure 7). The output mesh is then loaded and visualized into Bender (Task 3.b). Figure 8 presents the generated tetrahedral mesh inside Bender.

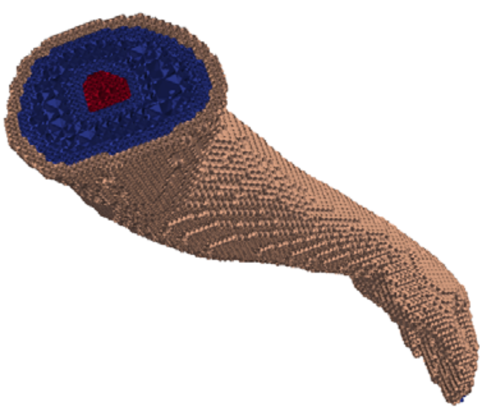


Figure 7. Cross sectional view of the tetrahedral mesh generated by the cleaver library

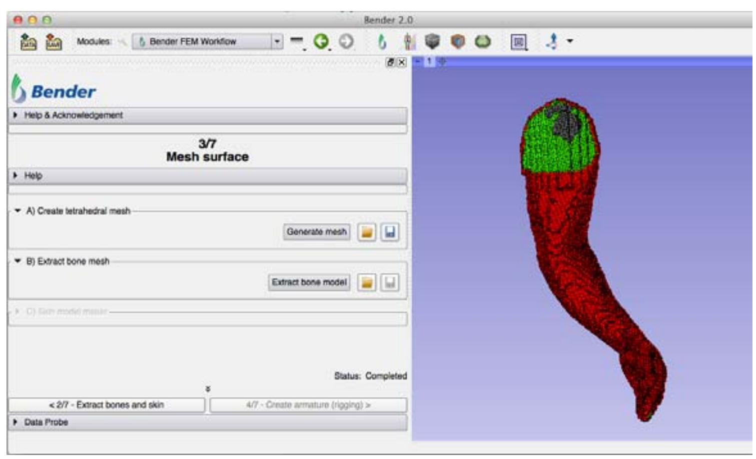


Figure 8. Generated tetrahedral mesh loaded into Bender

The manual rigging step is more conveniently performed in 3D with only the bones visible. To ease the process, we extracted the bones from the generated tetrahedral mesh. The skin surface is also helpful when rigging, and as is explained below, the outer skin of the mesh is also used for collision detection (see Figure 9).

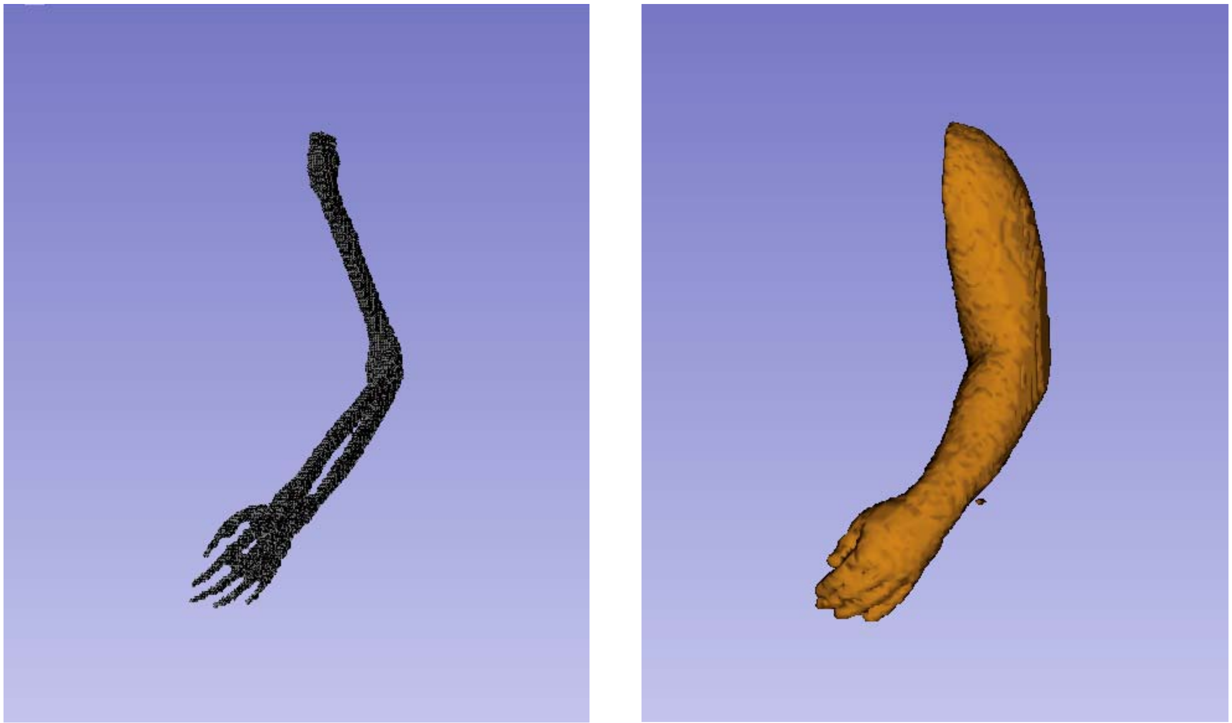


Figure 9. Extracted tetrahedral mesh of the bones (left) and computed surface mesh of the skin(right)

Similarly to the non-FEM workflow, the armature is then rigged and skinned onto the voxelized labelmap. The labelmap weights are then applied to the corresponding cells of the tetrahedral mesh along with the material properties, such as Young modulus and Poisson ratio (Task 3.c). Those material properties are defined in a space-separated-value file that can be edited by the user to tweak the parameters. As part of the workflow, the parameter file is read from disk and material properties are assigned to each cell. Finally the user poses the armature (manually or using BVH) and the tetrahedral mesh can be transformed.

In order to transform the tetrahedral mesh, forces must be applied to the FEM model of the mesh. The main challenge is to apply the correct forces at the right time and place. Our first approach was to map the transformed rig segments to each cell of the skin surface mesh. However this solution failed to detect collisions because no force was generated in the process. To correct this, we changed our approach in order to move the cells of the mesh by applying forces instead of moving the cells directly.

Because bones move rigidly (no internal deformation), we can define forces only on the tetrahedral cells of the bones. Due to physical constraints, the other cells of the mesh deform to follow and adhere to the bone cells. The forces applied on the bone cells are spring forces that connect the original bone cells to the final pose of each bone cells (see Figure 10); the final pose being computed using the weights and dual quaternion transforms (Task 3.d) similarly to the non-FEM method.

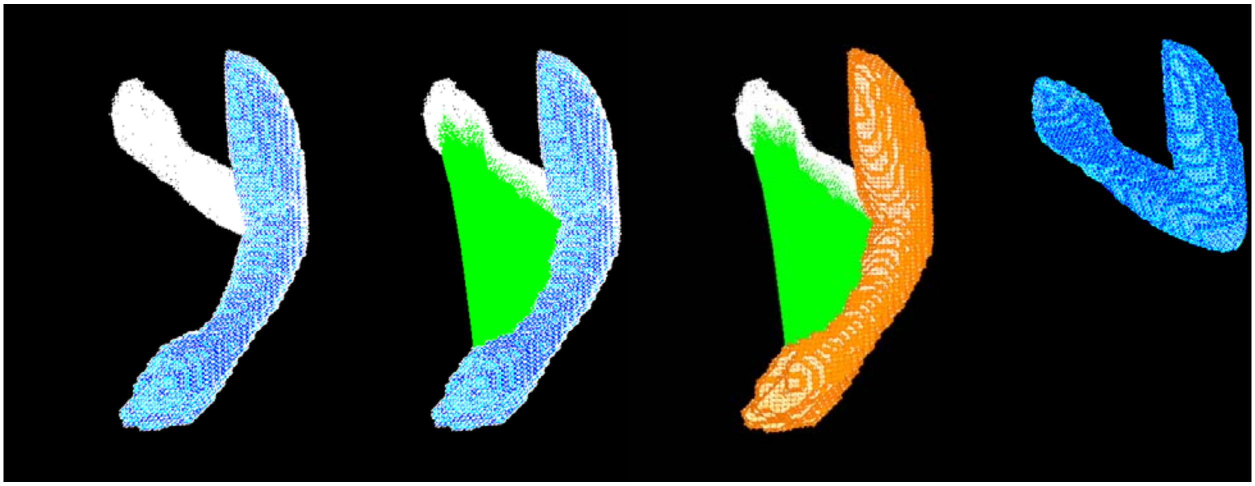
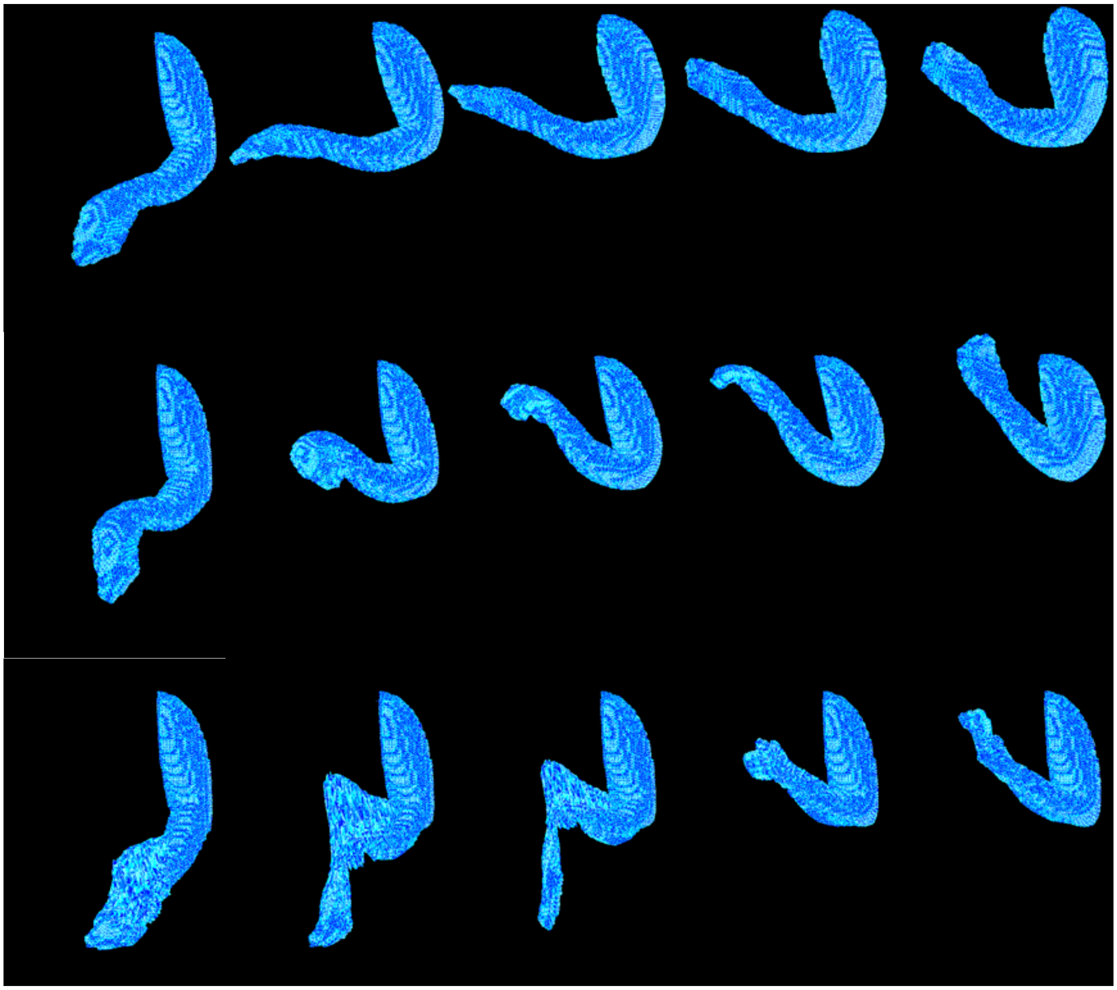


Figure 10. FEM simulation using SOFA.
(FEM force field in blue, pre-computed final pose in white,
spring forces in green and collision surface model in orange)

The behavior of the cells attached to the bone cells is controlled by multiple material parameters, two of them, the Young modulus and the Poisson ratio, control the viscosity and elasticity of the cells, respectively. Figure 11 presents different results when the Young's modulus coefficient is being varied. Please note that the hand fingers are not labeled with bone material in the simulation.



…

…

…

Figure 11. Comparison of a bended arm simulation when using different Young modulus parameter (1,000,000 in top row, 100,000 in middle row and 1,000 in bottom row)

The finite element method approach relies on forces. External spring forces are supported but internal forces are necessary to ensure that the original shape and total volume are preserved. One type of internal force is the response generated by collisions. Collisions prevent self-penetration of the FEM model. Figure 12 demonstrates self-collision in the area inside of the elbow when the forearm is folded onto the arm. Figure 13 shows the final transformed tetrahedral mesh with 2 labels.

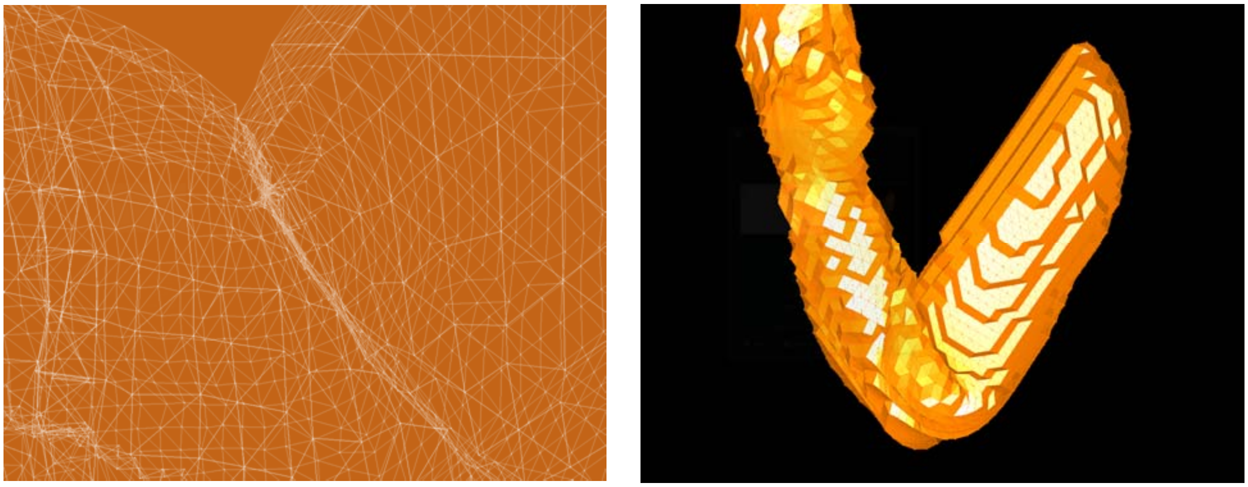


Figure 12. Detail of collision detection inside the elbow area

Figure 13. Volume rendering of an arm tetrahedral mesh with 2 Materials (left) and the mesh posed within Bender using Sofa (right)

It can be noted that our solution for Task 3.b is different than we discussed in our proposal. Instead of using SOFA for generating the tetrahedral mesh, we used the Cleaver library. The motivation behind this choice was the lack of multi-material support in the Sofa mesher. Multi-material meshes are required to use different material properties (such as stiffness) according to the anatomy.

The parameter selection to achieve optimal results can be a challenge. Nonetheless, we were able to find material properties that reproduce natural and anatomically correct behaviors.

Finally, as we will describe next, we managed to implement acceptable processing time for computationally intensive simulations. Additional techniques are available in SOFA to further decrease the simulation processing time.

5.3. Quantification

In order to evaluate our solutions, we quantified the speed and memory requirements for model posing as shown in Table 1.

Table 1. Time & memory analysis of the most demanding steps in the FEM workflow.

Dataset	Dimensions in voxels(v)	Create Tetrahedral Mesh		Rigging	Posing	Posing with BVH	Pose Tetrahedral Mesh	
		Time	Memory	Time	Time	Time	Time	Memory
Arm in 8mm	36*46*87	4s	138MB	~1200s	~400s	60s	1008s	550MB
Requirements	N	91120v/s	2034v/MB	>>1200s	>>400s	60s	143v/s	262v/MB

The *Requirements* row presents the time that would be required for posing a model based on its voxel count. By incorporating motion-capture data via BVH files, we have significantly reduced the user interaction time required to achieve realistic anatomical poses. Skinning and rigging are both one-time tasks for a given voxel model, and once completed, new poses can be easily generated in ~10 minutes, which was the proposed target runtime.

Even faster model posing should be possible via GPU processing. The use of SOFA provided excellent interactivity, but currently we are only using CPU-based computations. SOFA includes GPU-based methods, and we anticipate approximately a 4x speedup when they are enabled.

6.0 . CONCLUSIONS

Our proposed solution of applying a FEM method to the rigging & skinning technique has produced results which meet or exceed our Phase I expectations. The anatomical voxelized models are being transformed with more anatomically correct poses by avoiding self-intersection of tissues using computer graphics' collision detection algorithms. Additionally, the import of motion capture BVH files has yielded major time and effort reductions. With BVH being used in multiple, free, online databases covering almost every possible human and animal pose, almost no major alteration of any pose is needed. Furthermore, we paved the road for enabling anthropological measure manipulation by enabling the definition of a region of interest around the organs to be altered. Building upon our FEM implementation, we will be able to apply anthropomorphic changes to measures such as BMI in order to generate new models.

As a result of this Phase I, we released version 2.0 of our Bender software. Documentation and tutorials have been written to guide the use of the application. Further information concerning the software is available at http://public.kitware.com/Wiki/Bender.

7.0 REFERENCES

Wikipedia, “Skeletal animation”, URL http://en.wikipedia.org/wiki/Skeletal_animation, last modified February 27, 2014. Accessed March 19, 2014

Wu X., et al. “Integration of a Multigrid ODE Solver into an Open Medical Simulation Framework”, IEEE EMBS Conference, August 2012, pp. 3090-3093

Allard J., Cotin S., Faure F., Bensoussan P.-J., Poyer F., Duriez C., Delingette H. and Grisoni L., “SOFA – an Open Source Framework for Medical Simulation”, *Medicine Meets Virtual Reality (MMVR'15)*, pp. 13-18, 2007

Bronson J., Levine, J., Whitaker R., “Lattice Cleaving: Conforming Tetrahedral Meshes of Multimaterial Domains with Bounded Quality”, *Proceedings of the 21st International Meshing Roundtable*, San Jose, CA, Oct 7-10, 2012

LIST OF SYMBOLS, ABBREVIATIONS, AND ACRONYMS

BVH	biovision hierarchy
CLI	Command-Line Interface
CT	computed tomography
DQ	dual quaternion
EM	electromagnetic
FEM	finite element method
GPU	graphics processing unit
GUI	graphical user interface
ITK	Insight Segmentation and Registration Toolkit
MRI	magnetic resonance imaging
MRML	Medical Reality Markup Language
NIH	National Institutes of Health
RF	radio frequency
ROI	Region of Interest
SAR	specific absorption rate
SOFA	Simulation Open Framework Architecture
VTK	The Visualization Toolkit