



MICCAI 2007 Workshop Proceedings

Computational Biomechanics for Medicine II

Karol Miller, Keith D. Paulsen, Alistair A. Young, Poul M.F. Nielsen

<http://cbm2007.mech.uwa.edu.au>



Preface:

A novel partnership between surgeons and machines, made possible by advances in computing and engineering technology, could overcome many of the limitations of traditional surgery. By extending surgeons' ability to plan and carry out surgical interventions more accurately and with less trauma, Computer-Integrated Surgery (CIS) systems could help to improve clinical outcomes and the efficiency of health care delivery. CIS systems could have a similar impact on surgery to that long since realized in Computer-Integrated Manufacturing (CIM). Mathematical modeling and computer simulation have proved tremendously successful in engineering. Computational mechanics has enabled technological developments in virtually every area of our lives. One of the greatest challenges for mechanists is to extend the success of computational mechanics to fields outside traditional engineering, in particular to biology, the biomedical sciences, and medicine.

Computational Biomechanics for Medicine Workshop series was established in 2006 with the first meeting held in Copenhagen. The second workshop was held in conjunction with the Medical Image Computing and Computer Assisted Intervention Conference (MICCAI 2007) in Brisbane on 29 October 2007. It provided an opportunity for specialists in computational sciences to present and exchange opinions on the possibilities of applying their techniques to computer-integrated medicine.

Computational Biomechanics for Medicine II was organized into two streams: Computational Solid Mechanics, and Computational Fluid Mechanics and Thermodynamics. The application of advanced computational methods to the following areas was discussed:

- Medical image analysis;
- Image-guided surgery;
- Surgical simulation;
- Surgical intervention planning;
- Disease prognosis and diagnosis;
- Injury mechanism analysis;
- Implant and prostheses design;
- Medical robotics.

We received many more submissions than we could accommodate in a one-day workshop. After rigorous review of full (six-to-ten page) manuscripts we accepted 16 papers, collected in this volume. They were split equally between podium and poster presentations. The proceedings also include abstracts of two invited lectures by world-leading researchers Professors Peter Hunter and Dimitris Metaxas.

Information about Computational Biomechanics for Medicine Workshops, including Proceedings of previous meetings is available at <http://cbm.mech.uwa.edu.au/>

We would like to thank the MICCAI 2007 organizers for help with administering the Workshop, invited lecturers for deep insights into their research fields, the authors for submitting high quality work, and the reviewers for helping with paper selection.

Karol Miller
Keith D. Paulsen
Alistair A. Young
Poul M.F. Nielsen

Contents:

Invited Lectures

- | | | |
|---|---|---|
| 1 | A model sharing infrastructure for computational physiology
<i>Peter Hunter</i> | 2 |
| 2 | Integration of Multiple Imaging Data for improved Volumetric Cardiac Motion Analysis
<i>Dimitris Metaxas</i> | 3 |

Part 1. Computational Solid Mechanics

- | | | |
|---|--|----|
| 3 | Physiological Integration of Structural and Functional Cardiac Magnetic Resonance Imaging Using Finite Element Modelling
<i>Hoi Ieng Lam, Vicky Yang Wang, Daniel B. Ennis, Alistair A. Young, Martyn P. Nash</i> | 5 |
| 4 | Meshless Methods for LV Strain Computations from Tagged MRI
<i>Suejung Huh, Xiaoxu Wang, Dimitris Metaxas, and Leon Axel</i> | 15 |
| 5 | PPU-based deformable models for Catheterisation training
<i>Jixiang Guo, Shun Li, Yim Pan Chui, Qiang Meng, Howard Zhang, Simon Chun Ho Yu, Pheng Ann Heng</i> | 24 |
| 6 | 3D FEM/XFEM-based Biomechanical Brain Modeling for Preoperative Image Update
<i>Lara M. Vigneron, Romain C. Boman, Jean-Philippe Ponthot, Pierre A. Robe, Simon K. Warfield, and Jacques G. Verly</i> | 33 |
| 7 | Spherical Harmonics 3D Active Contours for Membrane Bilayer-Bound Surfaces
<i>Khaled Khairy, Jacques Pecreaux, Jonathon Howard</i> | 43 |
| 8 | Modelling Cerebral Cortical Folding
<i>Guangqiang Geng, Leigh Johnston, Edwin Yan, David Walker and Gary Egan</i> | 55 |
| 9 | Suite of finite element algorithms for accurate computation of soft tissue deformation for surgical simulation
<i>Grand Roman Joldes, Adam Wittek, Karol Miller</i> | 65 |

10	Finite Element Simulation of the Beating Heart for Image-Guided Robotic Cardiac Surgery <i>Philip Pratt, Fernando Bello, Eddie Edwards, and Daniel Rueckert</i>	74
11	Purely Evidence Based Multiscale Cardiac Tracking Using Optic Flow <i>Hans van Assen, Luc Florack, Avan Suinesiaputra, Jos Westenberg, and Bart ter Haar Romeny</i>	84
12	3-D Non-Linear Finite Element Analysis of Normal Pressure Hydrocephalus <i>Tomoy Dutta-Roy, Adam Wittek, Karol Miller</i>	94
13	Computational Biomechanics of the Breast: The Importance of the Reference State <i>Vijay Rajagopal, Angela Lee Jae-Hoon Chung, Poul M.F. Nielsen, and Martyn P. Nash</i>	103
14	High Resolution Ultrasound Elastography: a Dynamic Programming Approach <i>Hassan Rivaz, Pezhman Foroughi, Emad Boctor, Richard Zellars, Gabor Fichtinger and Gregory Hager</i>	113
15	Multi-resolution 3D Nonrigid Registration via Optimal Mass Transport on the GPU <i>Tauseef ur Rehman, Gallagher Pryor, John Melonakos, and Allen Tannenbaum</i>	122

Part 2. Computational Fluid Mechanics and Thermodynamics

16	Hardware-accelerated Bleeding Simulation for Virtual Surgery <i>Jing Qin, Wai-Man Pang, Yim-Pan Chui, Yong-Ming Xie, Tien-Tsin Wong, Wai-Sang Poon, Kwok Sui Leung, Pheng-Ann Heng</i>	133
17	Modeling Air-flow in the Tracheobronchial tree using Computational Fluid Dynamics <i>Ilhan Kaya, Anand P. Santhanam, Celina Imielinska and Jannick Rolland</i>	142
18	Imaging Facial Signs of Neuro-Physiological Responses <i>Dvijesh Shastri, Arcangelo Merla, Panagiotis Tsiamyrtzis, Ioannis Pavlidis</i>	152

Invited Lectures

A model sharing infrastructure for computational physiology

P.J. Hunter

Auckland Bioengineering Institute (ABI), University of Auckland, New Zealand
p.hunter@auckland.ac.nz

Abstract

The Physiome Project of the International Union of Physiological Sciences (IUPS) is attempting to provide a comprehensive framework for modelling the human body using computational methods which can incorporate the biochemistry, biophysics and anatomy of cells, tissues and organs [1-4]. A major goal of the project is to use computational modelling to analyse integrative biological function in terms of underlying structure and molecular mechanisms. It is also establishing web-accessible physiological databases dealing with model-related data at the cell, tissue, organ and organ system levels. Two major developments in current medicine are, on the one hand, the much publicised genomics (and soon proteomics) revolution and, on the other, the revolution in medical imaging in which the physiological function of the human body can be studied with a plethora of imaging devices such as MRI, CT, PET, ultrasound, electrical mapping, etc. The challenge for the Physiome Project is to link these two developments for an individual - to use complementary genomic and medical imaging data, together with computational modelling tailored to the anatomy, physiology and genetics of that individual, for patient-specific diagnosis and treatment.

To support these goals the IUPS Physiome project is developing XML markup languages (CellML & FieldML) for encoding models, together with model repositories and software tools for creating, visualizing and executing these models [5].

The talk will describe current progress in the development of these markup languages, the model repositories, graphical user interfaces and the open source computational software being developed under the IUPS Physiome Project for computational physiology.

References

1. Hunter, P.J. and Borg, T.K. Integration from proteins to organs: The Physiome Project. *Nature Reviews Molecular and Cell Biology*. **4**, 237-243, 2003.
2. Crampin, E.J., Halstead, M., Hunter, P.J., Nielsen, P.M.F., Noble, D., Smith, N.P. and Tawhai, M. Computational physiology and the Physiome Project. *Exp. Physiol.* **89**, 1-26, 2004.
3. Hunter, P.J. and Nielsen, P.M.F. A strategy for integrative computational physiology. *Physiology*. **20**, 316-325, 2005.
4. Hunter, P.J. Modeling living systems: the IUPS/EMBS Physiome Project. *Proceedings of the IEEE*. **94**:678-691, 2006.
5. www.cellml.org

Integration of Multiple Imaging Data for improved Volumetric Cardiac Motion Analysis

Dimitris Metaxas
Rutgers University, Piscataway NJ, 08854, USA
dnm@cs.rutgers.edu

Abstract

We present our recent efforts for the improved Volumetric Cardiac Motion Analysis based on data from multiple imaging modalities. First, we will present our framework for the automated spatiotemporal analysis of the heart's ventricles based on CT and tMRI data. Recent advances in CT have allowed the acquisition of high spatial resolution data that based on our deformable modeling methods we can build a detailed model of the ventricles. We then estimate the cardiac motion for a full cardiac cycle using tagged data, which is hard to achieve with a model constructed from only sparse clinical tagged MR images. Our accurate estimation algorithms compute two sets of cues from tagged MRI, the intersections of the three tagging planes, and the intersections of the cardiac boundary with the tagging planes. The image forces on the intersections are interpolated onto the cardiac mesh vertices by tessellation and meshless FEMs. The LV motion reconstruction provides information for further analysis of cardiac mechanisms. Results on normal and pathologic hearts will be presented. Finally, we will present recent results on the accuracy of 2D ultrasound-based cardiac analysis by comparing it to tMRI based analysis.

Part 1

Computational Solid Mechanics

Physiological Integration of Structural and Functional Cardiac Magnetic Resonance Imaging Using Finite Element Modelling

Hoi Ieng Lam¹, Vicky Yang Wang¹, Daniel B. Ennis², Alistair A. Young¹,
Martyn P. Nash¹

¹ Bioengineering Institute, University of Auckland, New Zealand
{h.lam, vicky.wang, a.young, martyn.nash}@auckland.ac.nz,

² Department of Cardiothoracic Surgery, Stanford University, Stanford, CA, USA
dbe@stanford.edu

Abstract. The left ventricle (LV) of the heart adapts its structure and function during diseases such as diabetes, hypertension, and myocardial infarction. However, there exists insufficient knowledge about the biophysical processes underlying normal and impaired cardiac function. We implemented a finite element approach to integrate physiological, microstructural, and biomechanical information into a canine LV mathematical model, using data obtained from *in vivo* magnetic resonance imaging (MRI) tissue tagging, *in vivo* LV pressure recordings, and *ex vivo* diffusion tensor MRI (DTMRI). Initially, a regular ellipsoid was constructed based on estimates of base-to-apex and wall thickness dimensions obtained from MRI in the end-diastolic state. The epicardial and endocardial surface data, segmented from the tagged MRI data, were then used to generate a customised canine LV geometrical model using nonlinear finite element fitting techniques. Myofiber orientations, obtained from DTMRI of the same heart, were incorporated into the model using host mesh fitting. LV pressure recordings were temporally synchronized to the MRI tissue tagging data. This methodology allows biophysical model parameters, such as the mechanical properties of the myocardium and activation characteristics, to be optimized to match the observed deformations and ventricular cavity pressures. Integrated physiological models for both normal and diseased conditions will then enable the comparison of biophysical parameters influencing cardiac function throughout the heart cycle.

Keywords: Mathematical Modeling; Cardiac Magnetic Resonance Imaging (MRI); Diffusion Tensor MRI (DTMRI); Left Ventricle (LV); Tissue Mechanics; Finite Element.

1 Introduction

In diabetes or myocardial infarction, heart cells adapt to physiological, geometric and loading changes in the cardiac muscle that arise from hemodynamic and geometric changes or pathologic processes. This leads to chronic regional thickening or thinning

of the ventricular wall, and enhancement or degradation in regional muscle function. Studying the regional function of the ventricles can lead to an improved understanding of the underlying structural basis of ventricular mechanics. In particular, information about the regional ventricular function provides important insight into pathological conditions, such as myocardial ischemia and infarction, where there can be significant localized mechanical changes in the myocardium whilst the global function is unaffected [1].

Magnetic Resonance Imaging (MRI) is well suited to the investigation of cardiac disease effects, due to its ability to non-invasively quantify three-dimensional (3D) changes in geometry and function of the heart. MRI tissue tagging with dynamic MRI enables quantitative evaluation of cardiac mechanical function with high spatial and temporal resolution. MRI tissue tagging is a technique that saturates the MRI signal in parallel bands of tissue, thus creating high contrast image features that accurately reflect the deformation of the underlying tissue at any point in the cardiac cycle. Reconstruction of the 3D motion of the heart from the tag positions during the cardiac cycle requires specialized image processing and mathematical techniques [2].

Diffusion tensor magnetic resonance imaging (DTMRI) measures the preferred orientations of the local self-diffusion of water molecules in biological tissues. Earlier studies have shown that the direction of maximum diffusion (the primary eigenvector) correlates to the observed myofiber orientation from histological studies [3]. Therefore, the primary eigenvector can be used for mapping the true 3D orientation of the myocardial fibers throughout the myocardium [4]. Myocardial fiber orientation is an important determinant of myocardial wall stress [5] and shares a large regional and transmural variation.

Integrating experimental information obtained from *in vivo* tagged MRI and *ex vivo* DTMRI adds insight to normal and abnormal regional cardiac function. This integration can be achieved by taking advantage of computer modelling to incorporate detailed information on ventricular geometry and myofiber orientation. A mathematical model of the heart is essential to this integrative approach. Previously, Augenstein *et al* developed methods for integrating MRI tagging, DTMRI and pressure recordings in *ex vivo* passive inflation experiments [6,7]. In this study, we extended this method to *in vivo* tagging and pressure recordings in dogs, using data acquired at the National Institutes of Health and Johns Hopkins University [8]. A graphical user interface (GUI) was developed for the segmentation of the epicardial and endocardial contours of DTMRI images, and readily identifiable landmark points on both the tagged and DTMRI images. Based on the DTMRI landmark points and MRI tissue tagging target points, a host mesh deformation approach was used to warp the fiber orientation data from the DTMRI images to the LV model by minimizing the distance between landmark and target points. The fiber orientations of the LV myocardium were extracted from each re-sampled DTMRI image by using the segmented contours.

2 Imaging and Segmentation

2.1 MRI Tissue Tagging

Imaging was performed using a General Electric 1.5T CV/i scanner and a 4 element phased array knee coil. Short axis stripe tagged images (Fig. 1) were acquired using the 3D fast gradient echo pulse sequence with the following parameters: 180mm x 180mm x 128-160mm field of view, 384 x 128 x 32 acquisition matrix, 12° imaging flip angle, ± 62.5 kHz bandwidth, TE/TR=3.4/8.0ms, 5 pixel tag spacing, and 4 mm slice thickness. Long axis radially-oriented stripe tag images were acquired with a 2D fast gradient echo pulse sequence and the following parameters: 200mm x 200mm x 8mm, 256 x 128 acquisition matrix, 12° imaging flip angle, ± 31.25 kHz bandwidth, TE/TR=3.2/8.0, 1 view per segment, and 7 pixel tag spacing.

LV epicardial and endocardial contour segmentation and tag detection were performed on 11 evenly-spaced short axis MR tagged images spanning from the base to apex, and 12 long axis MR tagged images with an angular separation of 30°, using the Findtags program by Guttman *et al* [9]. The LV motion obtained from the tagged images was analysed using four-dimensional b-spline based motion analysis [10], where the positions and strains of regularly-spaced material points within the myocardium were tracked three-dimensionally and in time. LV pressure was also measured throughout the cardiac cycle during the tagged MRI.

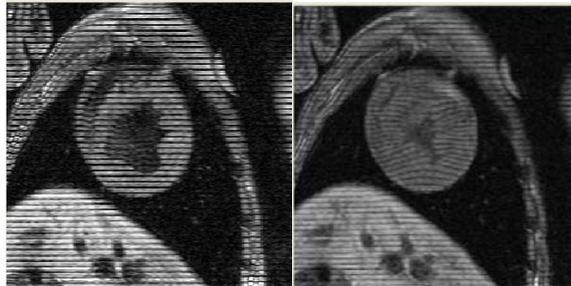


Fig. 1. A short axis MR tagged image at end-diastole (left) and end-systole (right).

2.2 Diffusion Tensor MRI

After performing the MR tagging study, the heart was excised and fixed in the end diastolic configuration for collecting DTMRI data (Fig. 2). The procedures are described in [8]. Diffusion tensor data was reconstructed from the diffusion weighted images and the eigenvector associated with maximum diffusion within each voxel was calculated in each DTMRI image. Each image had 256 x 256 in-plane measurements and there were 116 slices. The resolution of the DTMRI data was 390 μ m x 390 μ m x 800 μ m.

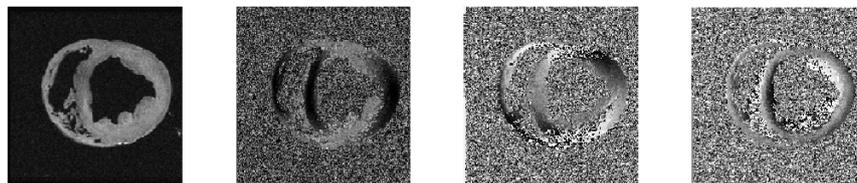


Fig. 2. (Left to right) Anatomical ($b=0$) image, and maps of x , y , and z component of the maximum diffusion eigenvector.

2.3 Surface Contour Segmentation of DTMRI Images

The segmentation of the epicardial and endocardial contours of each DTMRI image was performed manually using an in-house developed GUI. The GUI can display the tagged MR images simultaneously with the DTMRI images such that comparison between the two sets of image data can be done more conveniently. The DTMRI images can be rotated in-plane in the GUI to line up their orientation with the MR tagged images.

2.4 DTMRI Image Resampling

As illustrated in Fig. 3(a), the MR tagged images and the DTMRI images were orientated and scaled differently relative to the heart. This is because the shape of the heart was different when imaged *in vivo* versus *ex vivo* and the orientation of the prescribed cardiac long axis varied between imaging studies. Furthermore, the through-plane resolution of the two image data sets was substantially different, with each MR tagged image slice and each DTMRI image slice having a thickness of 4 mm and 0.8 mm, respectively. The low through-plane resolution of the MR tagged images implied that standard image based non-rigid registration between the two data sets would not be the optimal choice. However, the high through-plane resolution of the DTMRI images would allow new DTMRI images to be reformatted with the same orientation and location as the tagged images within the heart.

Since the tagged MR images and the DTMRI images were acquired under different conditions, both data sets were transformed into a standard cardiac coordinate system, where the x -axis is defined to be the long-axis of the heart i.e. running from base to apex, the y -axis points from the LV centre towards the right ventricle (RV) and z -axis points from the anterior towards the posterior of the heart. The origin of the cardiac coordinate system is defined to be at one third from the base of the heart along the x -axis.

Image resampling of the DTMRI images was achieved by matching cardiac coordinate systems between the two datasets, and then locating the coordinates of the corners of the MR tagged images within the DTMRI image volume matrix (Fig. 3(a)). The coordinates of each pixel of each MR tagged image within the DTMRI image volume matrix were then obtained (Fig. 3(b)). The DTMRI image volume matrix was

then resampled at the locations of the MR tagged images within the volume, and also at evenly-spaced parallel planes between the MR tagged images (Fig. 3(c)).

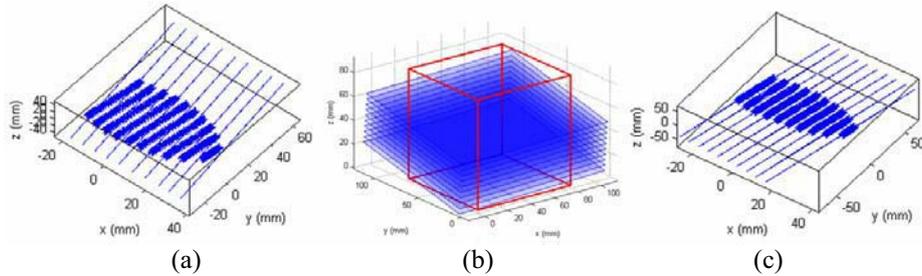


Fig. 3. (a) The DTMRI image slice planes (lines) and the segmented contours of the MR tagged images (markers) plotted in the cardiac coordinate system, showing the misalignment between the two data sets. (b) The slice positions of the MR tagged images transformed into the DTMRI image volume (confined by the box). The DTMRI image volume was resampled at these slice positions of the tagged images. (c) The resampled DTMRI image slice planes (lines) and the contours of MR tagged images (markers) plotted in the cardiac coordinate system.

2.5 Fibre Orientation Data Extraction

The three components of the maximum diffusion eigenvector, which were stored in three separate matrices, were associated with each pixel of the DTMRI image. Only the eigenvectors associated with the LV myocardium were of interest in this study, thus it was necessary to extract the three components of these eigenvectors from the three matrices. First, contours of the re-sampled DTMRI images were manually segmented using the GUI (Fig. 4). These contours were then used to create a mask such that pixels not within the LV myocardium were excluded from the image. The mask for the DTMRI image was eroded by 1 pixel to exclude the noisy pixels at the edge of the myocardium.

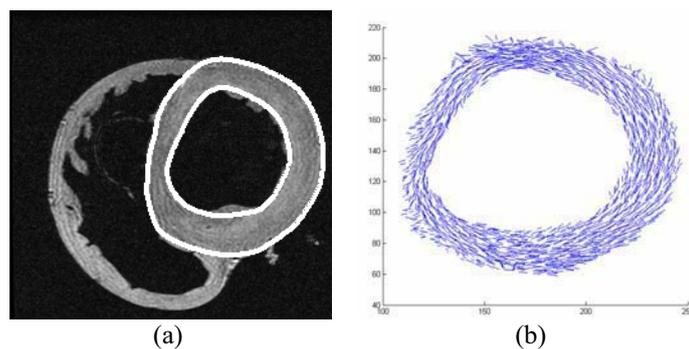


Fig. 4. (a) DTMRI anatomical image with its segmented contour. (b) Myocardial fibre orientation (in two-dimension) at the location of the myocardium extracted using the mask created from the segmented contours.

2.6 Landmark Selection

The GUI described above also allows the selection of fiducial markers (e.g. papillary muscles, RV inserts) on the MR tagged images and the corresponding re-sampled DTMRI images (Fig. 5). For each pair of corresponding MR tagged image and DTMRI resampled image, fiducial markers were first selected from the tagged image and each marker was sequentially assigned a number to indicate the order of selection. Then the same number of fiducial markers was selected on the DTMRI image in the same sequential order. The fiducial markers selected on the MR tagged images were referred to as *target points*, and those on the DTMRI images were referred to as *landmark points* and were used for host mesh fitting.

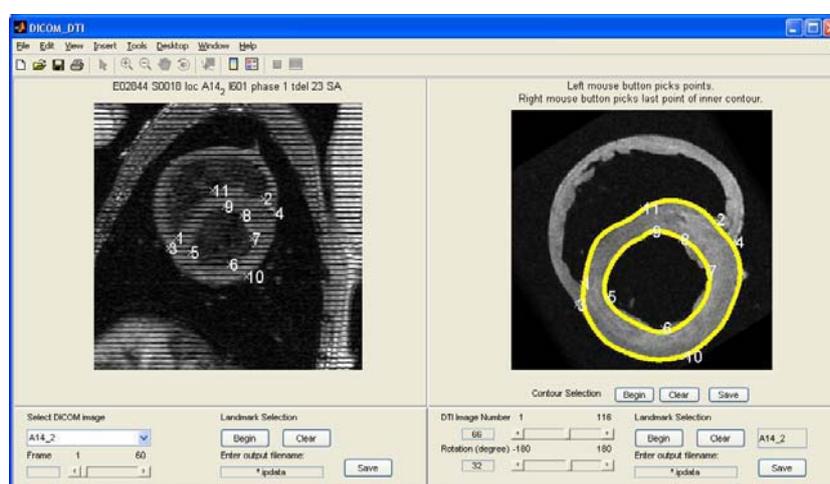


Fig. 5. A screenshot of the GUI showing the manually segmented contours on the DTMRI image (right side), and the manually selected fiducial markers on both the MR tagged image (left side) and DTMRI image.

3 Host Mesh Fiber Mapping

As described above, the segmented contours of the DTMRI images and the tagged MRI images exhibited different orientations in 3D, since the canine heart was imaged during different conditions. Therefore, it is necessary to ensure that the geometry defined by the DTMRI images and the tagged MR images was consistent before the fiber orientation data were incorporated to the finite element geometric model created. This was achieved by performing host mesh fitting, a technique which was designed to customize generic models to specific cases [11]. In our study, we used the host mesh fitting technique to warp the DTMRI data into the *in vivo* geometric model defined by the tagged MRI data. This host mesh fitting approach involved minimizing the distance between two sets of points: *landmark points* selected from the resampled

DTMRI images and corresponding *target points* selected from the MR tagged images. This minimization was implemented by embedding the landmark points in another finite element mesh, called the *host mesh* and minimizing the total squared error with respect to *host mesh nodal parameters*. The host mesh used for this study was a simple tri-cubic mesh consisting of 1 element connected with 8 element nodes (Fig 6). Since the target points were embedded in the host mesh, any deformation the host mesh underwent caused an interpolated degree of deformation for the target points. That is, the local coordinates of the target points with respect to the host mesh remained unchanged before and after deformation. Because the host mesh had a simpler geometry and fewer number of elements, the computational cost of this minimization was reduced significantly, an important benefit of host mesh fitting. Once the optimum host mesh nodal parameters were evaluated, the global coordinates of landmark points were updated based on the local coordinates defined with respect to the host mesh nodal parameters and the transformation matrix obtained during fitting. The same transformation matrix was also applied to the DTMRI data so that the geometries defined by DTMRI and tagged MRI were consistent. Subsequently, fiber orientations could then be embedded into the geometric model using the deformation gradients from the host mesh transformation, and used for mechanical analysis.

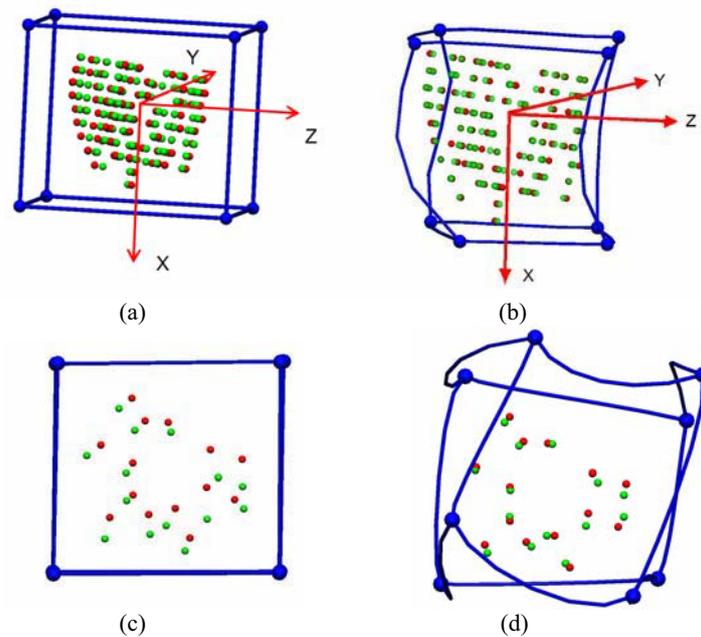


Fig. 6. (a) Undeformed host mesh (lines) with landmark points (light) and target points (dark). (b) Deformed host mesh with target points and transformed landmark. (c) A through-plane view of the landmark and target points for one slice before host mesh fitting. (d) A through-plane view of the landmark and target points from the same slice after host mesh fitting. The host mesh fit reduced the root mean squared error between landmark and target points from 4.1 mm to 0.8 mm.

4 Left Ventricular Finite Element Model

4.1 Left Ventricular Anatomy and Structure

A mathematical model of the LV was created in the cardiac coordinate system based on non-linear finite element optimization of the model geometry to the segmented contours of the MR tagged images (Fig. 7). A regular ellipsoid was initially created in a prolate spheroidal coordinate system, using dimensions based on the base-to-apex dimension and wall thickness estimated from MR tagged images. Prolate spheroidal coordinates were used to define the heart geometry in order to reduce the number of elements needed to represent the complex three-dimensional ventricular geometry. A rectangular Cartesian coordinate system, however, was employed for tissue mechanics analysis. The ellipsoid consisted of 16 finite elements which included 4 circumferential elements, 4 longitudinal elements and 1 transmural element (Fig. 4). A material coordinate system was also defined such that it was attached to material points and moved with the myocardium as it deformed. Finite element material coordinates (ξ_1, ξ_2, ξ_3) were directly associated with element geometry, with ξ_1 in the circumferential direction, ξ_2 in the transmural direction, and ξ_3 in the longitudinal (apex-base) direction. The spatial variation of geometric information within each element was approximated using tri-cubic Hermite interpolation of parameters defined at the element nodes, which implicitly enforces spatial gradient continuity across element boundaries [12]. The nodal parameters obtained after minimization constituted the optimized geometric model.

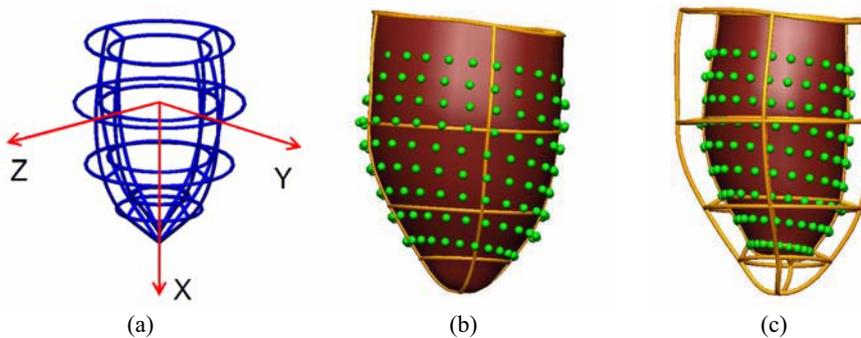


Fig. 7. (a) A regular ellipsoid created as the initial estimate of the LV geometric model where x : base-to-apex, y : left-to-right, and z : anterior-to-posterior. (b) Epicardial surface of the initial ellipsoid fitted to the short axis epicardial contour. (c) Endocardial surface of the initial ellipsoid fitted to the short axis endocardial contour. The root mean squared error in the surface fit was 0.8 mm.

The LV fiber architecture can be defined throughout the LV geometry using nonlinear optimization of a fiber field to the transformed fiber vectors derived from DTMRI, using the methods outlined in [7].

4.2 Left Ventricular Mechanics

LV pressures were sampled at each MR time frame as illustrated in Fig. 8. Given the boundary conditions from the LV pressure, the finite deformation elasticity problem can be solved using the method outlined in [13]. The mechanical properties and active parameters can be tuned such that deformation predictions reliably match the observed displacements of material points derived from the tagged MRI data [7]. In this way we can then gain insight into regional distributions of myocardial stress and thus functional measures such as local energy consumption.

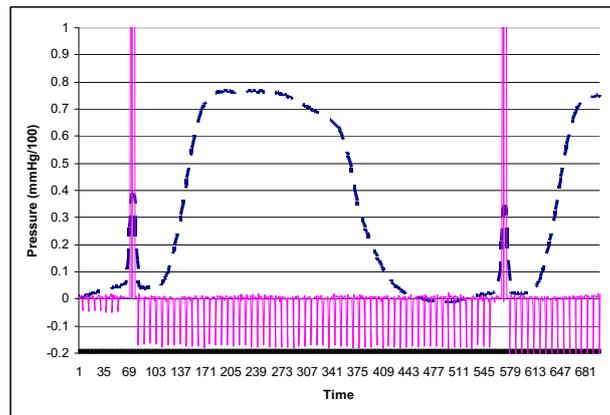


Fig. 8. LV pressure over one cardiac cycle (dashed line), with spikes indicating the timing of the MR imaging.

5 Conclusions

We have extended previous finite element based modelling methods to the characterization of cardiac structure and function from *in vivo* canine MRI tissue tagging and pressure data, and *ex vivo* DTMRI derived microstructural information. Due to the different orientation of the MR tagged images and the DTMRI images, image resampling was performed to obtain DTMRI images which closely correspond to the MR tagged images. In order to incorporate the myocardial fibre orientation data from the DTMRI images, fiducial markers were selected on the MR tagged images and the resampled DTMRI images for host mesh fitting. This method enables integration of data from different acquisitions into a coherent model of structure and function. In the future these methods will facilitate characterization of biophysical parameters from cardiac MR data.

References

1. Glass, L., Hunter, P. and McCulloch, A. (eds.): Theory of heart: biomechanics, biophysics, and nonlinear dynamics of cardiac function. Springer-Verlag, New York (1991)
2. O'Dell, W.G., Moore, C.C., Hunter, W.C., Zerhouni, E.A., McVeigh, E.R.: Three-dimensional Myocardial Deformations: Calculation with Displacement Field Fitting to Tagged MR Images. *Radiology* 195 (1995) 829-835
3. Le Bihan, D., Mangin, J.F., Poupon, C., Clark, C.A., Pappata, S., Molko, N., Chabriat, H.: Diffusion Tensor Imaging: Concepts and Applications. *Journal of Magnetic Resonance Imaging* 13 (2001) 534-546
4. Douek, P., Turner, R., Pekar, J., Patronas N. J., Le Bihan, D.: MR color mapping of myelin fiber orientation. *Journal of Computer Assisted Tomography* 15 (1991) 923-929
5. Geerts, L., Bovendeerd, P., Nicolay, K., Arts, T.: Characterization of the normal cardiac myofiber field in goat measured with MR-diffusion tensor imaging. *American Journal of Physiology (Heart and Circulatory Physiology)* 283 (2002) H139-H145
6. Augenstein, K.F., Cowan, B.R., LeGrice, I.J., Young, A.A.: Estimation of cardiac hyperelastic material properties from MRI tissue tagging and diffusion tensor imaging. *MICCAI* 9:1 (2006) 628-635
7. Augenstein, K.F., Cowan, B.R., LeGrice, I.J., Nielsen, P.M., Young, A.A.: Method and apparatus for soft tissue material parameter estimation using tissue tagged Magnetic Resonance Imaging. *J Biomech Eng.* 127:1 (2005) 148-157
8. Ennis, D.B.: Assessment of Myocardial Structure and Function Using Magnetic Resonance Imaging. PhD Thesis, John Hopkins University (2004). (See also <http://www.ccbm.jhu.edu>)
9. Guttman, M.A., Prince, J.L., McVeigh, E.R.: Tag and contour detection in tagged MR images of the left ventricle. *IEEE Trans. Med. Imaging* 13 (1997) 74-88
10. Ozturk, C., McVeigh, E.R.: Four dimensional B-Spline based motion analysis of tagged MR Images: introduction and in-vivo validation. *Phys. Med. Bio.* 45 (2000) 1683-1702
11. LeGrice, I., Hunter, P.J., Young, A.A., Smaill, B.: The architecture of the heart: a data-based model. *Phil. Trans. R. Soc. Lond.* 359 (2001) 1217-1232
12. Bradley, C.P., Pullan, A.J., Hunter, P.J.: Geometric Modeling of the Human Torso Using Cubic Hermite Elements. *Annals of Biomedical Engineering* 25 (1997) 96-111
13. Nash, M.P., Hunter, P.J.: Computational Mechanics of the Heart: From Tissue Structure to Ventricular Function. *Journal of Elasticity* 61 (2000) 113-141

Meshless Methods for LV Strain Computations from Tagged MRI

Suejung Huh¹, Xiaoxu Wang¹, Dimitis Metaxas¹, and Leon Axel²

¹ Rutgers University, Piscataway NJ, 08854, USA
suejung@cs.rutgers.edu,

² New York University, 660 First Avenue, New York, NY, 10016, USA

Abstract. This paper presents a meshless framework to compute strain in the left ventricle from Tagged MRI. Our meshless framework allows the computation of a complex but smooth strain field, since it is not bounded to the underlying mesh structure of a model. In this paper, we used Tagged MR images of left ventricles. However, the suggested formulation is independent of the image modality and the choice of fitting mechanism.

1 Introduction

A meshless method to solve partial differential equations was introduced in computational mechanics by [1] according to [2], although the meshless method itself was introduced in the 70's [3]. Despite their great flexibility, meshless methods have not been utilized significantly in the medical field. Recently the meshless approach was utilized to model surgical processes [4, 5]. Meshless approaches are well suited to model a biomedical organ in surgical simulations, when the topology of the organ changes interactively by surgical processes such as cutting and stitching. The fundamental flexibility of the meshless approach lies in the following : 1) a field is defined by the not-explicitly-connected participating particles, 2) we can change the density of the constitutive particle populations without changing the field property, and 3) any changes in such constitutive particles can be handled locally. Since a meshless approach has such flexibility, it is appropriate for strain computations, in areas where we observe drastic changes in deformation patterns.

The left ventricle (LV) exhibits large deformations during the cardiac cycle [2], and has a very complex geometry with varying material properties in the endocardium [6]. Galerkin approaches on discretized domains, such as like Finite Element Methods (FEM) or Boundary Element Methods (BEM), have been used in numerous studies of strain analysis of LV deformations, including [7–12]. However, when we compute strains during the LV deformation, spatially discretized elements like the ones used in FEM or BEM could be too coarse to encompass all the details of the strain field. Furthermore, increasing the number of volumetric elements to express all the details could be computationally expensive, and more importantly erroneous, since it could produce too many degenerate volumetric

elements during a simulation. Having such irregularly shaped elements could undermine the stability of simulations. Hence we introduce a meshless formulation to compute the strain field for LV during systole.

In Sec. 2, we introduce our meshless strain formulation method, and in Sec. 3, the methodology for fitting is explained. Sec. 4 shows the experimental results tested on two slices of the in-vivo myocardium, and a discussion with future directions follows in Sec. 5.

2 Meshless Strain Computation

In order to measure the deformation of the LV, strain fields of the LV have been analyzed by many researchers. In this paper, we measure the strain field using the meshless method. In the meshless method, an object is represented by points only. All points are not explicitly connected. The strain tensor of a point is computed based on the neighboring points. Obtained strain fields will be smooth because all the strain tensors are based on their neighbors. Among several ways to compute strain [13], we use the lagrangian strain tensor since it is more suitable for large deformations and reports deformations with respect to the original shape of an object.

2.1 Strain Computation for Large deformations

Given the initial position of a point $x_0 = (x, y, z)$ in a global coordinate system and the displacement $u(t) = (u_x, u_y, u_z)$ at time t , the current position of the point is $x(t) = x_0 + u(t)$. The deformation gradient J is defined as $J = \delta x(t) / \delta x_0$.

$$J = \delta(x_0 + u(t)) / \delta x_0 = \begin{bmatrix} 1 + u_{x,x} & u_{x,y} & u_{x,z} \\ u_{y,x} & 1 + u_{y,y} & u_{y,z} \\ u_{z,x} & u_{z,y} & 1 + u_{z,z} \end{bmatrix} = I + \nabla U^T \quad (1)$$

Given the deformation gradient J , the lagrangian strain tensor ϵ is computed as

$$\epsilon = \frac{1}{2}(J^T J - I) \quad (2)$$

In our strain formulation, a point is not a member of any spatial elements. Since a point does not have any element to which the point belongs, the straight forward strain computation method in FEM can not be used. Instead a method which approximates the strain tensor of a point in the relation to its neighboring points has to be used.

2.2 Approximating ∇U

To compute the strain tensor of a point, an approximation method based on the relation to its neighboring points has been adopted. In this paper, we adopt an

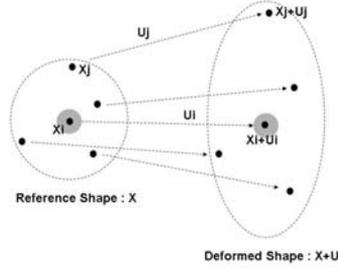


Fig. 1. The strain of a point x_i is computed based on the relation with the neighboring points x_j .

approximation method to compute the strain tensor of a point using the meshless Galerkin's approach [14]. The density of a mass centered at a point is weighted based on smoothed particle hydrodynamics (SPH). The set of neighboring points as shown in Fig. 1 is determined at the initial reference shape of an organ (not necessarily the least deformed shape). We determine the neighboring radius $h = 3R$ where R is the average distance of the closest point pairs in all points. The weight function $w(i, x)$ originating from x_i is normalized as $\int_x w(i, x) dx = 1$. Using the moving least squares formulation method [15], without loss of the generality, the x component of the displacement gradient ∇U at the node i is computed as

$$\nabla u_x|_{x_i} = A^{-1} \left(\sum_j (u_x(j) - u_x(i)) x_{ij} w(i, j) \right), \quad (3)$$

where $x_{ij} = x_i - x_j$, $u_x(j)$ is the x component of the displacement u at the node j in the vicinity of the node i , and the momentum matrix $A = \sum_j x_{ij} x_{ij}^T w(i, j)$. Refer [16] for the detailed derivation of Δu_x . The other y and z components of ΔU can be computed similarly. Once we have ∇U , we can compute the strain tensor from Eq. 2. Given a 3x3 strain tensor, principal strain analysis is performed. Principal strain analysis is used to find eigenvalues and eigenvectors of a tensor to eliminate shearing components. Positive/negative eigenvalues of principal strains represent elongations/contraction deformations, respectively, while the corresponding eigenvectors represent the direction of deformations.

Computation: Using Tagged MR images of LVs, we first fit a 3D LV mesh to the image data, based on the technique explained in Sec. 3. From the fitted model to each frame, we compute the strain field of the LV with respect to the reference shape. For an area of interest, to compute a more detailed strain field, we increase the number of points scattered, without remeshing the original mesh used for fitting. The points used for meshless strain computations are

defined based on their barycentric coordinates with respect to the underlying mesh structure. Thus more points can be added for strain estimation after the fitting.

2.3 Comparison with FEM

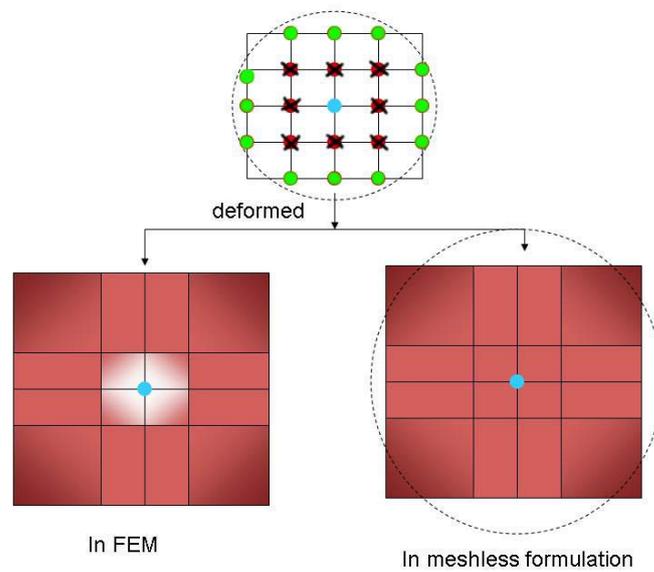


Fig. 2. From the reference shape shown at the top, all the elements surrounded by (X) points have elongated, while the elements adjacent to the center point have not deformed because of a fitting error. The strain field computed using FEM approaches will illustrate zero strain fields around the center point. A meshless strain approach will compute a smooth strain field as shown in the right picture, since the strain in meshless methods is computed based on neighboring points. In this example, the grey scale has been used to visualize the density of $\sum |\epsilon_i|$ for all eigenvalues ϵ_i of principal strains.

In FEM, any error in fitting could result in errors in strain computations directly. In Fig. 2, consider the case where the four elements neighboring the blue node have not deformed at all since they failed to fit to the image data while the other elements with the green nodes have fitted to the image data. In FEM, the strain computed in the blue node will be zero (actually, a zero tensor). Any failures of fitting directly affect the strain analysis of that area.

In meshless formulation, this adverse effect of a fitting failure can be minimized, provided the vicinity of the center node includes more neighboring nodes other than (x) marked nodes, since the strain field is computed in conjunction with all its neighboring nodes (Extended kernel). Although the volume defined by the (x) marked nodes and the center node have not deformed at all, the strain field on the center node still could be a smooth tensor as it is influenced by other green nodes. Hence a small fitting error does not affect the strain analysis directly. For the case, where we actually expect a small scale of deformations, we can populate the area with more nodes with a smaller vicinity threshold to capture small details. Expanding the kernel size of the meshless approach excessively will smoothen the strain field unnecessarily. Picking the appropriate kernel size for each application is essential to maintain the correctness of the system.

3 LV Fitting

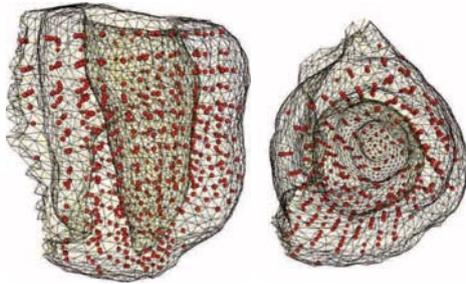


Fig. 3. A fitted generic heart model

In this paper, we used Tagged MR images to estimate LV movements. We fit a 3D generic heart mesh, based on the approach used in [17]. First, we obtain contours and tagging line information from tagged MRI using Gabor Filters [18], Metamorphs [19] and spline model [20]. The intersections of the three tagging planes are calculated, as well as the intersections of the LV boundary and the tagging planes. These intersections can be used as material markers in LV motion tracking. Then we register a generic heart mesh to the image data by affine registration and non-rigid thin plate spline local fitting. We build a Delaunay tessellation on intersection points and interpolate the image forces onto the mesh points by FEM. FEM dynamics is used to reconstruct the LV motion while preserving the topology and shape of the model. The fitted image is shown in Fig. 3, about 6K elements have been used.

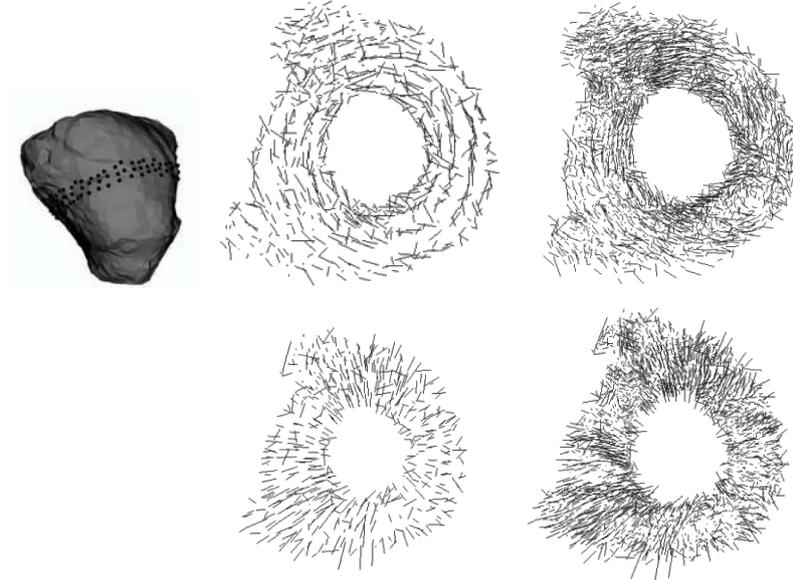


Fig. 4. The strain field from a slice in the middle of LV, visualized in different eigenvectors. The middle slice is shown in the left most picture. The strain elements of interest are visualized in 3D and the images are taken from the top view. **Circumferential Strain(Contraction)**: The upper pictures show the contraction deformation of the LV. Each edge represents the eigenvector corresponding to the negative eigenvalue of the principal strains. The length of each eigenvector is determined by its eigenvalue. **Radial Strain(Elongation)**: Lower Pictures show the elongation deformation of LV. Each edge represents the eigenvector corresponding to the largest positive eigenvalue of the principal strains, whose length is determined by its eigenvalue. **Increased Points**: In both type of deformations, the right pictures show results with more populated point cloud.

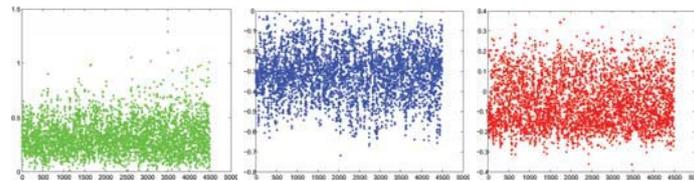


Fig. 5. All the eigenvalues from ten subjects were plotted with respect to the points used in the meshless method. **Left(Elongation)**: The largest positive eigenvalues. **Center(Contraction)**: The largest magnitude least negative eigenvalues. **Right**: The third principal (smallest magnitude) eigenvalues.

4 Results

We used ten normal subjects to measure the strain field. After fitting the heart model to Tagged MR images, the strain field is computed using our meshless approach. We visualize the results of the principal strain analysis with the strain tensors based on the sign of eigenvalues. It is very natural to separately analyze elongation and contraction deformations, since it gives us more information of LV movements. The largest positive eigenvalues, which mostly represent the transmural thickening (radial elongation) deformation, are shown in the lower two pictures in Fig. 4. The pictures of the largest magnitude of negative eigenvalues represent in general circumferential contractions in myocardium. The data of the eigenvalues are shown in Fig 5. The average of both largest positive and negative eigenvalues were around 0.3 in their magnitude, while the average of the smallest eigenvalues was close to zero. 0.3196, -0.3134, and 0.0487 were the average of the positive, negative, and smallest eigenvalues respectively. This means that during systole the overall volume of the LV wall is approximately preserved. In Fig. 6, the strain field shows the contraction (negative eigenvalues) along the muscle direction. As shown, the results are a lot more smoother compared to an FEM approach which would be significantly more computationally expensive and would be not possible to guarantee second order smoothness in the strains across elements.



Fig. 6. Contractions in side view

The smoothness effect of using meshless formulation, as discussed in Sec. 2.3, is tested using the Frobenius norm $\sqrt{\sum_{i=0}^n \sum_{j=0}^n D_{ij}^2}$ where $D_{ij} = E_{ij}^A - E_{ij}^B$ for the strain fields E_{ij}^A and E_{ij}^B of the nodes A and B, respectively. When the strain field is smoother, we have a smaller Frobenius norm. Fig. 7 shows strain fields become smoother as the kernel sizes increase. The Frobenius norm of the strain field using FEM was 0.086 while the one of the smallest kernel size of $2.25 r_{avg}$ was 0.061. This shows the smoothing effect of the meshless approach.

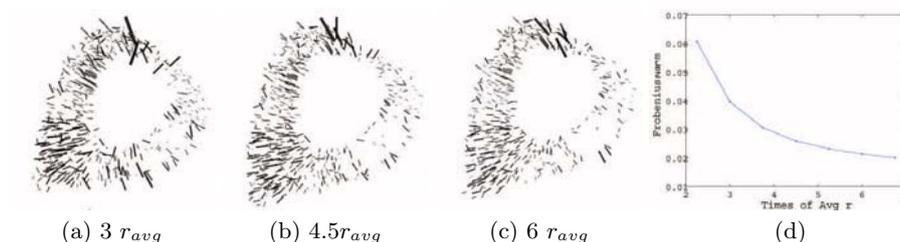


Fig. 7. The x axis in smoothness is the multiples of the average r of 10 nearest neighbors. (a),(b) and (c) shows the contracting strain computed with the kernel size $3 r_{avg}$, $4.5 r_{avg}$ and $6 r_{avg}$ respectively. All three are taken near the middle ventricle, where r_{avg} is the average distance of all closest node pairs. (d) plots the Frobenius norms of the different kernel sizes.

5 Discussion

In this paper, we introduced a meshless formulation to compute the strain field of the LV. We have introduced a new meshless framework to model accurately the LV strain. Using this framework we have shown more accurate and smoother results than traditional FEM methods. Our meshless framework for strain analysis is versatile, so it can be used for other organs.

References

1. Nayroles, B., Touzot, G., Villon, P.: Generalizing the finite element method: Diffuse approximation and diffuse elements. *Journal Computational Mechanics* **10**(5) 307–318
2. N., S.: Meshless methods and partition of unity finite elements. In of the Sixth International ESAFORM Conference on Material Forming (2003) 603606
3. Liu, G.: *Mesh Free Methods: Moving Beyond the Finite Element Method*. CRC Press (2002)
4. Vigneron, L.M., Verly, J.G., Warfield, S.K.: Modelling surgical cuts, retractions, and resections via extended finite element method. In: MICCAI (2). (2004) 311–318
5. Horton, A., Wittek, A., Miller, K.: Towards meshless methods for surgical simulation. In: *Proceedings of MICCAI Workshop*. Copenhagen. Oct. 2006. ISBN 10:87-7611-149-0. 34-43 p.
6. Chen, T., Metaxas, D.N., Axel, L.: 3d cardiac anatomy reconstruction using high resolution ct data. In: MICCAI (1). (2004) 411–418
7. Haber, E., Metaxas, D.N., Axel, L.: Motion analysis of the right ventricle from mri images. In: MICCAI '98: *Proceedings of the First International Conference on Medical Image Computing and Computer-Assisted Intervention*, London, UK, Springer-Verlag (1998) 177–188
8. Kerwin, W., Prince, J.: *Cardiac material markers from tagged mr images* (1998)

9. Shi, P., Sinusas, A.J., Constable, R.T., Duncan, J.S.: Volumetric deformation analysis using mechanics-based data fusion: Applications in cardiac motion recovery. *Int. J. Comput. Vision* **35**(1) (1999) 87–107
10. Hu, Z., Metaxas, D.N., Axel, L.: In-vivo strain and stress estimation of the left ventricle from mri images. In: *MICCAI '02: Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention-Part I*, London, UK, Springer-Verlag (2002) 706–713
11. Park, K., Metaxas, D.N., Axel, L.: A finite element model for functional analysis of 4d cardiac-tagged mr images. In: *MICCAI* (1). (2003) 491–498
12. Yan, P., Lin, N., Sinusas, A.J., Duncan, J.S.: A boundary element-based approach to analysis of lv deformation. In: *MICCAI*. (2005) 778–785
13. Fung, Y.C.: *A first course in continuum mechanics*. Prentice-Hall (1969)
14. Belytschko, T., Krongauz, Y., Organ, D., Fleming, M.: *Meshless methods : An overview and recent developments*. Civil and Mechanical Engineering, Northwestern University (1996)
15. P., L., K., S.: Surfaces generated by moving least squares methods. *Mathematics of Computation* **87** (1981) 141–158
16. Müller, M., Keiser, R., Nealen, A., Pauly, M., Gross, M., Alexa, M.: Point based animation of elastic, plastic and melting objects. In: *SCA '04: Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation*, Aire-la-Ville, Switzerland, Switzerland, Eurographics Association (2004) 141–151
17. Wang, X., Schaerer, J., Qian, Z., Metaxas, D., Chen, T., Axel, L.: Reconstruction of detailed left ventricle motion using deformable model from tagged mrimage
18. Qian, Z., Metaxas, D.N., Axel, L.: Extraction and tracking of MRI tagging sheets using a 3D gabor filter bank. *Proceedings of Intl Conf. of the Engineering in Medicine and Biology Society* (2006)
19. Huang, X., Metaxas, D., Chen, T.: *Metamorphs: Deformable shape and texture models* xiaolei huang metaxas, d. ting chen. *Computer Vision and Pattern Recognition, 2004*. (2004)
20. Axel, L., Chen, T., Manglik, T.: Dense myocardium deformation estimation for 2d Tagged MRI. In: *FIMH*. (2005) 446–456

PPU-based deformable models for Catheterisation Training

Jixiang Guo¹, Shun Li¹, Yim Pan Chui¹, Qiang Meng¹, Howard Zhang¹,
Simon Chun Ho Yu², Pheng Ann Heng^{1,3}

¹Department of Computer Science and Engineering,
The Chinese University of Hong Kong, Shatin, Hong Kong

²Department of Diagnostic Radiology and Organ Imaging,
The Chinese University of Hong Kong, Shatin, Hong Kong

³Shun Hing Institute of Advanced Engineering,
The Chinese University of Hong Kong, Shatin, Hong Kong

Abstract. In this paper, we propose a framework for generating deformable models for catheterisation training applications. Through exploiting centreline extraction, graph reconstruction, curve fitting and curve framing techniques, we can model vascular structures and other virtual catheterisation devices using mathematically contrived geometries with minimal user interactions. A Physics Processing Unit (PPU) based incremental voigt model has been proposed for incorporating non-linear biomechanical/mechanical properties into the deformable models. Experiments have shown a reasonable increase in frame rates from 2-fold to 4-fold over non-PPU-based simulations. Our results have demonstrated the feasibility of using this newly evolved multi-core physics accelerator for speeding up medical training applications such as virtual vascular catheterisation.

1 Introduction

Catheterisation has been one of the major medical procedures being used in vascular interventional radiology for the remedy of stenosis, aneurysm, etc. Based on image-guided X-ray fluoroscopy or ultrasonography, a thin flexible tube called catheter is inserted into a vessel for vascular procedures such as angiography, angioplasty and embolization. Due to the limited visual perception during the image-guided procedures, training of these procedures is difficult. Recently, virtual reality (VR) based medical simulations have become more popular due to their reusability and flexibility.

Within a virtual simulation environment for vascular intervention procedures, virtual devices such as guide wires or catheters can often be interacted with each other. Thus, effective collision computation between various deformable models is an essential task. Currently, real-time simulation of deformable models in medical simulation remains a challenging task. Furthermore, visualizations of these procedures that are of high quality and high fidelity demands even more intensive computation.

In this paper, a novel deformable modeling framework for virtual catheterisation is proposed. The new personal computer (PC) grade, multicore processor called the Physics Processing Unit (PPU) has been exploited in developing our framework for reconstructing deformable models suitable for interactive medical simulations. Vascular structure and virtual devices can be built through an automated topological geometric modeling engine. PPU provides built-in support of a mass-spring-damper model for describing linear elastic motions. We extend the model to an incremental-voigt one for simulating the non-linear biomechanical stress-strain behaviour of soft tissue. With such an hardware-acceleration, real-time interactivity can be achieved.

2 Related Work

VR based medical simulations for minimally invasive surgery (MIS) has become more prevalent in the recent decade. Virtual endoscopy has been exploited to simulate colonoscopy [8], arthroscopy [6], laparoscopy [3] and hysteroscopy [5] etc. Developing simulation systems for vascular interventional procedures have been a very active research area [7][2][14]. Many existing works focus on the geometric modeling of vascular structure for catheter navigation. Nowinski et al. proposed a virtual environment for simulating deformable vascular modeling [11], where by assuming a static external vessel wall structure, interactive performance can be achieved. In order to further enhance the realism of the simulation, a more flexible deformable modeling for vascular structure is needed. In light of this, we propose the PPU-based framework for modeling and simulating realistic biomechanical features of the vascular system so that deformable models can be used for virtual catheterisation.

3 Modeling Framework

In catheterisation procedures, guide wire or catheter is directed to the region of interest through blood vessels. Simulated vessel models, and virtual devices constitute the basic components of a virtual catheterisation simulation. Based on angiographic data such as magnetic resonance angiography (MRA) or computed tomographic angiography (CTA), centrelines or namely skeletons of the interested vessels are extracted. An automated topological reconstruction process is carried out in order to build a geometric model of the vessel. Virtual devices such as guide wires or catheters can also be built through the modeling framework.

3.1 Skeleton Extraction

The extraction of a vascular skeleton can be done by a two pass procedure. First, a thinning process is performed to locate the skeleton voxels, which represent the abstract vessel information, within a 3D volume of angiographic data. We adopt a fully automatic centerline calculation algorithm based on a 3D topological thinning proposed in [15]. Then a vessel graph can be constructed.

3.2 Vessel Graph Construction

Based on the extracted skeleton points, a graph is constructed for later topological reconstruction process. We can consider the structure of the blood vessel as a directed graph. An adjacency list is deployed for such representation where centerline points are restored in a linked link. A junction is represented as a vertex which has several edges linking to others. The rest of the vertices are regarded as segment points. Adjacent points with no junction should be arranged sequentially, the line ends when a junction or end point is found.

Only a subset of a given set of points in a vessel graph are used in the modeling. We select feature points based on the local curvature along the skeleton curve. For the initial n segment points $p_0, p_1 \dots, p_{n-1}$, with the approximated local curvature defined as $\|p_i''\|$, a feature point will be selected if the condition $\|p_i''\| > \xi$ is satisfied, where ξ denotes the curvature threshold (Fig. 1(a)) Eventually, m feature points s_i ($i = 0, \dots, m - 1$) are selected. Then, a number of curve segments are fitted onto selected feature skeleton points. Feature points are exploited as the end point of individual Bézier segments. Extra control points are computed through colinearize local end point tangents between adjacent segments. In this sense, a C^1 continuous curve can be guaranteed.

3.3 Vessel Frames

From a particular volume data set containing the vessel information, such as one from MRA or CTA, the surface may be obtained through marching cubes or skeleton climbing [13]. However, the irregular mesh structure generated is usually not suitable for the reconstruction of computationally efficient deformable models. We propose a re-framing process so that the surface mesh can be reparameterized into a more regular grid structure. The whole process undergoes several major steps: tubular segment creation, bifurcate creation, parallel transport framing and tetrahedral grid reconstruction.

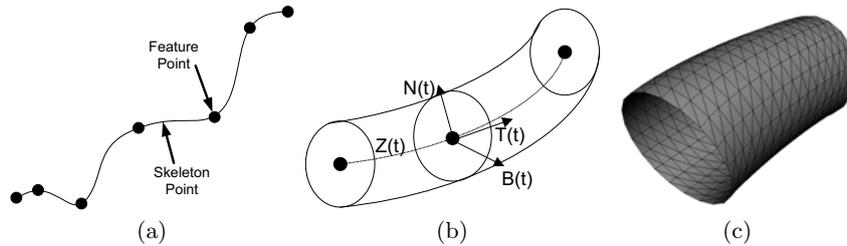


Fig. 1. (a) Feature point selection (b) The Frenet frame on tubular structure (C) Surface swept from the frames.

Tubular Segment Reconstruction Frenet-Serret frames are used to thicken or namely frame the tubular and bifurcate skeleton. For one particular Bézier

4

segment $Z_i(t)$ $t \in [0, 1]$ (Fig. 1(b)), the unit tangent $T_i(t)$, unit normal $N_i(t)$ and unit bi-normal $B_i(t)$ are given by:

$$T_i(t) = \frac{Z'_i(t)}{\|Z'_i(t)\|}, B_i(t) = \frac{Z'_i(t) \times Z''_i(t)}{\|Z'_i(t) \times Z''_i(t)\|}, \text{ and } N_i(t) = B_i(t) \times T_i(t), \quad t \in [0, 1]$$

One common problem occurring in Frenet framing is that in some cases, the binormal is not varying smoothly or well-defined at a singular point, e.g. when the curve is locally straight. We adopt a correction method which deploys a modified version of parallel transport [1] to tackle these singularities. The main purpose of parallel transport is to ensure one particular reference frame (a set of tangent T , normal N and binormal B) is transported as parallel to the previous frame as possible. This method can resolve singularities regardless of the local curvature throughout the curve. A numerical method for computing the frame is used. For two consecutive tangents, the axis perpendicular to both of them is used as the rotation axis. In this sense, the frame f_{i+1} can be computed by rotating the previous frame f_i by the angle between the two tangents.

After all frames have been computed and corrected, we can generate a series of rings for final surface sweeping:

$$F_i(t, \theta) = B_i(t) + r(t)(\cos\theta N_i(t) + \sin\theta B_i(t)), \quad t \in [0, 1], \theta \in [0, 2\pi],$$

where $r(t)$ denotes the radius of the tubular or bifurcate frame. $r(t)$ can be determined from the original patient data. Fig. 1(c) shows the resultant surface-swept curve.

Bifurcation Reconstruction Bifurcation framing has to be handled separately. First, we reconstruct three Bézier segments based on the bifurcate point, C_0 , and the radius of three tubular segments r_1 , r_2 , and r_3 , respectively. The segment end points and the internal control points can be calculated by:

$$\begin{aligned} C_{11} &= C_0 + r_1/2, & C_{12} &= C_0 + r_1; \\ C_{21} &= C_0 + r_2/2, & C_{22} &= C_0 + r_2; \\ C_{31} &= C_0 + r_3/2, & C_{32} &= C_0 + r_3; \end{aligned}$$

Fig. 2 shows the creation of three bifurcate Bézier segments based on the bifurcate point. Then, the thickening of these segments can be done by sweeping three half tubular surfaces and re-triangulating the inner Bézier triangle. Figs. 3(a) shows an example of tubular grid structure. Fig. 3(b) shows a composite structure.

Tetrahedral Model Based on the geometric model, a dual-layered tetrahedral mass-spring model can be built. The outer layer is the grid-based surface mesh. The inner layer is generated by shooting rays from the centroid of every triangle (of vertices v_{ij} , where $j = 0, 1, 2$) towards its facet normal N_i until the layer depth has been reached. One bottom vertex is corresponding to one upper triangle. The position of this bottom vertex b_i can be given by $b_i = \frac{v_{i0} + v_{i1} + v_{i2}}{3} + N_i l$, where l denotes the layer depth. We can then attach each triangle vertex to

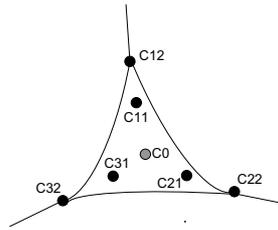


Fig. 2. Geometric model of the bifurcation grid

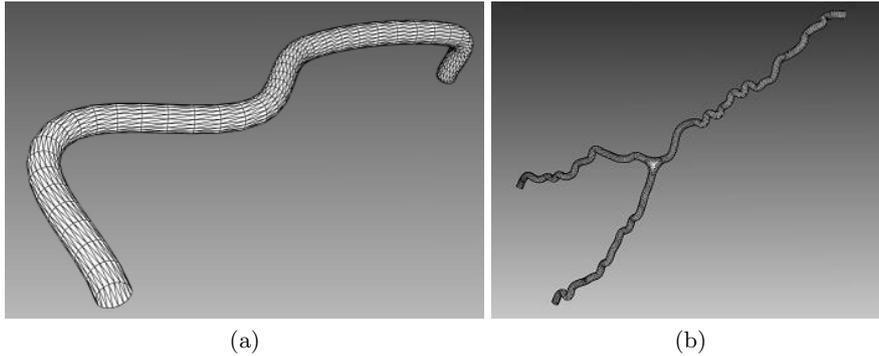


Fig. 3. Geometric model of the vascular model (a) the tubular structure, (b) a composite grid structure with both tubular segment and bifurcation.

the bottom vertex through three springs, i.e. $v_{i0} \rightarrow b_i$, $v_{i1} \rightarrow b_i$, and $v_{i2} \rightarrow b_i$. Fig. 4(a) demonstrates how the tetrahedral grid is reconstructed. A tetrahedral mesh structure can thus be built between the two layers of surface through spring connections. Triangulation is then carried out on these projected centroids to form a triangular mesh in the inner layer. Fig. 4 shows the layered deformable model for various different structures. The resultant tetrahedral grid structure is shown in Fig. 4(b) & (c). It is obvious that the same modeling mechanism applies to the reconstruction of the virtual devices. In our catheterisation simulation application, the virtual guide wire is modeled by a single-layered structure while the virtual catheter is modeled by a dual-layered structure.

3.4 PPU-based incremental-Voigt Model

To simulate a virtual catheterisation surgery in a realistic manner, efficient modeling of soft tissue deformation and virtual device mechanics is essential. Among various simulation tasks, soft tissue deformation and solid-solid interactions are the most computationally intensive parts in a catheterisation simulator since extensive physics computations are required. Although the currently available PPU do not yet provide a finite element analysis (FEA) solver, it does provide a built-in actor and effector mechanism for resolving solid mechanics, which can

6

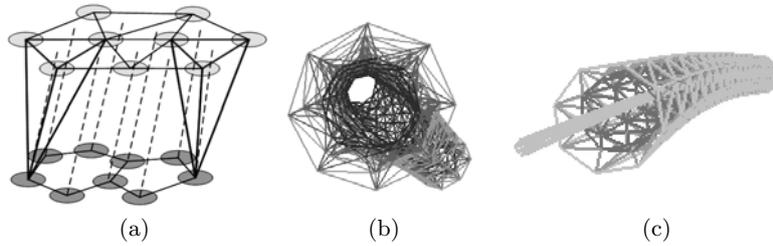


Fig. 4. Deformable model (a) layered construction of volumetric grid, (b) deformable grid structure of bifurcation structure, and (c) a single-layered wire is inserted into the bi-layer structured catheter.

be understood as a variation of mass-spring-damper implementation. This provides a convenient way for modeling linear solid mechanics. Most soft tissue, including the vessel wall, exhibit a non-linear stress-strain biomechanical behaviour [10][4][9]. To capture this behaviour interactively, we extend the PPU built-in model to an incremental-voigt model. In our model, two springs, each with an attached damper, are connected in parallel so that a biphasic biomechanical behaviour can be simulated.

As we are modeling the vessel tissue as a mass-spring system, data for the biomechanical properties of vessel tissue serves, therefore, as a reference for the behavior of the entire macroscopic mass-spring system, rather than the microscopic elasticity of individual single springs. To determine the micro-properties, we deploy our previously proposed optimization process [12] to compute the macroscopic elasticity of the whole mass-spring system which conforms with real tissue biomechanics.

4 Simulation and Discussion

The guiding of the catheter/wire is a relatively complex component of the real-time simulation process since it involves interior interactions between the vessel wall and catheter and on the exterior as well. The solid-solid interaction of the vascular structure and catheter/ guidewire is resolved by the built-in PPU collision detection mechanism. Various collision models have been adopted to speed up the overall simulation performance. In order to increase the system responsiveness, a dual-model approach has been adopted for handling collision detection of the outer and inner layers. We apply a mesh-based collision model for the inner layer structures while the collision model for the outer layer is based on a chain of spheres.

The PPU-based implementation has been compared with pure CPU-based implementation. Experiments are conducted on a Pentium 4 Dual Core 3.2HZ PC equipped with NVIDIA GeForce 8800. The physics accelerator used is AGEIA PhysXP1. Fig. 5 shows a comparison of the frame rate between the two implementations. Since the major physics computation is done on the effectors (i.e.

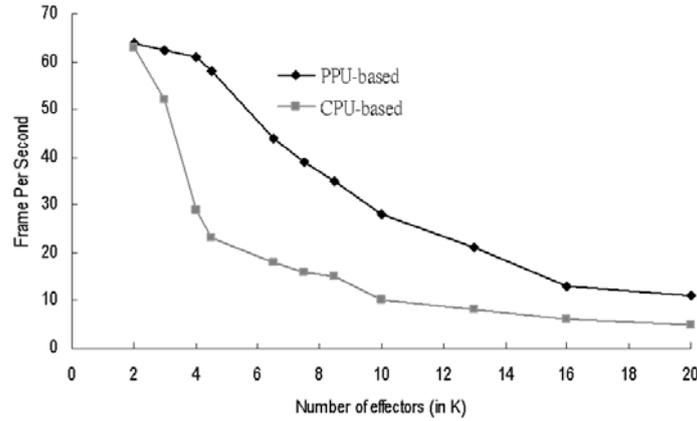


Fig. 5. Quantitative comparison between different size of deformable models.

the damper attached springs), we mainly compare the frame-rate against the total number of effectors being simulated. From the experimental results, we can observe that the performance gain resulted from PPU-based acceleration is significant. An increase in frame rates from 2-fold to 4-fold can be observed over non-PPU-based simulations. Interactive frame rates can still be guaranteed when the number of effectors is more than 10000.

The physics accelerator has been more prevalent in gaming industry in recent years. However, relatively little research has been conducted in exploring whether PPU is suitable for medical applications as well. In this work, we have tried to exploit the new consumer-level physics processing unit to accelerate catheterisation simulations. Preliminary results has demonstrated that PPU can greatly improve the speed performance while satisfying the realism requirement. PPU-based modules are found to be easily programmed and flexible for extension. Although finite element analysis (FEA) solver is not yet found on available PPU boards, the built-in actor and effector mechanisms in resolving solid mechanics is efficient for simulating solid-solid interaction in surgical simulations.

Although PPU has been found to be effective in improving the real-time responsiveness of medical simulations, one interesting finding regarding the PPU-based deformable model is that the frame rate drops dramatically when the number of effectors increases to a certain level. Such a situation occurs regardless whether the hardware-accelerated collision detection has been enabled or not. One possible reason is that the current hardware still relies on the relatively slow PCI bus, thus, the interactivity would be greatly affected once the maximum data transfer has been reached. Therefore, the next generation physics accelerator, which can be ported to higher throughput data bus e.g. AGP or PCX, would be demanded for applications involving high-quality simulation environment. Despite these limitations, the acceleration being achieved in the collision detection part is noteworthy. Once the programming capability of the

hardware-accelerated physics computation (may be in the form of an interface to various numerical solvers) has become more flexible, the use of PPU in the medical community shall be more widely accepted.

In conclusion, a new deformable modeling framework has been proposed for the medical simulation of catheterisation procedures. The framework is integrated with PPU so that interactive responsiveness can be achieved. Experimental results have demonstrated the robustness of our proposed geometric deformable vessel modeling framework. The next step of our work would be the modeling of other vascular interventional devices such as stent, balloon for simulation of angioplasty and stent implantation. In future work, we shall investigate the possibility of exploiting PPU to accelerate solid-fluid interaction. This would be important for simulating blood-tissue interaction as well as blood-device interaction within different catheterisation procedures.

Acknowledgement

This work was supported by a grant from the Innovation and Technology Commission of the Hong Kong Special Administrative Region (Project No. UIM/174). This work is also affiliated with the Virtual Reality, Visualization and Imaging Research Center at The Chinese University of Hong Kong as well as the Microsoft-CUHK Joint Laboratory for Human-Centric Computing and Interface Technologies.

References

1. BLOOMENTHAL, J. Calculation of reference frames along a space curve. *Graphics Gems* (1990), 567–571.
2. COTIN, S., DURIEZ, C., LENOIR, J., NEUMANN, P., AND DAWSON, S. New approaches to catheter navigation for interventional radiology simulation. In *Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2005 Part II* (October 2005), pp. 534–542.
3. FIEDLER, M. J., CHEN, S. J., JUDKINS, T. N., OLEYNIKOV, D., AND STERGIU, N. Virtual reality for robotic laparoscopic surgical training. In *Proceedings of Medicine Meets Virtual Reality (MMVR15)* (February 2007), pp. 127–129.
4. FUNG, Y. C. *Biomechanics: Mechanical Properties of Living Tissues*, second ed. Springer, Berlin Germany, 1993.
5. HARDERS, M., STEINEMANN, D., GROSS, M., AND SZEKELY, G. A hybrid cutting approach for hysteroscopy simulation. In *Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2005* (October 2005), pp. 567–574.
6. HENG, P. A., CHENG, C. Y., WONG, T. T., XU, Y. S., CHUI, Y. P., CHAN, K. M., AND TSO, S. K. A virtual reality training system for knee arthroscopic surgery. *IEEE Transactions on Information Technology in Biomedicine* 8, 2 (June 2004), 217–227.
7. IKEDA, S., ARAI, F., FUKUDA, T., NEGORO, M., IRIE, K., AND TAKAHASHI, I. An in vitro patient-tailored model of human cerebral artery for simulating endovascular intervention. In *Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2005 Part I* (October 2005), pp. 925–932.

8. KIM, W. S., WOO, H. S., AHN, W., LEE, K., CHO, J. H., LEE, D. Y., AND YI, S. Y. Non-clinical evaluation of the kaist-ewha colonoscopy simulator ii. In *Proceedings of Medicine Meets Virtual Reality (MMVR15)* (February 2007), pp. 214–216.
9. MAUREL, W., WU, Y., THALMANN, N. M., AND THALMANN, D. *Biomechanical Models for Soft Tissue Simulation*. Springer, Berlin Germany, 1998.
10. NEMIROVSKII, Y. V., AND KHEINLOO, M. L. States of stress in multilayer spherical vessels, cylindrical tubes, and circular disks of linear viscoelastic material. *Mechanics of Composite Materials* 9, 1 (1973), 94–99.
11. NOWINSKI, W. L., AND CHUI, C. K. Simulation of interventional neuroradiology procedures. In *Proceedings of the International Workshop on Medical Imaging and Augmented Reality (MIAR)* (2001), pp. 87–94.
12. PANG, W. M., QIN, J., CHUI, Y. P., WONG, T. T., HO, S. Z. M., AND HENG, P. A. PPU-accelerated medical simulation for skin deformation using chinese visible human data. In *Proceedings of Computer Assisted Radiology and Surgery (CARS 2007), 21st International Congress and Exhibition* (Berlin, Germany, June 2007). accepted for publication.
13. POSTON, T., WONG, T. T., AND HENG, P. A. Multiresolution isosurface extraction with adaptive skeleton climbing. *Computer Graphics Forum* 17, 3 (September 1998), 137–148.
14. SZCZERBA, D., AND SZEKELY, G. Simulating vascular systems in arbitrary anatomies. In *Proceedings of Medical Image Computing and Computer-Assisted Intervention (MICCAI) 2005 Part II* (October 2005), pp. 641–648.
15. TSAO, Y., AND FU, K. A parallel thinning algorithm for 3-d picture. In *Proceedings of Computer Vision, Graphics, and Image Processing* (1981), vol. 17, pp. 315–331.

3D FEM/XFEM-based Biomechanical Brain Modeling for Preoperative Image Update

Lara M. Vigneron¹, Romain C. Boman², Jean-Philippe Ponthot²,
Pierre A. Robe³, Simon K. Warfield⁴, and Jacques G. Verly¹

¹ Department of Electrical Engineering and Computer Science,

² Department of Mechanical Engineering,

³ Department of Neurosurgery, University Hospital,
University of Liège, Liège, Belgium;

⁴ Computational Radiology Laboratory, Children's Hospital,
Harvard Medical School, Boston, MA, USA.

Abstract. We present an end-to-end system for updating 3D preoperative images in the presence of brain shift and successive resections. The tissue discontinuities due to resections are handled via the eXtended Finite Element Method (XFEM), which has the appealing feature of handle arbitrarily-shaped discontinuity without any remeshing. The main novelty of the paper lies in the use of XFEM in 3D.

1 Introduction

The main goal of brain surgery is to remove as much as possible of lesional tissues, while avoiding contacts with eloquent areas and white matter fiber tracts. Surgery is planned on the basis of preoperative images of multiple modalities, such as CT, sMRI, fMRI, PET, DTI, and is generally performed using an image-guided navigation system that relates the 3D preoperative images to patient coordinates. However, throughout surgery, the brain deforms, mostly as a result of the leakage of the cerebrospinal fluid out of the skull cavity and of surgical acts, such as retraction and resection. As surgery progresses, preoperative images become progressively less representative of the brain, and navigation accuracy decreases. One solution is to evaluate brain deformations from reduced-quality intraoperative images acquired at several critical points during surgery, and to update, i.e. to deform, all high-quality preoperative images using a nonrigid registration.

One category of nonrigid registration techniques uses physics-based models. Prior to surgery, a biomechanical brain model specific to the patient is built from preoperative images. The model consists of a 3D volume mesh and one or more mechanical-behavior laws. A number of key anatomical landmarks are extracted and tracked through successive intraoperative images. The biomechanical model is deformed, generally based on the Finite Element Method (FEM), in accordance with the displacement fields of these landmarks. The resulting deformation is then used to update the preoperative images.

Most studies of brain deformation based on biomechanical models have focused on the early stages of surgery, i.e. prior to any significant deformation and any cut [1–4]. The reported accuracy for deformation prediction is about 1 voxel. The situation becomes more complex when the surgeon performs cuts, retractions,

or resections [1, 5, 6], the last two necessarily involving a cut. The main difficulty associated with a cut is the discontinuity it implies in the tissue. Indeed, FEM cannot handle such discontinuities directly and, consequently, FEM has to be used in conjunction with mesh adaptation [7] or remeshing [8] techniques. However, it is likely that current remeshers, mainly developed for mechanical engineering applications, would not work properly on irregular objects such as a brain (which is furthermore extracted from an image), especially in a automatic mode. While human intervention may improve results, the time required is significant and unpredictable, which makes it unsuitable for surgery where a timely response is essential.

Besides FEM, other methods have also been employed in the medical field to model tissue discontinuities, like the boundary element method (BEM) [9] and meshless methods [10, 11]. We propose an approach based on the eXtended Finite Element Method (XFEM or X-FEM) [12]. This method allows the object to be modeled by finite elements without explicitly meshing the discontinuities, which can then be located arbitrary with respect to the underlying finite-element mesh. Here, we describe, and report on the performance of, a 3D FEM- and XFEM-based end-to-end system capable of updating preoperative images in the presence of brain shift followed by successive resections [13]. The main novelty is that the problem is treated in 3D.

The structure of the paper is as follows. In Sect. 2, we introduce the basic principles of FEM and XFEM. In Sect. 3, we describe our preoperative image-update system and underlying algorithms. In Sect. 4, we show our results for one patient case, while, in Sect. 5, we validate our results. In Sect. 6, we conclude and point out future work.

2 Basic principles of FEM and XFEM

We have to solve the static problem of finding the displacement field that corresponds to the deformation of a solid (a brain in the present case), subjected to external forces. With FEM, the solid is discretized into a mesh, i.e. into a set of finite elements interconnected by nodes, and the displacement field is approximated by

$$\mathbf{u}^{FEM}(\mathbf{x}) = \sum_{i \in I} \varphi_i(\mathbf{x}) \mathbf{u}_i, \quad (1)$$

where I is the set of nodes, the φ_i 's are the nodal shape functions (NSFs), and the \mathbf{u}_i 's are nodal degrees of freedom (DOFs).

In Eq. (1), each NSF $\varphi_i(\mathbf{x})$ is defined as being continuous over each FE, implying the same property for the displacement field $\mathbf{u}^{FEM}(\mathbf{x})$. Furthermore, the displacement \mathbf{u}_i at any node can only take a single value. Consequently, the handling of a discontinuity with FEM requires one to align the discontinuity with element boundaries and to duplicate the nodes lying on these boundaries. These operations can be performed by mesh adaptation or remeshing.

XFEM [12, 14] handles a discontinuity by allowing the displacement field to

be discontinuous within mesh elements. Arbitrarily-shaped discontinuities can then be modeled without any remeshing. The XFEM displacement field is approximated by generalizing the FEM displacement field (1) with

$$\mathbf{u}^{XFEM}(\mathbf{x}) = \sum_{i \in I} \varphi_i(\mathbf{x}) \mathbf{u}_i + \sum_{i \in J} \varphi_i(\mathbf{x}) \sum_{j=1}^{n^{E_i}} g_j(\mathbf{x}) \mathbf{a}_{ji}. \quad (2)$$

The first term corresponds to the FEM displacement field approximation (1), where I is the set of nodes, the $\varphi_i(\mathbf{x})$'s the FEM NSF's, and the \mathbf{u}_i 's the nodal FEM DOFs. The key of XFEM is the "enrichment" that adds a number, n^{E_i} , of DOFs \mathbf{a}_{ji} to each node i of set J , which is the subset of nodes of I whose support is intersected by the discontinuity. These DOFs are multiplied by the NSF's $\varphi_i(\mathbf{x})$ and the discontinuous functions $g_j(\mathbf{x})$. The simplest choice for the $g_j(\mathbf{x})$'s is a piecewise-constant function that changes sign at the discontinuity, i.e. the Heaviside function

$$H(\mathbf{x}) = \begin{cases} 1 & \text{for } (\mathbf{x} - \mathbf{x}^*) \cdot \mathbf{e}_n > 0 \\ -1 & \text{for } (\mathbf{x} - \mathbf{x}^*) \cdot \mathbf{e}_n < 0, \end{cases} \quad (3)$$

where \mathbf{x} is a point of the solid, \mathbf{x}^* is the point on the discontinuity that is the closest to \mathbf{x} , and \mathbf{e}_n is the outward normal to the discontinuity at \mathbf{x}^* .

Figure 1 illustrates the use of 2D XFEM for a mesh cut horizontally, with each part being subjected to a distinct translation. This example shows that the two parts of the mesh can move independently without one having to remesh the object along the discontinuity.

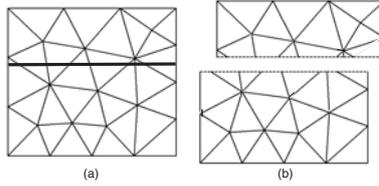


Fig. 1. Example of 2D XFEM calculation. **(a)** FE mesh and discontinuity geometries prior to deformation. **(b)** Deformation results provided by XFEM when top and bottom parts are subjected to distinct translations.

3 Methods

The inputs are (1) a set of multimodality preoperative images and (2) a sequence of intraoperative MR (iMR) T1 images, acquired with a 0.5 Tesla intraoperative GE Signa scanner⁵. The protocol of iMR acquisition is defined such that the 1st iMR image is acquired prior to the opening of the skull. Before surgery, a patient-specific biomechanical model is built from the set of preoperative images. Once the 1st iMR image is acquired, the set of preoperative images and the biomechanical model are brought into the intraoperative coordinate system by a rigid registration. Without any loss of generality, we deal with the problem of updating the 1st iMR image, which thus plays the role of a fictive preoperative

⁵ iMR image size is $256 \times 256 \times 60$ voxels; voxel size is $0.9375 \times 0.9375 \times 2.5$ mm.

image. Except for the rigid registration between the preoperative images, the biomechanical model, and the 1st iMR image, all key aspects of the system can be discussed and illustrated.

To build the biomechanical model, we proceed as follows. From the 1st iMR image (once again, as a substitute for true preoperative images), we manually segment out the cortex and the tumor using 3D SLICER⁶, and smooth both regions. Then, we mesh the cortex surface with triangles using ISOSURF⁷. Besides, we define a set of points lying on the tumor boundary. Based on the cortex triangular mesh surface and this set of points, we mesh the cortex volume with tetrahedra using TETGEN⁸. We do not create a surface mesh of the tumor boundary because TETGEN cannot handle internal surfaces. Finally, we assign to this mesh a homogeneous linear elastic ($E = 7kPa$, $\nu = 0.45$)⁹ behavior law.

To drive the deformation of the biomechanical model, we track the surface deformation of the cortex and tumor through successive pairs of iMR images. For this purpose, we segment out the cortex and tumor from each iMR image using 3D SLICER and smooth both regions. The matching of corresponding surfaces is performed with an active surface algorithm [1]. The resulting surface displacements fields are applied to the biomechanical model, and its deformations are computed using FEM or XFEM, depending upon the type of circumstances, e.g. brain shift or resection. We developed our FEM and XFEM codes within the software METAFOR¹⁰. For each (X)FEM computation, we assume a zero initial stress. We perform FEM computations in a nonlinear formulation (large deformations) because this has been shown to provide better results than in a linear formulation (small deformations) [15], and to reduce the problem of element flipping. By contrast, we perform XFEM computations in a linear formulation, because a XFEM nonlinear formulation is not yet available. The set of preoperative images (the 1st iMR image in our case) are deformed, i.e. warped, based on the volume displacement field of the biomechanical model. We use VTK¹¹ and ITK¹² for all image processing.

4 Results

The top row of Fig. 3 shows a sequence of five iMR images. The 1st image was acquired prior to the opening of the skull. The 2nd image was acquired after the opening of the skull and dura, and shows some brain shift. The 3rd and 4th images were acquired after each of two successive resections. The 5th image was acquired at the end of the surgery after the 3rd resection, and shows some

⁶ www.slicer.org/

⁷ <http://mi.eng.cam.ac.uk/~gmt11/software/isosurf/isosurf.html>

⁸ <http://tetgen.berlios.de/>

⁹ Experiments show that our current implementation appear sensitive to the values of the parameters, in particular that of E : some values of E have indeed been observed to cause element flipping.

¹⁰ <http://garfield.ltas.ulg.ac.be/oo.meta/>

¹¹ www.vtk.org

¹² www.itk.org

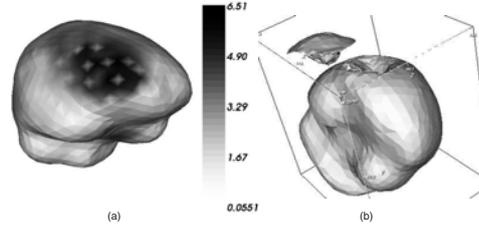


Fig. 2. (a) Initial active surface with color levels corresponding to the magnitude of the final displacements. The location of the brain shift is clearly visible. (Discussed in Sect. 4.1.) (b) Deformed mesh resulting from the XFEM-based modeling of the 2^{nd} resection. (Discussed in Sect. 4.3.)

postoperative brain shift. The modeling of brain shift, 1^{st} , 2^{nd} , and 3^{rd} resection are performed using different techniques, as detailed below.

4.1 Modeling of brain shift via FEM

To model brain shift, we estimate the surface displacement field of the cortex and tumor, and use them to deform the biomechanical model. For the cortex, the initial active surface corresponds to the surface of the biomechanical model. Once the result of the active surface algorithm is obtained (Fig. 2a), the displacements of the active surface nodes are directly applied to the cortex nodes. For the tumor, the displacements are applied differently because the tumor is not defined by a surface mesh. An active surface computation is performed on the healthy brain (defined as the whole brain minus the tumor). Then, we apply, to the nodes of the volume mesh located along the tumor boundary, displacements extrapolated from the surface displacement field of the active surface along the tumor boundary. In the present case, no tissue discontinuity is involved, so the volume deformation is computed via FEM. The resulting volume displacement field is used to warp the whole brain image extracted from the 1^{st} iMR image. The result is shown in Fig. 3.2b.

4.2 Modeling of 1^{st} resection via FEM

Matching two surfaces to get a displacement field makes sense only if the surfaces correspond to the same physical entity. In the case of resection, we cannot rely on the totality of the cortex surface, since a part of it is now missing. Consequently, we evaluate the displacement field for the combined surfaces of the intact cortex and of the tumor, which effectively constitutes the boundary surface of the healthy brain.

Based on the 2^{nd} and 3^{rd} iMR images, we cannot determine the volume and location of the tissue removed by the resection. This is because the 3^{rd} iMR image shows the combined effect of tissue removal and subsequent deformation. In fact, because the 2^{nd} and 3^{rd} iMR images do not show the same volume of brain tissue, the problem of modeling resection and brain shift is fundamentally different. Nevertheless, we decided to model the 1^{st} resection by still relying on the displacement fields of key surfaces, here the healthy-brain boundary, to deform

the biomechanical model. This indeed appears to be the only reliable information concerning the deformation due to resection that we can extract from the 2^{nd} and 3^{rd} iMR images. Consequently, we do not model explicitly the removal of tissue, but we model directly the deformation resulting from it, without introducing any tissue discontinuity. Using the displacement field of the healthy brain boundary, we compute the deformation of the biomechanical model via FEM. Then, using the resulting volume displacement field, we warp the fictive preoperative image in its current state of update (Fig. 3.2b), in the same way as for brain shift. The resulting image is shown in Fig. 3.3b, which is now registered to the 3^{rd} iMR image (Fig. 3.3a), except outside of the healthy-brain boundary, i.e. for the tumor. Finally, we alter the resulting image to reflect the effect of resection. For this, we assign the background color to the voxels of Fig. 3.3b corresponding to the resected tissue volume “absent” in the 3^{rd} iMR image (Fig. 3.3a). The result of the warping with resection, performed by masking the warped image (Fig. 3.3b) with the segmentation of the whole brain from the 3^{rd} iMR image (Fig. 3.3a), is shown in Fig. 3.3c.

4.3 Modeling of 2^{nd} resection via XFEM

The significant feature of the 2^{nd} resection is that some tissue has already been removed by the 1^{st} resection, which means that this tissue cannot have any physical influence on subsequent brain deformations because it does not “exist” anymore. Consequently, the 1^{st} resection must be reflected in the biomechanical model. Recall that, to model the 1^{st} resection, the biomechanical model has been deformed to be registered to the 3^{rd} iMR image. So, using the 3^{rd} iMR image, we can define the boundary of the 1^{st} resection, i.e. the tissue discontinuity to include in the biomechanical model. With a FEM-based biomechanical model, we would remesh the model to take into account the discontinuity. Then, we would just remove the part of the mesh corresponding to the resected tissue, and use the other part of the mesh to model the 2^{nd} resection. Instead, with XFEM, we enrich the nodes whose support is intersected by the discontinuity with XFEM Heaviside DOFs. Consequently, when the XFEM-based biomechanical model deforms, the part corresponding to tissue removed by the 1^{st} resection has no influence on the deformation of the remaining part of the brain.

Except for the fact the biomechanical model is deformed with XFEM rather than FEM, the modeling of 2^{nd} resection is identical to that of the 1^{st} resection. The biomechanical model is deformed in accordance with the displacement field of the healthy-brain boundary evaluated from the 3^{rd} and 4^{th} iMR images. Fig. 2b shows the deformed mesh, result of the XFEM computation. The bottom part of the mesh, representing the tissue remaining after the 1^{st} resection, was deformed according to the displacement field of the healthy brain boundary, while the top part, representing the tissue removed by the 1^{st} resection, was subjected to a translation, but only for visualization purposes. The two parts of the mesh could indeed overlap around the discontinuity after deformation of the bottom part. Using the XFEM volume displacement field, we warp the fictive preoperative image in its current state of update (Fig. 3.3b). The resulting image is shown in Fig. 3.4b, while the result of the warping with resection, performed

by masking Fig. 3.4b with the segmentation of the whole brain from the 4th iMR image (Fig. 3.4a), is shown in Fig. 3.4c.

4.4 Modeling of 3rd resection via XFEM

If we had complete freedom, we would model the 3rd resection and postoperative brain shift based on the bottom part of the output mesh of the 2nd resection modeling (Fig. 2b). However, even though the mesh is displayed as two separate parts, it is, in fact, a single entity. Indeed, a main feature of XFEM is to be able to handle the effect of a discontinuity without modifying the underlying mesh, i.e. without remeshing. For modeling the 2nd resection, the edges of elements straddling the discontinuity have been “cut” and their nodes moved apart. However, it is not possible to start an XFEM calculation with a mesh already deformed because the finite elements are not supposed to be already cut. Consequently, XFEM, expressed in linear formulation, does not permit us to proceed with a new discontinuity from a configuration already deformed by XFEM. Indeed, we should use a nonlinear formulation of XFEM [16] to proceed as desired.

For temporarily avoiding going nonlinear, we devised a new method that continues to preserve the spirit of XFEM, which is mainly to avoid remeshing in the presence of a discontinuity. However, this method implies additional processing. The idea of our method is “to reconnect the pieces” of the deformed mesh (Fig. 2b), but without remeshing, in such a way we can further resect it. Since we already have the correct deformed coordinates for the bottom part of the mesh, we need to find a way “to restore” a meaningful top part. We cannot simply lower the top part of the mesh, because it would not fit the bottom part. Indeed, the bottom part has been deformed during the modeling of the 2nd resection, with consequence that the discontinuity surface of the bottom part has changed shape. While this might not be the best from a computational standpoint, our current solution is to use, for the coordinates of the nodes of the top part of the mesh, the coordinates that we would had got if we had modeled the 2nd resection based on FEM, rather than on XFEM, i.e. without taking into account the removal of tissue by the 1st resection. These coordinates are erroneous, but this is not an issue, since these nodes will again be resected, given that the 3rd resection is necessarily deeper than the 2nd resection. We must be careful to ensure that the restored elements have good aspect-ratios. FEM is a way to achieve this, although there is no guarantee that good results will be obtained for the composite elements straddling the discontinuity.

After the reconnection of the mesh, the modeling of the 3rd resection is identical to that of the 2nd resection. The tissue discontinuity due to the 2nd resection is defined from the 4th iMR image, and used to enrich the nodes of the reconnected mesh. The biomechanical model is deformed based on XFEM in accordance with the displacement field of the healthy-brain boundary computed from the 4th and 5th iMR images. One significant feature of the procedure described for modeling the 3rd resection, i.e. mesh reconnection followed by model deformation, is that it can be applied iteratively for each subsequent resection visible on successive iMR images, no matter how many there are.

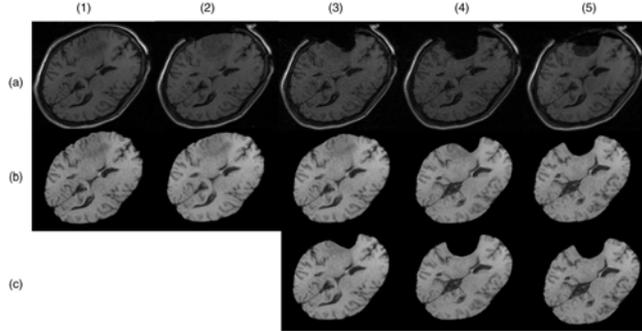


Fig. 3. (1st row) Sequence of five input iMR images. (1a) is used as a fictive preoperative MR image. (2nd and 3rd rows) Results of the warping of the fictive preoperative image. Details are in text.

For this patient case, a simplification for the modeling of the 3rd resection can be made because, by the time the 5th iMR image is acquired, the resection is complete. This means that we actually only need to compute the volume displacement field of the healthy brain tissue. Since we applied displacements exactly to the external surface of the healthy brain volume, the results would be the same with FEM and XFEM. Using the FEM volume displacement field, we warp the fictive preoperative image in its current state of update (Fig. 3.4b). The resulting image is shown in Fig. 3.5b, while the result of the warping with resection, performed by masking Fig. 3.5b with the segmentation of the whole brain from the 5th iMR image (Fig. 3.5a), is shown in Fig. 3.5c.

5 Validation

The visual comparison of the topmost and bottommost image in each column of Fig. 3 appears satisfying up to the 3rd column. In the 4th and 5th columns, which corresponding to the 2nd and 3rd resections, we observe a positional discrepancy near the ventricles. To evaluate our results, we selected for each column a particular slice from of the topmost image (masked to keep only the brain) and the bottommost image, we extracted their edges using a Canny edge detector, and we evaluated the nonrigid registration by computing the 95% Hausdorff distance between the edges of these corresponding pairs of slices. For a perfect match, this distance should be zero. To evaluate the benefit of performing the nonrigid registration, as opposed to the rigid registration of preoperative images that neurosurgeons typically perform, we also computed the 95% Hausdorff distance for the edges of the whole brain region of the corresponding slice of the 1st iMR image, which plays the role of a fictive preoperative image, and the edges of the slice of the topmost images (as before). When a resection was actually involved in the deformation, we did not use the 1st iMR image directly because the 95% Hausdorff distance would be artificially high at the location where the cortex had been resected. Consequently, we used, for the 3rd, 4th, and 5th columns, the 1st iMR slice, masked with the whole brain region of the target image, i.e. the image showing the current deformation. The values of the 95% Hausdorff distance are

	Brain shift	1 st resection	2 nd resection	3 rd resection
(a) Deformation of 1 st iMR	3.38/2.81	3.38/3.38	2.96/3.75	3.38/3.75
(b) Deformation of initial iMR	3.38/2.81	3.38/2.96	2.96/2.96	3.38/3.38

Table 1. Comparison of 95% Hausdorff distance (mm) *without/with* nonrigid registration. (a) Case where the 1st iMR image is successively deformed. (b) Case where the initial iMR image, i.e. the iMR image prior to the deformation being considered, is deformed for each deformation. The comparison of (a) and (b) shows the effect of the propagation of registration errors.

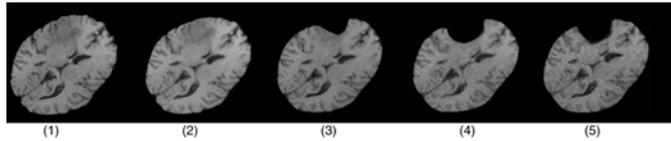


Fig. 4. (1-5) Results of deforming the initial iMR image (defined in text) for each stage. These should be compared to the bottommost images in Fig. 3 where the 1st iMR image was deformed in successive stages.

shown in row (a) of Tab. 5. We see that the distance increases for the deformations corresponding to 2nd and 3rd resections when a nonrigid registration is performed. This can be partly explained by the positional discrepancy near the ventricles, and this points out the need to include ventricle surface deformations in the biomechanical model. In addition, our nonrigid registration being based on the surface deformations of the cortex and tumor, it is strongly dependent on the quality of their segmentation. Another explanation for the behavior of the 95% Hausdorff distance could be the propagation of registration errors across the 4 successive deformations. To evaluate this effect, we deformed the initial iMR image, i.e. prior to the deformation being considered, for each stage of deformation, rather than deforming the 1st iMR image in successive stages. The resulting images are shown in Fig. 4. As compared to the bottommost images in Fig. 3, the positional discrepancy near the ventricles has clearly decreased. The 95% Hausdorff distances of the selected slices in the images of Fig. 4 are shown in row (b) of Tab. 5. We see that the value of 95% Hausdorff distance now decreases for the brain shift and 1st resection, or remains identical for the 2nd and 3rd resection, where the nonrigid registration has no effect.

6 Conclusions

We have developed and tested an end-to-end system for updating preoperative images using intraoperative images in the presence of successive deformations due to brain shift and resections. (We have, however, skipped the important issue of the initial rigid registration.). The nonrigid registration technique used is based on a biomechanical model driven by the deformations of key surfaces, extracted and tracked from successive intraoperative images. We use surface deformations since they can generally be extracted more reliably than volume deformations [6] can, as a result of the reduced-quality of intraoperative images. However, the performance of the system is contingent upon the quality of segmentation of key objects. In that regard, a contrast agent would be helpful.

The key and novel feature of our system is the use of 3D XFEM to handle

tissue discontinuities without having to resort to remeshing. For a biomechanical model as simple as the one used here for illustration purposes, it would be straightforward to build a new model for each deformation. However, our goal is to include as much information as possible from preoperative images and, as a result, to use our system with complex biomechanical models. The modeling of discontinuities will then surely involve some remeshing, and in that case, XFEM could be a good alternative. While dealing with a one-time resection is relatively straightforward with XFEM, we have found that XFEM expressed in a linear formulation cannot deal with successive resections. However, this paper shows an effective method for going around this problem. The results obtained give us confidence that the approach proposed works correctly. A nonlinear formulation of XFEM should provide a more systematic approach for dealing with successive resections. In future work, we will examine the XFEM nonlinear formulation.

Acknowledgements This investigation was supported in part by the “Fonds National de la Recherche Scientifique (F.N.R.S.)”, Brussels, Belgium, by NSF grant ITR 0426558, a research grant from CIMIT, by NIH grants R03 CA126466 and R01 RR021885.

References

1. Ferrant, M., et al.: Serial registration of intra-operative MR images of the brain. *Med. Image Anal.* **6** (2002) 337–359
2. Miller, K., et al.: Towards computing brain deformations for diagnosis, prognosis and neurosurgical simulation. *J. Mech. Med. Biol.* **5**(1) (2005) 105–121
3. Paulsen, K., et al.: A computational model for tracking subsurface tissue deformation during stereotactic neurosurgery. *IEEE Trans. Biom. Eng.* **46**(2) (1999) 213–225
4. Warfield, S.K., et al.: Capturing intraoperative deformations: research experience at Brigham and Women’s Hospital. *Med. Image Anal.* **9**(2) (2005) 145–162
5. Miga, M.I., et al.: Modeling of retraction and resection for intraoperative updating of images. *Neurosurgery* **49**(1) (2001) 75–84
6. Clatz, O., et al.: Robust non-rigid registration to capture brain shift from intra-operative MRI. *IEEE Trans. Medical Imaging* **24**(11) (2005) 1417–1427
7. Serby, D., et al.: A new approach to cutting into finite element models. *MICCAI’01* **2208** (2001) 425–433
8. Nienhuys, H.W.: Cutting in deformable objects. PhD thesis, Institute for Information and Computing Sciences, Utrecht University (2003)
9. Ecabert, O., et al.: Brain shift correction based on a boundary element biomechanical model with different material properties. *MICCAI’03* **2878** (2003) 41–49
10. De, S., et al.: On the method of finite spheres in applications: Towards the use with ADINA and in a surgical simulator. *Computational Mechanics* **31** (2003) 27–37
11. Liu, H., Shi, P.: Meshfree representation and computation: Applications to cardiac motion analysis. *IPMI* **18** (2003) 560–72
12. Moës, et al.: A finite element method for crack growth without remeshing. *Int. J. Numer. Meth. Engng.* **46** (1999) 131–150
13. Vigneron, L.M., et al.: XFEM-based modeling of successive resections for preoperative image updating. *SPIE Medical Imaging* **5367** (2006) 735–746
14. Sukumar, N., Prévost, J.H.: Modeling quasi-static crack growth with the extended finite element method. Part I: Computer implementation. *Int. J. Solids and Structures* **40**(26) (2003) 7513–7537
15. Hawkins, T., et al.: Comparison of constitutive models of brain tissue for non-rigid image registration. 2nd Workshop on Computer Assisted Diagnosis and Surgery (2006)
16. Legrain, G., et al.: Stress analysis around crack tips in finite strain problems using the extended finite element method. *Int. J. Numer. Meth. Engng.* **63**(2) (2005) 290–314

Spherical Harmonics 3D Active Contours for Membrane Bilayer- Bound Surfaces

Khaled Khairy*, Jacques Pecreaux*, Jonathon Howard

Max-Planck Institute of Molecular Cell Biology and Genetics, Dresden, Germany

* These authors contributed equally to this work.

Abstract: Active contours is a powerful image segmentation technique based on simultaneously optimizing the overlap of a surface contour with the intensity image (external energy) on the one hand, and a constraining image-independent penalty based on the first and second derivatives of the contour (internal energy) on the other. Although the above form is applicable to a wide class of images, including prior information about the topology and smoothness as well as insights from physical theories regarding specific material properties of the object under study are expected to result in faster and more accurate segmentations.

In this work we extend the formulation of the active contour internal energy for the common case of 3D-imaging lipid-bilayer membrane-bound objects of topological genus zero. Examples include organelles, cells and artificial vesicles. In the non-supervised method presented here, the internal energy takes into account membrane bending elasticity as well as constraints imposed by the fact that the two bilayer leaflets are allowed to slide relative to each other. An additional topology constraint is implicitly accounted for by using a spherical harmonics parametric contour representation. The balance between internal and external energies (i.e. the regularization parameter) is determined using the L-curve method.

To ensure convergence and numerical stability a good starting guess for the contour is essential. We show in detail a method, that also makes use of the L-curve, for calculating this guess, and apply the complete procedure to a representative synthetic data set using realistic physical quantities based on membrane biophysical theories and known experimental results.

1 Introduction

Active contours [1] have been used extensively as a means for automatic and accurate image segmentation. The main idea is to allow a contour that is superimposed on the image to change shape until it minimizes a cost functional (E) that takes the form

$$E = E_{\text{internal}} + \lambda E_{\text{external}} \quad (1)$$

with E_{internal} a shape prior that enforces smoothness constraints and typically depends on the first and higher derivatives of the contour, E_{external} a quantity sensitive to the amount of overlap of the contour with the intensity values of the image, and λ a parameter that balances the relative importance of the two terms. Equation 1 is generally applicable in 2 and 3 space dimensions. Implementations of the method differ in the choice for λ , the exact forms for E_{internal} and E_{external} , and the mathematical description of the contour. Subjective choices of λ and physically unrealistic forms for E_{internal} result in convergence to suboptimal shapes. This problem has partially been alleviated by the introduction of statistical shape models (see for example [2]), which constrain the contour search to a subset of the shapes allowed by the shape description. However, statistical shape models require a training set and may impose constraints that are too stringent for detecting significant variations in shape. Most importantly, they generally do not provide a natural connection to the underlying biophysics that gave rise to a particular shape. Therefore we will address the issue of segmenting 3D intensity images in the context of Equation 1, when the object under study has—at least partially—known mechanical properties.

1.1 Incorporation of Prior Information into the Active Contour Procedure

In this paper, we focus on objects that are bounded by lipid membrane bilayers. Examples include organelles, cells and artificial bilayer vesicles (liposomes). For these objects biophysical theories [3] and experimental mechanical measurements [4-6] provide prior information that can be incorporated into our edge-finding procedure. The theoretical predictions alone are already able to reproduce shapes that closely resemble experimental observations both qualitatively [7] and quantitatively [8] in cases where the membrane bilayer is expected to primarily determine the morphology (Fig.1a).

E_{internal} of Equation 1 is an appropriate vehicle for incorporating such prior information [9], which primarily depends on the membrane bending elasticity and the resistance of the area difference between the bilayer leaflets to deviate from some preferred value (Fig.1b). Fig.1a shows some of the “default” shapes that the active contour will tend to when such an E_{internal} is minimized on its own.

Moreover, the topology of such biological objects is often restricted to that of the sphere. This piece of prior information we implicitly include by using a spherical harmonics parameterization (SHP) of the surface [10]. The spherical harmonics are the 3D equivalent of the Fourier series defined on the surface of a sphere (see below).

1.2 Other Issues: Regularization and the Starting Shape

To avoid artificially favoring E_{internal} over E_{external} or vice versa, we use an objective way for determining the regularization parameter λ in Equation 1, namely the L-curve method [11], which determines λ as the point of maximum curvature on the curve obtained when plotting the logarithm of E_{internal} vs. the logarithm of E_{external} when performing the optimization for a series of values of the regularization parameter.

However, before optimizing Equation 1, we need a good starting set of shape coefficients to improve convergence and numerical stability. To obtain such a set we map an initial surface triangulation (obtained for instance by using a marching cubes algorithm) to the unit sphere [12, 13]. In this article we introduce a method of spherical mapping that also benefits from the objectivity of the L-curve procedure.

Our complete method is illustrated on a noisy synthetic 3D intensity data set that simulates 3D fluorescence microscopy images of a human red blood cell discocyte. To our knowledge, this is the first demonstration of 3D active contours based on membrane biophysics.

2 Theory

2.1 Membrane-biophysics-derived (internal) Energies

Here we give the expressions used to calculate the internal (image-independent) energy of the contour, when we know that the observed object is bounded by a lipid-bilayer membrane. Theoretically the shape of a membrane bilayer contour is assumed to minimize an energy functional (E) that –among other possible contributions– includes the bending energy of the membrane (E_b) [14], and the area difference elasticity energy (E_{ADE}), which is the resistance of the area difference between the outer and inner leaflets (ΔA) to deviations from some preferred value (ΔA_o)[15-18].

$$E_{\text{internal}} = E_b + E_{ADE} \quad (2)$$

The first term is given by,

$$E_b = \frac{\kappa_b}{2} \oint_S (2H - C_o)^2 dA \quad (3)$$

where κ_b ($\sim 25 \times 10^{-20} \text{J}$ [5]) is the bending modulus, H is the local curvature, and C_o is the preferred curvature. C_o is a local term that depends on the local lipid composition and lipid molecule geometry (Fig.1b). The second term in Equation 1 is given by

$$E_{ADE} = \frac{\bar{\kappa}\pi}{2AD^2} (\Delta A - \Delta A_o)^2 \quad (4)$$

where A is the total surface area of the contour, D is the separation between the two bilayer leaflets and $\bar{\kappa}$ is a global elastic modulus. The relative importance of E_b vs. E_{ADE} is controlled by the ratio $\alpha = \bar{\kappa} / \kappa_b \approx 2 / \pi$ [7]. Equation 4 constitutes a global term, justified by the fact that the two bilayer leaflets are allowed to slide relative to each other, so the effect of—for example—adding lipid molecules to one of the leaflets, would be an instantaneous redistribution of this perturbation over the whole shape (Fig.1b). In the preceding Equations, all integrations are performed over the closed surface \vec{S} . C_o and ΔA_o cannot be independently determined and are mathematically not separable [19]. They enter the calculation through a unitless effective preferred area difference parameter $\Delta \bar{a}_o = \Delta A_o / A + \kappa_b DC_o / \pi \bar{\kappa}$. Typically $\Delta \bar{a}_o$ varies between 0.2 and -0.2, and considerably influences the default shape (see [7] and Figure 1a).

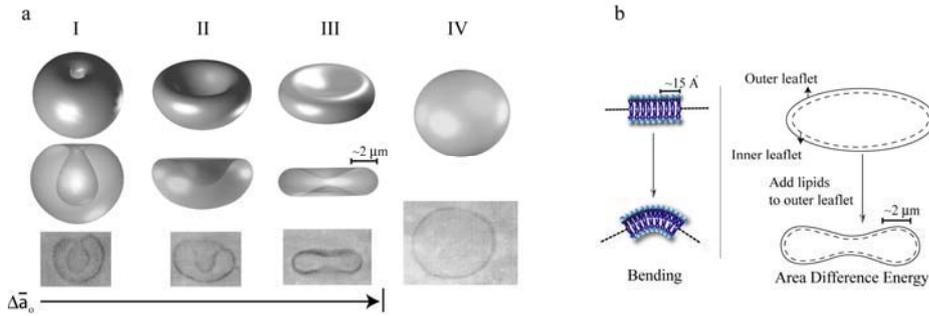


Fig.1. Shape energy and typical predicted shapes. (a) Theoretically predicted minimum energy shapes considering bending and area difference elasticities under constraints of total surface area and volume. Going from contour I to III, $\Delta \bar{a}_o$ the parameter responsible for the preferred curvature and preferred area difference between the two bilayer leaflets is increased, while keeping the area and volume constraint the same. Contour IV shows that the theory can also predict nearly spherical shapes given the appropriate ratio of area to volume. Images shown in the bottom row are reproduced from [3] **(b)** Schematic of lipid bilayer showing the two main energy contributions entering the calculation of shape energy; bending and area difference elasticities.

2.2 External energy

Our external energy term follows the Chan et Vese formulation [20] based on the Mumford-Shah approach [21]

$$E_{\text{external}} = \int_{\text{innerVolume}} \lambda_{\text{in}} (I - c_{\text{in}})^2 dV + \int_{\text{outerVolume}} \lambda_{\text{out}} (I - c_{\text{out}})^2 dV \quad (5)$$

where I is the image intensity, V the volume, c_{in} and c_{out} are the average intensities inside and outside the contour respectively, and λ_{in} and λ_{out} are hyperparameters.

2.3 Spherical harmonics surface parameterization

A function r of the spherical coordinates (θ, ϕ) may be represented as a series expansion,

$$r(\theta, \phi) = \sum_{L=0}^{\infty} \sum_{K=-L}^L C_{LK} y_{LK}(\theta, \phi) \quad (6)$$

where $0 < \theta < \pi$ and $0 < \phi < 2\pi$. The C_{LK} s are the expansion coefficients, indexed by the integers L and K with $-L \leq K \leq L$ and $0 \leq L \leq \infty$. $y_{LK}(\theta, \phi)$ are the spherical harmonics basis functions defined by

$$y_{LK}(\theta, \phi) = N_{LK} P_{L,K}(\cos \theta) \cos(K\phi) \quad \text{when } K \geq 0, \quad (7)$$

and

$$y_{LK}(\theta, \phi) = N_{LK} P_{L,K}(\cos \theta) \sin(|K|\phi) \quad \text{when } K < 0, \quad (8)$$

where $P_{L,K}(\cos \theta)$ are the associated Legendre polynomials and N_{LK} are normalization constants. The $y_{LK}(\theta, \phi)$ s form a complete orthogonal basis set of well known properties [22]. The above representation is limited to surfaces (described as stellar surfaces) that contain an interior point that can be connected to every point on the surface by a straight line without intersecting the surface. We represent a general (stellar or non-stellar) surface \vec{S} that is topologically equivalent to the sphere parametrically by expanding its individual Cartesian coordinates using spherical harmonics series,

$$\vec{S}(\theta, \phi) = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} X(\theta, \phi) \\ Y(\theta, \phi) \\ Z(\theta, \phi) \end{bmatrix} \quad (9)$$

where $X(\theta, \phi)$, $Y(\theta, \phi)$ and $Z(\theta, \phi)$ are individually expanded using Equation 6, giving three sets of expansion coefficients $(C_{LK}^X, C_{LK}^Y, C_{LK}^Z)$ which completely define the shape. The numerical implementation of SH calculations necessitates choosing a series truncation (L_{max}). Also, given a set of data points, the C_{LK} s are calculated according to,

$$C_{LK}^X = \int_0^{2\pi} \int_0^\pi X(\theta, \phi) y_{LK}(\theta, \phi) \sin \theta d\theta d\phi \quad (10)$$

with similar equations for $Y(\theta, \phi)$ and $Z(\theta, \phi)$. It should be noted that the SHP is particularly economical and is not confined to any particular symmetry.

2.4 Balancing internal and external energies: the L-curve method

The regularization parameter λ in Equation 1, determines the balance between our prior information about the observed object (usually a smoothing function), expressed through the internal energy, on the one hand, and fitting of the contour to the image data (also called the residual function), our external energy, on the other. An objective procedure for choosing λ is the L-curve method [11]. It is constructed by plotting the logarithm of the residual function vs. the logarithm of the smoothing function for a sufficiently large range of λ values. The optimal regularization parameter value corresponds to the point of maximum curvature on this usually L-shaped curve, which we determine graphically (Fig.3b). If an L-curve calculation does not give the proper vertical part of the “L”, then the series is artificially (implicitly) over-truncated and the series truncation must be relaxed to include higher order coefficients. On the other hand, if the L-curve is missing the horizontal part then the theoretical prediction (encoded in E_{internal}) coincides with the actual contour, and one could conclude that the shape is accounted for completely by the theory.

Note that we also use the L-curve method in this work in the context of finding the optimal spherical mapping for the determination of a starting guess (Fig.2f) (see Section 3.2).

3 Computational Methods

3.1 Method Overview

We begin by generating a starting guess of our surface contour, where the 3D fluorescence image is thresholded using Laplacian-of-Gaussian zero-crossing edge detection [23]. From the resulting point-cloud a surface triangulation is generated that must not contain small handles or holes. This is followed by uniformly mapping the surface to a unit sphere so that Equation 10 can be applied, to calculate three series expansions corresponding to the individual x , y and z coordinates. The coefficients of these expansions form the starting parametric approximation of the surface, which is then refined by iteratively minimizing the energy expression of Equation 1. We assume that changes to the coefficients introduced throughout the fitting are small enough so a modification of the initial spherical mapping is not necessary. The minimization is repeated for a sufficient range of λ values and an L-curve is constructed. Our final surface is the one fitted with the λ value that corresponds to the corner of the L-curve, which we determine graphically. The calculation of the internal energy necessitates evaluation of A , V , and H of the surface, for which we use the expressions in [24]. Our formula for calculating the external energy is given in the

Appendix. For the minimization of Equation 1 we used the downhill simplex (Simplex) algorithm [25]. Below we provide details regarding the spherical mapping.

3.2 The Spherical Mapping Step

Starting with the surface triangulation, each surface point (x, y, z) must be mapped onto a point (θ, ϕ) on the surface of the unit sphere, while maintaining connectivity and nearest neighbors, preserving relative triangle areas and minimizing triangle shear deformation. We demonstrate our procedure for accomplishing this on the spherical mapping of the triangulated surface of a letter E (Fig.2a).

The first step in the topological mapping is a rough mapping of surface points to the unit sphere conserving connectivity. We follow the method of [10] after modifying it for surface triangulations. In short, two poles (vertices) are chosen on the surface mesh. One is identified as the "North pole" (θ_N) and the second as the "South pole" (θ_S). For assigning a (latitude) θ value, a Laplace equation $\nabla^2\theta = 0$, with Dirichlet conditions $\theta_N = 0$ and $\theta_S = \pi$ is solved, (a stationary heat diffusion equation) (Fig.2b). To calculate (longitude) ϕ , a date line is introduced along which ϕ is incremented or decremented by 2π , and the cyclic Laplace equation is then solved (Fig.2c). Now each vertex has associated with it a unique (θ, ϕ) coordinate and can be placed on the unit sphere (Fig.2d).

For a proper final shape representation the vertices must be uniformly distributed on the unit sphere. As a starting point, we use a modified version of equation 6 in [12]. The problem is to minimize

$$\sum_i^m \left(\left(\frac{a_{p,i}}{4\pi} - \frac{a_{o,i}}{\sum_i a_{o,i}} \right)^2 + \gamma \beta_i^2 \right) \quad (11)$$

where $a_{p,i}$ is the geodesic area of triangle $i = 1, 2, \dots, m$, on the parametric sphere, $a_{o,i}$ is the area of triangle i on the original object, β_i is the measure of shear deformation of triangle i when mapping from object to parameter space, calculated from the local principal stretches $\lambda_{i,1}$ and $\lambda_{i,2}$ [26], $\beta_i = (\lambda_{i,1} - \lambda_{i,2})^2 / 2\lambda_{i,1}\lambda_{i,2}$, and γ is a scaling factor that controls the linear combination and is in essence a regularization parameter. The first term in Equation 11 drives the optimization to equate the relative triangle areas in both object and parameter spaces. The second term measures the extent of deformation of individual triangles relative to their undeformed state, which is the configuration in object space. The optimization of Equation 11 is performed for a series of γ values and an L-curve is constructed (Fig. 2f). The resulting mapping, when γ is chosen properly, is uniform (Fig.2e), and is used to calculate the coefficients according to Equation 10 (Fig.2g middle). Non-optimal choices of γ result in deformed surfaces that are unrealistic and unsuitable as starting shapes (see Figure 2g left and right).

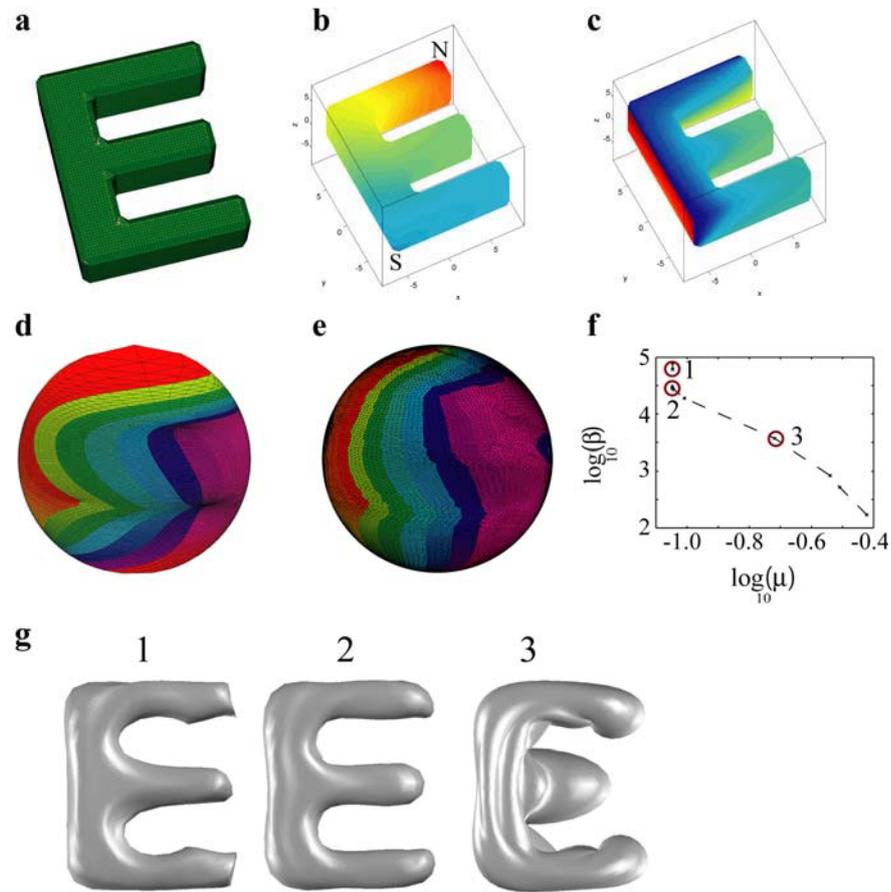


Fig.2. Spherical Mapping. Parameterization of the letter E by minimization of the balance between area dilation and shear deformation. **(a)** Surface triangulation of the capital letter E, **(b)** assignment of values for longitude (θ) and **(c)** latitude (ϕ) to the vertices by solving the stationary heat equation, which provides the initial mapping of the vertices onto the unit sphere, **(d)** θ, ϕ configuration in parameter space before performing the mapping optimization **(e)** θ, ϕ configuration in parameter space after performing the mapping optimization for the γ value corresponding to the corner of the L-curve in **(f)**, **(f)** L-curve for finding the optimal γ value (Equation 11) that balances area preservation μ (first term in Equation 11) and polygon deformation β (the second term), **(g)** shape models constructed by expanding the Cartesian coordinates in SH at an expansion truncation of $L_{\max} = 10$, after optimization of Equation 11 for three different values of γ (corresponding to marked positions on the L-curve in **(f)**), number 2 is the one with the best γ .

4 Results and Discussion

We demonstrate our proposed scheme for biophysics-guided 3D global parametric active contours on a synthetic 3D fluorescence microscopy data set.

4.1 Recovery of a known Shape from noisy 3D Pseudo-image Data

We constructed a noisy 3D synthetic intensity data set that mimics imaging a membrane bilayer-labeled discocytic surface under conditions typical for confocal fluorescence microscopy (Fig.3a). The discocyte surface was originally generated with $L_{\max} = 3$. This surface is of minimum (internal) energy at $\overline{\Delta a_o} = 0.00143$, for typical values of bending and area difference moduli (see Section 2.1) and under the constraints $A = 140 \mu\text{m}^2$ and $V = 100 \mu\text{m}^3$.

To test the consistency of our method, we applied it to the recovery of the discocyte shape from the noisy data set. We allowed spherical harmonic coefficients up to order $L_{\max} = 22$ as free parameters to be fitted, and repeated the optimization for decreasing values of λ ranging from 10^4 to 10^{-1} . At high λ the optimization finds coefficients that minimize the overlap of the contour with the noisy image, resulting in an unrealistic bumpy surface (Fig.3 b contour I). As λ decreases the surface gradually gets smoother (Fig.3 b contour II) until the corner of the L-curve is reached (Fig.3 b contour III).

When we performed these calculations with the correct (to us known beforehand) $\overline{\Delta a_o}$ value of 0.00143, all further points on the curve coincided with the (now) corner point shown and no L-shaped curve was obtained. This is expected because with the correct biophysical parameters the shape will converge to the discocyte, driven by the internal energy alone, independent of the image, which in this case happens to be the image of the same shape, so no further degradation of the image term occurs. However, in the general case $\overline{\Delta a_o}$ is not accurately known, so in order to test whether our method is robust against deviations of $\overline{\Delta a_o}$ from its true value, we repeated the calculation with $\overline{\Delta a_o} = 0.0005$, obtained the L-curve shown, and registered a clear corner that corresponds to the (smooth) discocyte. Further decrease in λ , led to the appearance of an elongated shape that minimized the internal energy under the given (guessed) $\overline{\Delta a_o}$.

5. Conclusion

In this work we extended the applicability of the method of biophysically-based active contours to image segmentation of membrane bilayer-bound objects into three space dimensions using the spherical harmonics parametric shape description. The

prior information about shape contour topology is incorporated implicitly in our choice of shape description. Our prior information about the mechanics of the membrane and its smoothness is explicitly formulated in an image-independent internal energy term that incorporates principal results from membrane biophysics theories and experimental membrane mechanics measurements. We also introduced a method for the step of initial spherical mapping. The regularization parameter needed for both the active contours and the spherical mapping optimizations was determined using the L-curve method. We demonstrated the applicability of our method using a realistic synthetic 3D image data set.

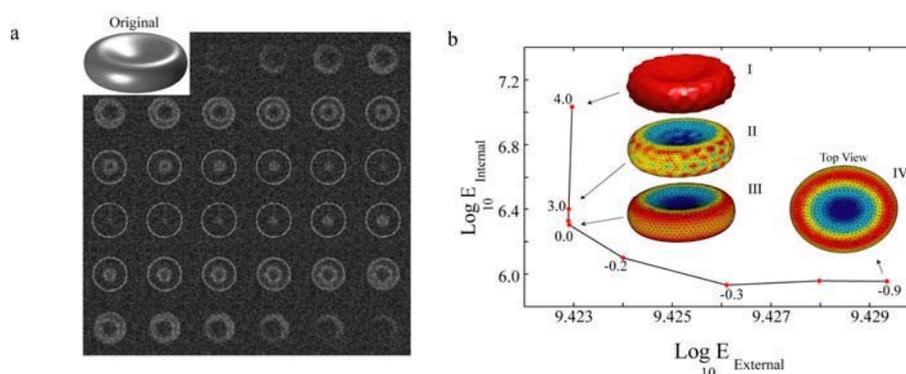


Fig.3. Recovery of a discocyte from noisy synthetic data. (a) Synthetic noisy image stack based on a slightly tilted discocyte surface ($L_{\max} = 3$) (shown in the inset), convolved with a theoretically calculated point spread function. (b) L-curve for a range of λ values (from 1×10^4 to 1×10^{-6}). The starting shape was obtained by spherical mapping with $L_{\max} = 22$, numbers next to the data points represent $\log_{10}(\lambda)$. To obtain this L-curve, $\Delta \bar{a}_o$ of 0.0005 was used instead of the true value 0.00143. At small λ values the discocytic shape changes to an elliptocyte (the default shape at this $\Delta \bar{a}_o$ value). At high λ the external energy dominates and because high expansion orders were allowed, the surface tries to fit the noise, and becomes irregular. The optimal surface is found at the corner of the L-curve.

Acknowledgements This work was funded by the Max-Planck Society. J. Pecreaux's work was also funded by the Human Frontier Science Program Organization.

References

1. Kass, M., A. Witkin, and D. Terzopoulos, *Snakes: active contour models*. International Journal of Computer Vision, 1988. **1**: p. 321-331.
2. Cootes, T., et al., *Active shape models - their training and application*. Computer vision and image understanding, 1995. **61**: p. 38-59.
3. Lipowsky, R., *The conformation of membranes*. Nature, 1991. **349**: p. 475-481.

4. Waugh, R. and R. Bauserman, *Physical measurements of bilayer-skeletal separation forces*. Ann Biomed Eng, 1995. **23**(3): p. 308-21.
5. Strey, H., M. Peterson, and E. Sackmann, *Measurement of erythrocyte membrane elasticity by flicker eigenmode decomposition*. Biophys J, 1995. **69**(2): p. 478-88.
6. Scheffer, L., et al., *Atomic force pulling: probing the local elasticity of the cell membrane*. European Biophysics Journal, 2001. **30**(2): p. 83-90.
7. Lim, H.W.G., M. Wortis, and R. Mukhopadhyay, *Stomatocyte-discocyte-echinocyte sequence of the human red blood cell: evidence for the bilayer-couple hypothesis from membrane mechanics*. Proc Natl Acad Sci USA, 2002. **99**(26): p. 16766-9.
8. Doebereiner, H.G., et al., *Mapping vesicle shapes into the phase diagram: A comparison of experiment and theory*. Phys. Rev. E, 1997. **55**(5): p. 4458-74.
9. Pécereaux, J., C. Zimmer, and J. Olivo-Marin. *Biophysical Active Contours for Cell Tracking I: Tension and Bending*. in ICIP 2006. 2006. Atlanta, USA.
10. Brechbühler, C., G. Gerig, and O. Kuebler, *Parametrization of closed surfaces for 3-D shape description*. Comput Vision Image Und, 1995. **61**(2): p. 154-170.
11. Hansen, P.C., *Rank-deficient and discrete ill-posed problems: numerical aspects of linear inversion*. 1997, Philadelphia: SIAM.
12. Quicken, M., et al. *Parameterization of Closed Surfaces for Parametric Surface Description*. in IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2000. 2000: IEEE Computer Society.
13. Shen, L. and F. Makedon, *Spherical mapping for processing of 3D closed surfaces*. Image and vision computing, 2006. **24**: p. 743-761.
14. Deuling, H.J. and W. Helfrich, *Red blood cell shapes as explained on the basis of curvature elasticity*. Biophys J, 1976. **16**(8): p. 861-8.
15. Sheetz, M.P. and S.J. Singer, *Biological membranes as bilayer couples*. Proc Natl Acad Sci USA, 1974. **71**: p. 4457-4461.
16. Seifert, U., K. Berndl, and R. Lipowsky, *Shape transformations of vesicles: Phase diagram for spontaneous- curvature and bilayer-coupling models*. Phys Rev A, 1991. **44**(2): p. 1182-1202.
17. Heinrich, V., S. Svetina, and B. Zeks, *Non-Axisymmetric Vesicle Shapes in a Generalized Bilayer-Couple Model and the Transition between Oblate and Prolate Axisymmetric Shapes*. Phys Rev E, 1993. **48**(4): p. 3112-3123.
18. Miao, L., et al., *Budding transitions of fluid-bilayer vesicles: The effect of area-difference elasticity*. Phys Rev E, 1994. **49**(6): p. 5389-5407.
19. Mukhopadhyay, R., H.W.G. Lim, and M. Wortis, *Echinocyte shapes: bending, stretching, and shear determine spicule shape and spacing*. Biophys J, 2002. **82**(4): p. 1756-72.
20. Chan, T.F. and L.A. Vese, *Active Contours without edges*. IEEE Transactions on image processing, 2001. **10**(2): p. 266-277.
21. Mumford, D. and J. Shah, *Optimal Approximations by Piecewise Smooth Functions and Associated Variational-Problems*. Communications on Pure and Applied Mathematics, 1989. **42**(5): p. 577-685.

22. Hobson, E.W., *The theory of spherical and ellipsoidal harmonics*. 1955, New York: Chelsea.
23. Marr, D. and E.C. Hildreth, *Theory of edge detection*. Proc R Soc Lond Ser B, 1980. **207**: p. 187-217.
24. Julicher, F., *Die Morphologie von Vesikeln*. 1993, University of Cologne. p. 131.
25. Nelder, J.A. and R. Mead, *A simplex method for function minimization*. Comput J, 1965. **7**: p. 308.
26. Evans, E.A. and R. Skalak, *Mechanics and Thermodynamics of Biomembranes*. 1980, Boca Raton, FL: CRC.

Appendix: Calculation of E_{external}

In analogy with electrostatics, one can write an “electric” field \vec{E}_{in} and \vec{E}_{out} corresponding to a distribution of charge $(I - c_{\text{in}})^2$ and $(I - c_{\text{out}})^2$. Using the Gauss theorem,

$$E_{\text{external}} = \lambda_{\text{in}} \int_{\text{surface}} \vec{E}_{\text{in}} \cdot d\vec{N} + \int_{\text{image}} (I - c_{\text{out}})^2 dV - \lambda_{\text{out}} \int_{\text{surface}} \vec{E}_{\text{out}} \cdot d\vec{N} \quad (\text{A-1})$$

where $d\vec{N} = \frac{\partial S}{\partial \theta} \times \frac{\partial S}{\partial \varphi}$ is the local surface normal.

This approach would necessitate a careful and precise solution of the Poisson equations for the potential from which \vec{E}_{in} and \vec{E}_{out} are derived. Such a treatment is beyond the scope of this paper, so for the sake of practicality we compute the value of the image at the contour positions by an integration of the functional derivative of the external energy,

$$\frac{\delta E_{\text{external}}}{\delta S} = \lambda_{\text{in}} \int_{\text{surface}} \text{div} \vec{E}_{\text{in}} d\vec{N} - \lambda_{\text{out}} \int_{\text{surface}} \text{div} \vec{E}_{\text{out}} d\vec{N} \quad (\text{A-2})$$

This only requires the computation of the surface normals using the image value on the surface only, which makes the computation more efficient.

Modelling Cerebral Cortical Folding

Guangqiang Geng^{1,2}, Leigh Johnston^{1,3}, Edwin Yan⁴, David Walker⁵ and Gary Egan^{1,6}

¹ Howard Florey Institute

² Graduate School of Biomedical Engineering, University of New South Wales

³ Dept. of Electrical & Electronic Engineering, University of Melbourne

⁴ National Trauma Research Institute, Alfred Hospital

⁵ Dept. of Physiology, Monash University

⁶ Centre for Neuroscience, University of Melbourne, Australia

Abstract. Understanding the biomechanical mechanisms by which the cerebral cortex folds is a fundamental problem in neuroscience. Current mathematical models of cortical folding do not include three dimensional geometry of developing brains extracted from experimental data. We present a biomechanical model which integrates 3D information extracted from MRI scans of fetal lamb brains at a series of developmental stages as an initial stage for the computation of cortical folding, and which utilises Diffusion Tensor Imaging (DTI) measurements of white matter fibre directions as a cue to the tension forces thought to regulate folding. We simulated the structures that result from different loadings and constraints, and compared the simulated with the actual geometry of the developing brain. This work is a proof of principle that finite element models and DTI can be combined to create a biologically meaningful model of the cortical folding process common to higher order mammals.

1 Introduction

The cerebral cortex covers most of the superficial part of the mammalian brain. Folding of the cerebral cortex in human and higher mammals such as the baboon and macaque monkey is related to higher intelligence, such that the greater the degree of cortical folding, the more intelligent the species [1]. During the initial stages of brain development in the human embryo, a smooth cortex forms by cell migration from subcortical structures. At the fifth or sixth month of gestation, the cortex begins to fold, forming the gyri (outward folds) and sulci (inward folds) from this stage until after birth [2]. The cortical folding increases the surface area of the cortex relative to the brain volume, thought to maintain a proportional increase of the cortical surface area during significant growth phases of the cortical volume [3].

Mathematical models have been proposed to investigate the biomechanisms of cortical folding, for improved understanding of the genesis of neurological disorders that result in abnormal cortical folding, such as Down's syndrome [4] and lissencephaly, in which the brain folds are less numerous and smaller than normal folds [3]. These modelling studies simulate cortical folding via mathematical

formulation of mechanisms including elasticity and plasticity of the cortex [5, 6], differential growth of cortical layers [3], limitation of the growth by the skull and white matter fibres [6], and modulation of tangential neuron migration by chemical activation and inhibition [7]. No model of the 3D geometry of the developing cerebral cortex, however, has yet been integrated with experimental data.

Current mathematical models initialise the shape of the cortex prior to folding with artificial geometries such as a closed ring [5], a flat sheet [3], or a 1D curve [6]. In addition, most models have no direct extension to 3D [5–7]. Furthermore, the resultant simulated geometries are either compared only visually with true cortical folding [7], or are quantitatively evaluated, but without comparison to a real cerebral cortex [5].

In order to understand the regulation of cortical folding by biomechanical factors including the 3D developing cortical geometry and white matter fibre tension forces, for which no experimental data are available from direct measurement, we propose a 3D biomechanical finite element model that integrates experimental data via Magnetic Resonance Imaging (MRI).

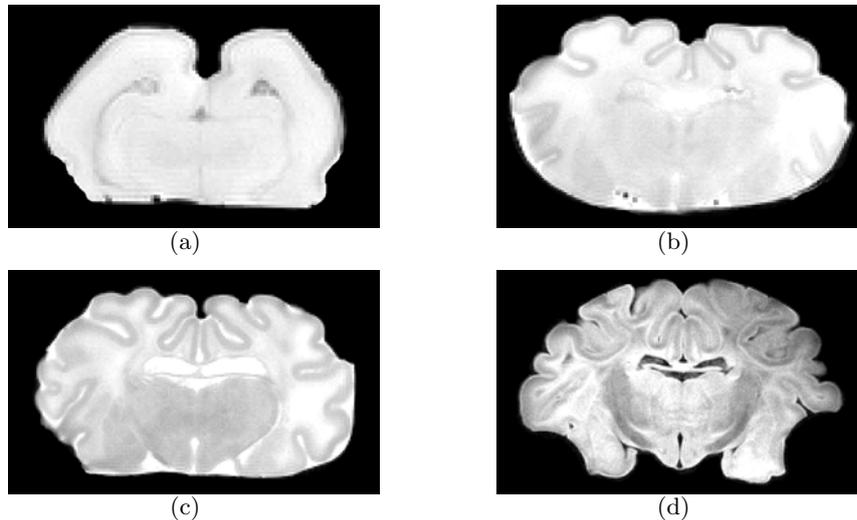


Fig. 1. Slices of T2-weighted MRI data for fetal sheep brains at a) 70 b) 90 c) 110 and d) 130 days gestation.

High-resolution MRI data are acquired for a set of fetal lamb brains at 70, 90, 110 and 130 days of gestation. As evident in Fig. 1, the surface of the 70 day brain is smooth with only a few shallow dimples. The greatest increase in the number of sulci occurs between 70 and 90 days gestation, as demonstrated in Fig. 2. For that reason, this study focuses on modelling the geometric deformation of a cortical region between 70 and 90 days gestation. The geometry of a smooth 70 day cortical region is extracted from the MRI data. The deformation of this initial shape is computed via a dynamic solid stress-strain model, and is subsequently compared with the true 90 day brain.

We integrate experimental MRI data by using white matter fibre anisotropy, as measured by Diffusion Tensor Imaging (DTI) [8], as a cue to tension applied between subcortical structures and the interior cortical surface. White matter fibre tension has been proposed by Van Essen to be a critical factor in the regulation of cortical folding [9].

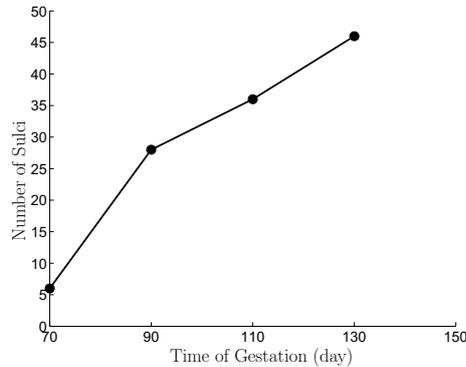


Fig. 2. Number of sulci in the fetal lamb brain as a function of gestational age.

2 Methods

2.1 Preparation of Fetal Lamb Brains

Fetal sheep brains are extracted at the relevant gestational time points. Each fetus is perfused transcardially with 500-1000ml of heparinized saline according to the size of the fetus, followed by the same volume of 4% Paraformaldehyde (PFA). The brain is subsequently removed from the fetus, and is placed in 4% PFA at least 48 hours before the MRI scan.

2.2 Acquisition of MRI Data

A Brüker Biospec 4.7T animal MRI scanner is used. T2-weighted and diffusion weighted imaging (DWI) data are acquired for each of the brains via a multi-slice multiecho sequence and a standard DTI sequence. Coronal slices of 1mm thickness with in-plane resolution of $0.258 \times 0.258 \text{ mm}^2$ are acquired perpendicular to the anterior-posterior commissure line. For the 70 day brain, the imaging matrix is $128 \times 128 \times 30$ (Fig. 3a), while the imaging matrix is $256 \times 256 \times 45$ (Fig. 3b) for the 90 day brain due to the increased brain size. TE = 50ms for both brains, TR = 4s and 10s for the 70 and 90 day brains, respectively. Twelve-direction DWI data are acquired following the T2-weighted scan with the same matrix size, spatial resolution, TE and TR, with a b -value of 1000 s/mm^2 for both brains.

2.3 Cortical Mesh Construction

To extract the 3D cortical geometry, a dorsal segment from each brain volume between the longitudinal cerebral fissure, which separates the left and right hemispheres, and the Sylvian sulcus of the left hemisphere, is selected, as shown in Fig. 3. The cortex is manually delineated across slices within the segment and saved as a binary image mask. The 2D cortical contour is extracted from each slice of the volume, and using COMSOL Multiphysics¹, a 3D volumetric geometry is produced through filling and lofting together the 2D geometries. COMSOL's meshing feature is used to generate meshes for the cortical geometries with Lagrange-Quadratic tetrahedral elements.

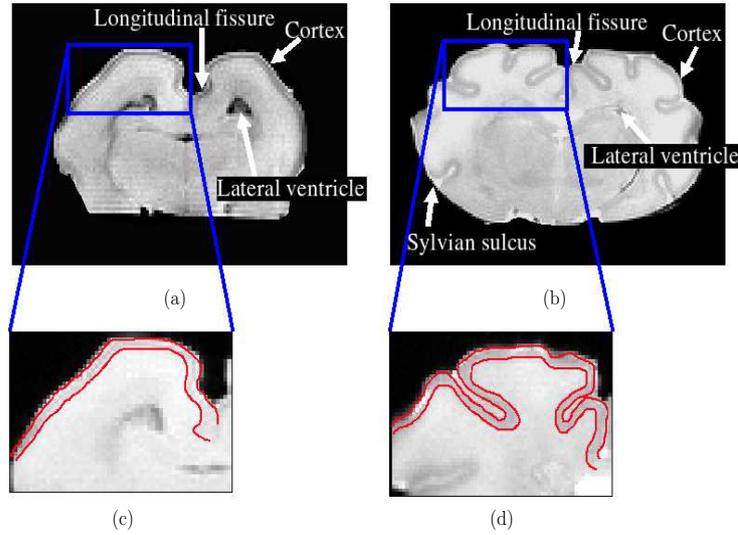


Fig. 3. Coronal slice of a) 70 day and b) 90 day brain, T2-weighted MRI. Manually delineated boundaries of cortical segment for c) 70 and d) 90 day brain.

2.4 Computation of Cortical Folding

Mathematical Model: Due to the dynamic and significant deformation involved in cortical folding, the process is formulated as a time-dependent large deformation. The Cauchy stress, σ , and Green-Lagrangian strain tensor, ϵ , are used as measurement of the stress and strain internal to the cortex [10]. The Cauchy stress is defined as the force per unit area in the deformed geometry, and an element of Green-Lagrangian strain tensor ϵ is as follows:

$$\epsilon_{ij} = \frac{1}{2} \left(\sum_{\alpha=1}^3 \frac{\partial x_{\alpha}}{\partial a_i} \frac{\partial x_{\alpha}}{\partial a_j} - \delta_{ij} \right) \quad (1)$$

¹ <http://www.comsol.com/>

where ϵ_{ij} is the element of strain tensor ϵ at row i and column j , a and x are coordinates for original and deformed positions respectively, and $\delta_{ij} = 1$, iff $i = j$.

The principle of virtual work evolves the geometry, and is expressed as:

$$\int_S \sigma_{ni} \delta u_i dS + \int_V B_i \delta u_i dV - \int_V \rho \ddot{u}_i \delta u_i dV = \int_V \sigma_{ji} \delta \epsilon_{ij} dV \quad (2)$$

where the four terms in (2) are the virtual work done by the surface stress force, the external body force, the inertial force and the internal force [11], respectively. σ_{ni} and B_i are surface stress force and external body force, respectively. ρ is mass density, and \ddot{u}_i is acceleration.

Although more realistic nonlinear constitutive relationships are to be tested in the future, the cortical tissue constitutive relationship is assumed to be linear and isotropic to simplify the problem. This follows the linear formulation of elastic material properties applied in neurosurgical simulation of human brain deformation [12].

Experimental measurements of the elastic properties of the fetal sheep brain are not available. We therefore use the relevant values for the human brain, for which the Young's modulus is 2.1×10^5 N/m², the Poisson ratio is 0.45, and the mass density of the cortical tissue is set to be 1.04×10^3 kg/m³, as in the human brain [13].

Loading and Constraints The cortical surface is categorized into three regions (Fig. 4) with different loadings and constraints for displacement, the exterior surface, the interior surface, and the anterior and posterior surfaces.



Fig. 4. Loading and constraints on surface regions. Blue: exterior surface, green: interior surface, red: anterior and posterior surfaces.

The exterior surface is the boundary between the cortex and cerebral-spinal fluid (CSF), onto which a pressure force is applied by the CSF with a spatially homogeneous distribution. The CSF pressure is set according to experimental measurement [14] to be 15mmHg, normal to the surface.

The interior surface is the boundary between the cortex and white matter fibres, with the latter applying tension forces to the cortex [9]. Based on observation of the principle eigenvector of the diffusion tensor reconstructed from the

DWI data for the 70 day brain (see Fig. 5), the force applied from white matter fibres onto the interior surface of the cortex is set to be along the normal direction of the local surface, in agreement with Hilgetag and Barbas’s observation in the rhesus brain [15].

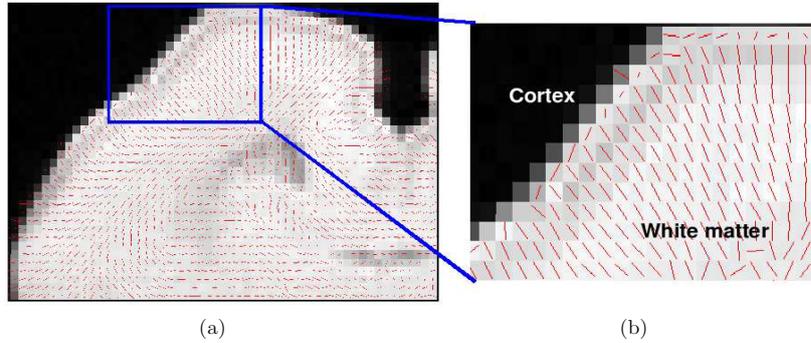


Fig. 5. (a) Principle diffusion tensor eigenvectors (red) overlaid on T2-weighted image. (b) Zoomed section displaying homogeneous distribution of vectors perpendicular to the boundary between cortex and white matter.

The anterior and posterior surfaces of the cortical section are the boundaries between different parts of the cortex. To limit the spatial domain of the problem within the cortical region extracted, no loading is applied to these two surfaces, and displacement normal to these two surface regions is inhibited. For the same reason, the ends of the cortical region are fixed to prevent displacement. A Rayleigh damping force is applied with a mass damping of 1s^{-1} , and a stiffness damping of 0.001s .

Solution of Equations The iterative Generalized Minimal Residual (GMRES) solver with incomplete lower-upper (LU) factorization preconditioner for non-symmetric matrix problems is selected from COMSOL Multiphysics’s modules to solve for the deformation of the geometry of the 70 day cortical region with 10^5 to 10^6 degrees of freedom.

3 Results

The 3D geometries of the 70 (Fig. 6) and 90 (Fig. 8c) day cortical regions are extracted from T2-weighted MRI data, as detailed in Sec. 2.3. The 70 day mesh, initially consisting of 2902 elements, is refined to 11199 tetrahedral elements (Fig. 6b).

Values for white matter fibre tension in developing brains are not available. We specify two loading conditions for simulation. Firstly, the magnitude of pressure on the interior surface is set to be 4000 N/m^2 , double the magnitude of the CSF pressure, and pushing normal to the surface in the central region (blue in

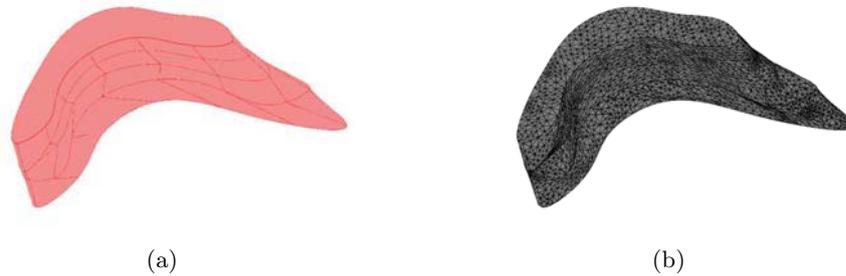


Fig. 6. Geometry of 70 day cortical region. (a) Volumetric rendering (b) Initial mesh.

Fig. 7a), while pulling the interior surface at the both lateral regions (green in Fig. 7a).

In the second scenario, a force with a pressure magnitude of 1000 N/m^2 is applied tangential to the interior surface at the central part of the cortical region (Fig. 7b), and the normal pressure is reduced to 3000 N/m^2 in order to prevent over compression of the cortex. This is to simulate the effects of tangential growth of the cortex, resulting in tangential tension at the superficial layer of the cortex.

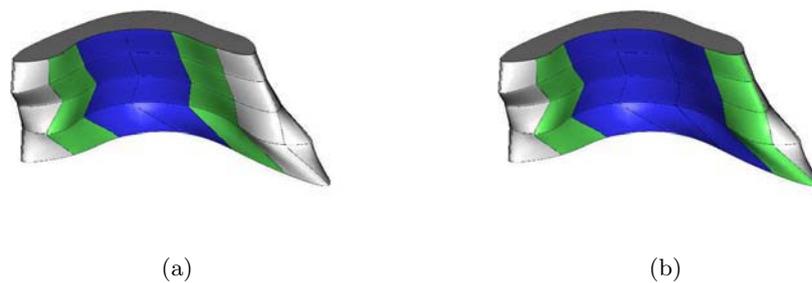


Fig. 7. Loadings on interior surface. Blue: Force applied onto cortical surface. Green: Force pulling away from cortical surface. (a) Loading without tangential force (b) Loading with tangential force.

Although simulation of the time scale for a real gestational period is planned for future work with reduced magnitudes of loading pressure, the simulation is stopped in order to prevent excessive deformation compared to the 90 day cortical region. The computation of the deformation of the 70 day cortical region takes 4-8 hours on a 2.4 GHz PC with 2.5Gb memory. The resultant geometries of the folded cortical region without and with tangential force are shown in Fig. 8a and Fig. 8b, with the true 90 day region in Fig. 8c.

The cortical thickness changes in the sulci and gyri, as summarised in Table 1. There is an increase in cortical thickness at the two lateral inward folds compared

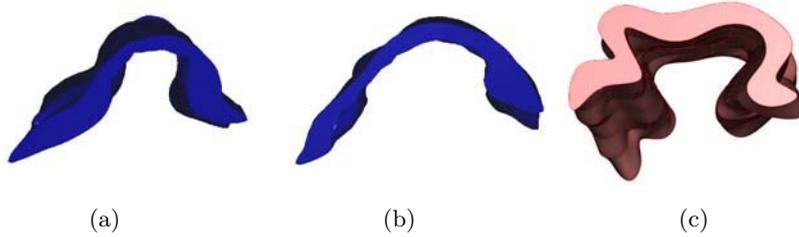


Fig. 8. (a) Simulated geometry without tangential force. (b) Simulated geometry with tangential force. (c) 90 day geometry extracted from T2-weighted MRI data.

	Gyri	Sulci			Ratio (gyri/sulci)
		left	right	mean	
No tangential force	1.01	1.29	1.63	1.46	0.69
With tangential force	1.00	1.15	1.63	1.39	0.72
90 day geometry	1.07	1.81	1.90	1.86	0.58

Table 1. Cortical thickness in gyri and sulci (mm)

to a homogeneous cortical thickness of 1.09 mm within the 70 day cortical region. This is due to the compression of the central part of the cortex by the CSF pressure from the exterior surface and the normal pushing force from the interior surface. At the two lateral inward folds, the normal force pulls the interior surface of the cortex away from its exterior surface.

The application of a tangential force on the interior surface at the central part of the cortical region results in more realistic geometry (Fig. 8b) with a broader gyral section as in the real 90 day cortical region. The cortical thickness in a scenario without tangential force is compared with that of a scenario with a tangential force. There is larger change in cortical thickness for the simulation with tangential force compared to the result without tangential force (Table 1) both in gyri and sulci. Hilgetag and Barbas [15] suggested this to be a result of the normal compression onto the deep layers of the cortex, induced by the tangential tension applied on the superficial layer of the cortex, verified in this simulation with tangential force. As is shown in Table 1, the cortical thickness in simulation results either with or without tangential force is less than that of the 90 day cortical geometry in both gyri and sulci. This is due to the limitation that cortical thickness growth is not simulated in this study.

4 Discussion and Conclusions

This study simulates the cortical folding of an initial 3D geometry directly extracted from MR volumetric data, for a fetal lamb brain at 70 days of gestation. This is immediately an improvement to existing theoretical 1D and 2D models of cortical folding that contain no experimental data. The physical properties and

displacement constraints for the brain tissue and CSF pressure are nominated up according to experimental human brain measurements. Our biologically meaningful mechanical model simulates the tension applied by white matter fibres, based on DTI data, to regulate cortical folding as proposed by Van Essen [9].

The 3D geometry resultant from the computation of cortical deformation is qualitatively similar to the geometry directly extracted from the MR volumetric data for the folded cortical region in the 90 day brain. The results of this simulation verify as a proof of principle that cortical folding can be simulated in 3D with an initial geometry directly extracted from a developing brain. The computed geometric change simulates the cortical folding both outward and inward. The simulated change of cortical thickness agrees with the observation and measurement on the real 90 day cortical region. Application of tangential tension at the interior surface of the cortical region results in broader outward folding as well as a further decrease in cortical thickness, which results in the simulated geometry being more similar to the actual cortical region. However, the increase of the thickness and tangential size of the cortex is not formulated in the current study, which leads to difference between the simulated and actual geometries.

We are currently working on a number of improvements to this preliminary investigation. Firstly, the nonlinear formulation of the constitutive properties of brain tissue, which are more realistic for soft tissues [10, 16, 17] and have been implemented in simulation of brain deformation for image registration required by neurosurgery [18] as well as for modelling the deformation of other soft tissues [19–21]. Secondly, the mechanical force applied on the interior surface of the cortex can be quantitatively integrated from DTI data as an improvement to the current qualitative formulation. In this approach, the orientations of the principle eigenvectors of the diffusion tensors will provide the direction of the white matter fibre tension, and the anisotropy of the diffusion will be proportional to the magnitude of the tension force. In addition, the full cortex will be considered, rather than the present section, and the brains of subsequent gestational ages will be integrated, for a complete picture of the folding pattern.

References

1. Buettner-Janusch, J.: Evolutionary and genetic biology of primates: Volume 1. Academic Press, New York (1963)
2. Ottoson, D.: Physiology of the nervous system. The Macmillan Press, London and Basingstoke (1983)
3. Richman, D.P., Stewart, R.M., Hutchinson, J.W., Caviness, J.V.S.: Developmental model of brain convolutional development. *Science* **189**(4196) (1975) 18–21
4. Venita, J.: Pathology in an infant with Down’s syndrome and Tuberous Sclerosis. *Pediatrics Neurology* **15** (1996) 57–59
5. Toro, R., Burnod, Y.: A morphogenetic model for the development of cortical convolutions. *Cerebral Cortex* **15** (2005) 1900–1913
6. Raghavan, R., Lawton, W., Ranjan, S., Viswannathan, R.: A continuum mechanics-based model for cortical growth. *Journal of Theoretical Biology* **187** (1997) 285–296

7. Cartwright, J.: Labyrinthine Turing pattern formation in the cerebral cortex. *Journal of Theoretical Biology* **217** (2002) 97–103
8. Basser, P.J., Mattiello, J., LeBihan, D.: MR diffusion tensor spectroscopy and imaging. *Biophysical Journal* **66** (1994) 259–267
9. Van Essen, D.C.: A tension-based theory of morphogenesis and compact wiring in the central nervous system. *Nature* **385** (1997) 313–318
10. Fung, Y.C.: *Biomechanics: Mechanical properties of living tissues*. 2 edn. Springer-Verlag, New York (1993)
11. Boresi, A.P., Chong, K.P.: *Elasticity in engineering mechanics*. Elsevier, New York (1987)
12. Warfield, S.K., Ferrant, M., Gallez, X., Nabavi, A., Jolesz, F.A., Kikinis, R.: Real-time biomechanical simulation of volumetric brain deformation for image guided neurosurgery. In: *SC 2000: High Performance Networking and Computing Conference; 2000 Nov 4–10; Dallas, USA.* (2000) 230:1–16
13. Gao, C., Tay, F.E.H., Nowinski, W.L.: Modelling of the human brain with detailed anatomy for numerical simulation of surgical interventions. *Journal of Physics: Conference Series* **34** (2006) 985–989
14. Neville, L., Egan, R.A.: Frequency and amplitude of elevation of cerebrospinal fluid resting pressure by the valsalva maneuver. *Canadian Journal of Ophthalmology* **40** (2005) 775–777
15. Hilgetag, L.C., Barbas, H.: Developmental mechanics of the primate cerebral cortex. *Anatomy and Embryology* **210** (2005) 411–417
16. Spencer, A.J.M.: *Continuum Mechanics*. Longman, New York (1980)
17. Malvern, L.E. In: *Introduction to the mechanics of a continuous medium*. Prentice-Hall (1969) 273
18. Hagemann, A., Rohr, K.: Biomechanical modeling of the human head for physically based, nonrigid image registration. *IEEE Transaction on medical imaging* **18**(10) (1999) 875–884
19. Pathmanathan, P., Gavaghan, D., Whiteley, J., Brady, S.M., Nash, M., Nielsen, P., Rajagopal, V.: Predicting tumour location by simulating large deformations of the breast using a 3D finite element model and nonlinear elasticity. In: *MICCAI.* (2004) pp. 217–224
20. Nash, M.P., Hunter, P.J.: Computational mechanics of the heart. *Journal of Elasticity* **61** (2000) 113–141
21. Fernandez, J.W., Hunter, P.J.: An anatomically based patient-specific finite element model of patella articulation: towards a diagnostic tool. *Biomechanical Model in Mechanobiology* **4** (2005) 20–38

Suite of finite element algorithms for accurate computation of soft tissue deformation for surgical simulation

Grand Roman Joldes, Adam Wittek, Karol Miller

Intelligent Systems for Medicine Lab., The University of Western Australia,
35 Stirling Highway, Crawley, WA 6009, AUSTRALIA
{grandj, adwit, kmiller}@mech.uwa.edu.au

Abstract. Real time computation of soft tissue deformation is important for the usage of augmented reality devices and for providing haptic feedback during operation or surgeon training. This requires algorithms that are both fast and accurate and can handle material and geometric nonlinearities. A combination of such algorithms is presented in this paper, starting with the finite element formulation and integration scheme used and addressing common problems such as hourglass control and locking. The computation example presented proves that by using these algorithms real time computations becomes possible without having to sacrifice the accuracy of the results.

Keywords: non-locking tetrahedron, hourglass control, real time computations, Total Lagrangian formulation, explicit time integration

1 Introduction

Systems using augmented reality for image guided surgery are important tools that can help surgeons to improve the accuracy and limit the adverse effects of surgery. The existing imaging technology, such as MRI, provides good quality pre-operative images that can be used in such a system. These images can be analyzed and registered on the real organs so that the surgeon can visualize the targeted area while the procedure is progressing.

Another area where fast computational algorithms are required is the Computer Integrated Surgery (CIS) systems that must provide haptic feedback to the surgeon. Various haptic interfaces for medical simulation are especially useful for training of minimally invasive procedures (laparoscopy/interventional radiology) and remote surgery using teleoperators. These systems must compute the interaction force between the robotic tool and the tissue and provide it to the surgeon at frequencies in the area of 500 Hz [1].

Biomechanical models are used for solving the haptic feedback problems using the finite element method, but many such models are simplified in order to decrease the computational effort, e.g. they consider only infinitesimal deformations and/or linear material laws. These simplifications have great influence on the accuracy of the ob-

tained results in a finite element analysis, inducing significant errors [2-4]. Therefore material law and geometric non-linearity must be considered when a solution method is chosen.

There are three methods of reducing the computation time: improvement of the algorithms, usage of faster hardware or usage of parallel computing. We will concentrate on the first method, as the usage of faster hardware is limited by the existing technology and the usage of parallel computing leads to more complex and more expensive hardware and software systems.

The paper is organized as follows: the proposed algorithms are presented in section 2, a computational example demonstrates the efficiency of these algorithms in section 3 and the conclusions are presented in section 4.

2 Finite element algorithms

When designing a finite element solution method there are many aspects that must be considered, such as the formulation used (Total or Updated Lagrangian), time integration scheme and the type of elements used for constructing the mesh. We will discuss these aspects in the following section.

2.1 Total Lagrangian Explicit integration

Various spatial discretisation schemes are possible while using the finite element method [5]. The algorithms used by the great majority of commercial finite element programs use the Updated Lagrangian formulation, where all variables are referred to the current (i.e. from the end of the previous time step) configuration of the system (Ansys [6], ABAQUS [7], ADINA [8], LS Dyna [9], etc.). The advantage of this approach is the simplicity of incremental strain description. The disadvantage is that all derivatives with respect to spatial coordinates must be recomputed in each time step, because the reference configuration is changing. The reason for this choice is historical – at the time of solver development the memory was expensive and caused more problems than the actual speed of computations. The first key idea in the finite element algorithms development was to use the Total Lagrangian formulation of finite element method, where all variables are referred to the original configuration of the system. Second-Piola Kirchoff stress and Green strain are used. The decisive advantage of this formulation is that all derivatives with respect to spatial coordinates are calculated with respect to original configuration and therefore can be precomputed. The proposed stress and strain measures are appropriate for handling geometric nonlinearities (finite deformations).

The usage of Total Lagrangian explicit integration for simulating physically realistic deformations was also proposed in [10]. A method for decreasing the computation time when using non-linear elasticity was presented in [11], but it only works for tetrahedral meshes and special elastic material laws.

Because biological tissues behavior can be described in general using hyper-elastic or hyper-visco-elastic models [12], the usage of the Total Lagrangian formulation also

leads to a simplification of material law implementation as these material models can be easily described using the deformation gradient.

The integration of equilibrium equations in time domain can be done using either implicit or explicit methods [13-15]. The most commonly used implicit integration methods, such as the Newmarks' constant acceleration method, are unconditionally stable. This implies that their time step is limited only by the convergence/accuracy considerations. However, the implicit methods require solution of set of non-linear algebraic equations at each time step. Furthermore, iterations need to be performed for each time step of implicit integration to control the error and prevent divergence. Therefore, the number of numerical operations per each time step can be three orders of magnitude larger than for explicit integration [13].

On the other hand, in explicit methods, such as a central difference method, treatment of non-linearities is very straightforward and no iterations are required. By using a lumped (diagonalised) mass matrix, the equations of motion can be decoupled and no system of equations must be solved. Computations are done at the element level eliminating the need for assembling the stiffness matrix of the entire model. Thus, computational cost of each time step and internal memory requirements are substantially smaller for explicit than for implicit integration. There is no need for iterations anywhere in the algorithm. These features make explicit integration suitable for real time applications.

However, the explicit methods are only conditionally stable. Normally a severe restriction on the time step size has to be included in order to receive satisfactory simulation results. Stiffness of soft tissue is very low [16-18]: e.g. stiffness of brain is about eight orders of magnitude lower than that of steel. Since the maximum time step allowed for stability is (roughly speaking) inversely proportional to the square root of Young's modulus divided by the mass density, it is possible to conduct simulations of brain deformation with much longer time steps than in typical dynamic simulations in engineering. A detailed description of the Total Lagrange Explicit Dynamics [TLED] algorithm is presented in [19].

2.2 Elements used in the mesh

Because of the computation time requirement, the mesh must be constructed using low order elements that are not computationally intensive, such as the linear tetrahedron or the linear under-integrated hexahedron. The standard formulation of the tetrahedral element exhibits volumetric locking [14], especially in case of soft tissues such as the brain, that are modeled as almost incompressible materials [17]. Therefore hexahedral elements are preferred.

Many algorithms are now available for fast and accurate automatic mesh generation using tetrahedral elements, but not for automatic hexahedral mesh generation [20-22]. This is one reason why many authors proposed the usage of tetrahedral meshes for their models [23-26]. In order to automate the simulation process, mixed meshes (having both hexahedral and tetrahedral elements) with predominantly hexahedral elements are the most convenient.

The under-integrated hexahedral elements require the usage of an hourglass control algorithm in order to eliminate the zero energy modes which arise from the one-point

integration [27]. Special algorithms for handling hourglass control for the hexahedral elements and to reduce locking of the tetrahedral elements must be implemented.

2.3 Hourglass control

One of the most popular and powerful hourglass control algorithms, that is currently available in many commercial software finite element packages, is the one proposed by Flanagan and Belytschko in [27]. This method is applicable for hexahedral elements with arbitrary geometry undergoing large deformations.

Starting from that algorithm proposed by Flanagan and Belytschko we could show that the Total Lagrangian formulation is also recommended from the point of view of efficient hourglass control implementation, as many quantities involved can be pre-computed. We have shown in [28] that the hourglass control forces for each element can be computed (in matrix form) as:

$${}^t_0\mathbf{F}^{Hg} = k_0 \mathbf{Y}_0 \mathbf{Y}_0^T {}^t_0\mathbf{u} \quad (1)$$

where k is a constant that depends on the element geometry and material properties, \mathbf{Y} is the matrix of hourglass shape vectors and \mathbf{u} is the matrix of current displacements. The notation from [14] is used, where the left superscript represents the current time and the left subscript represents the time of the reference configuration, which is 0 for Total Lagrangian. In (1) all quantities except \mathbf{u} are constant and can be pre-computed, making the hourglass control mechanism very efficient.

2.4 Improved tetrahedral elements

There are a number of improved (non-locking) linear tetrahedral elements already proposed by different authors [29-31]. These formulations are either much more computationally intensive than the standard formulation or the volumetric locking control mechanism depends on material properties (e.g. bulk modulus), making harder the interfacing of different materials. Our volumetric lock control mechanism is computationally inexpensive and depends solely on kinematic variables.

As defined in [30], the nodal Jacobian is the ratio between the current and initial nodal volumes. The nodal volume is computed as a sum of fractions of the surrounding element volumes.

Using the nodal Jacobians, an average Jacobian can be computed for each element. Because the element Jacobian is equal to the determinant of the element deformation gradient, we define a modified element deformation gradient that has the same isochoric part as the normal deformation gradient, but the volumetric part is modified so that its determinant (and therefore the volumetric deformation) is equal to the average element Jacobian.

The computation of the nodal forces (or stiffness matrix) can now be done in the usual manner, but using the modified deformation gradient instead of the normal deformation gradient for defining the strains. This way there is no need for computing the isochoric and deviatoric components of the internal forces separately, and the existing material law can be used [32].

A similar approach is used in ABAQUS [7] for defining the selectively reduced integration of the volumetric term for the fully integrated first-order isoparametric hexahedral elements, in order to prevent locking in incompressible or nearly incompressible cases. The difference is that while for the selectively reduced integrated hexahedra the Jacobians are averaged over an element, in our case the Jacobians are averaged between elements sharing the same node.

3 Computational example

A brain indentation simulation was conducted in order to assess the computational efficiency and the accuracy of the presented algorithms. The same experiment was conducted using the commercial software package Abaqus. The mesh we used has 2428 nodes and 2059 elements (2023 under-integrated hexahedron and 36 improved tetrahedral elements in the indentation area - see Fig. 1).

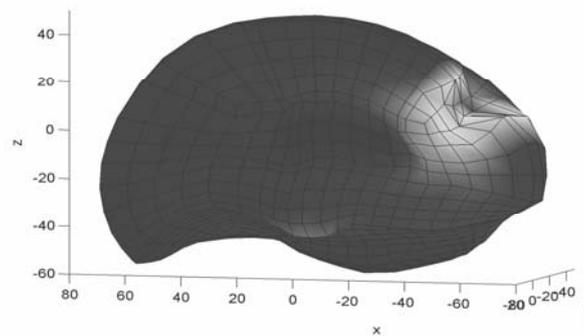


Fig. 1. Brain indentation experiment – the mixed mesh is deformed by displacing 4 nodes

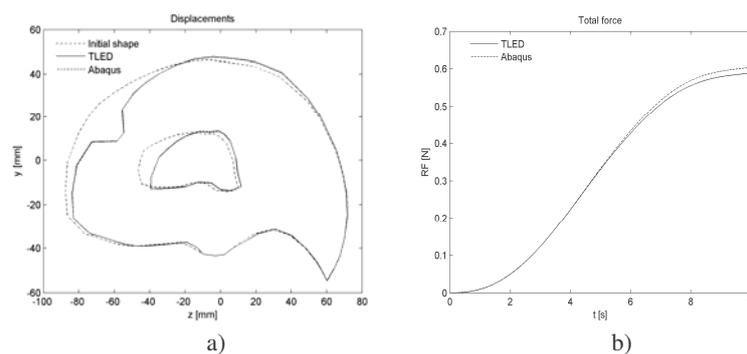


Fig. 2. Computed displacements (a) and reaction forces (b) using Abaqus implicit and our algorithms

The indentation was simulated by displacing 4 nodes in the direction normal to the brain surface by 20 mm using a smooth loading curve. An almost incompressible non-linear neo-Hookean material was used for the brain tissue (mass density of 1000 kg/m^3 , Young's modulus in undeformed state equal to 3000 Pa and Poisson's ratio 0.49) and an almost incompressible linear material for the ventricle (mass density of 1000 kg/m^3 , Young's modulus in undeformed state equal to 10 Pa and Poisson's ratio 0.49). The same constraints as in [33] were used and brain symmetry was assumed.

Computations were performed on a standard 3.2 GHz Pentium 4 desktop using Windows XP operating system. The default options were used for the Abaqus implicit solver. Fully integrated mixed formulation elements were used in Abaqus, which are the golden standard elements in case of almost incompressible materials simulations.

The simulation consisted of 2000 time steps and took 3.8s using TLED, giving a force feedback frequency of 526 Hz. The Abaqus implicit simulation performed 100 time steps in more than 5 minutes. There is a very good concordance between the results obtained using our software and the results from the Abaqus simulation, in case of both displacements and reaction forces (see Fig. 2) – the displacements almost overlap and the maximum relative error in reaction forces is 2.5%.

4 Conclusions

In this paper we presented a suite of finite element algorithms that can be used for accurate and fast computation of soft tissue deformation for surgical simulation. The basic concept behind these algorithms is the usage of the Total Lagrangian formulation for solving finite element problems. The presented algorithms cover issues related to time integration, hourglassing and volumetric locking. We use fully nonlinear formulation, accounting for large deformations, rigid body motions and material nonlinearities.

Explicit time integration is the preferred method for performing real time simulations. The treatment of non-linearities is straightforward, without the need of any iteration. Even if the method is only conditionally stable, the material properties of biological tissue make possible the usage of much larger time steps compared with other engineering applications.

A new tetrahedral element formulation based on the average nodal Jacobian was developed. This formulation uses only kinematic variables for controlling the volumetric locking, and therefore the usage of different materials and the implementation in an existing finite element code can be made without difficulties.

A very efficient hourglass control implementation is proposed for the under-integrated hexahedral element. Having only one integration point, this element is very inexpensive from the computational point of view, being a perfect candidate for real time surgical simulations. Using this type of element and the improved tetrahedral element for simulations using mixed meshes is a step towards complete automated patient specific surgical simulation.

The simulation example confirms the speed of the presented algorithms. We could compute reaction forces at frequencies higher than 500 Hz for a mesh having more than 2000 hexahedral elements on a simple PC workstation. The accuracy of our re-

sults was demonstrated by comparing them with the results of a similar simulation done using much more complex elements in the commercial finite element software Abaqus.

Acknowledgements: The first author was an IPRS scholar in Australia during the completion of this research. The financial support of the Australian Research Council (Grant No. DP0343112, DP0664534 and LX0560460) is gratefully acknowledged.

References

1. DiMaio, S.P. and Salcudean, S.E.: Interactive Simulation of Needle Insertion Models. *IEEE Trans. on Biomedical Engineering*. (2005) 52(7)
2. Carey, G.F.: A Unified Approach to Three Finite Element Theories for Geometric Nonlinearity. *Journal of Computer Methods in Applied Mechanics and Engineering* (1974) 4(1): p. 69-79
3. Martin, H.C. and Carey, G.F.: *Introduction to Finite Element Analysis: Theory and Application*. McGraw-Hill Book Co., New York (1973)
4. Oden, J.T. and Carey, G.F.: *Finite Elements: Special Problems in Solid Mechanics*. Prentice-Hall, (1983)
5. Belytschko, T.: An Overview of Semidiscretization and Time Integration Procedures, in *Computational Methods for Transient Analysis*, Editor: T. Belytschko and T.J.R. Hughes (1983) North-Holland: Amsterdam. p. 1-66
6. <http://www.ansys.com/services/documentation>, ANSYS documentation, Accessed: 18/10/2005
7. ABAQUS: ABAQUS Theory Manual, Version 5.8. Hibbitt, Karlsson & Sorensen, Inc., (1998)
8. www.adina.com, ADINA home page, Accessed: 22/11/2005
9. Hallquist, J.O.: *LS-DYNA Theory Manual*. Livermore Software Technology Corporation, Livermore, California 94551 (2005)
10. Zhuang, Y. and Canny, J.: Real-time Simulation of Physically Realistic Global Deformation. *IEEE Vis'99* (1999) San Francisco, California
11. Picinbono, G., Delingette, H., and Ayache, N.: Non-linear anisotropic elasticity for real-time surgery simulation. *Graphical Models* (2003) 65: p. 305-321
12. Fung, Y.C.: *Biomechanics. Mechanical Properties of Living Tissues*. Second ed. Springer-Verlag, New York (1993)
13. Belytschko, T.: A survey of numerical methods and computer programs for dynamic structural analysis. *Nuclear Engineering and Design* (1976) 37: p. 23-34
14. Bathe, K.-J.: *Finite Element Procedures*. Prentice-Hall, New Jersey (1996)
15. Crisfield, M.A.: Non-linear dynamics, in *Non-linear Finite Element Analysis of Solids and Structures*, (1998) John Wiley & Sons: Chichester. p. 447-489
16. Miller, K. and Chinzei, K.: Constitutive modelling of brain tissue; Experiment and Theory. *Journal of Biomechanics* (1997) 30(11/12): p. 1115-1121
17. Miller, K.: *Biomechanics of Brain for Computer Integrated Surgery*. Publishing House of Warsaw University of Technology, Warsaw (2002)

18. Miller, K. and Chinzei, K.: Mechanical properties of brain tissue in tension. *Journal of Biomechanics* (2002) 35: p. 483-490
19. Miller, K., et al.: Total Lagrangian Explicit Dynamics Finite Element Algorithm for Computing Soft Tissue Deformation. *Communications in Numerical Methods in Engineering* (2007) 23: p. 121-134
20. Owen, S.J.: A Survey of Unstructured Mesh Generation Technology. 7th International Meshing Roundtable (1998) Dearborn, Michigan, USA
21. Viceconti, M. and Taddei, F.: Automatic generation of finite element meshes from computed tomography data. *Critical Reviews in Biomedical Engineering* (2003) 31(1): p. 27-72
22. Owen, S.J.: Hex-dominant mesh generation using 3D constrained triangulation. *Computer-Aided Design* (2001) 33: p. 211-220
23. Ferrant, M., et al.: Deformable Modeling for Characterizing Biomedical Shape Changes. *Discrete Geometry for Computer Imagery: 9th International Conference* (2000) Uppsala, Sweden: Springer-Verlag GmbH
24. Ferrant, M., et al.: Serial registration of intraoperative MR images of the brain. *Medical Image Analysis* (2002) 6(4): p. 337-359
25. Clatz, O., et al.: Patient Specific Biomechanical Model of the Brain: Application to Parkinson's disease procedure. *International Symposium on Surgery Simulation and Soft Tissue Modeling (IS4TM'03)* (2003) Juan-les-Pins, France: Springer-Verlag
26. Warfield, S.K., et al.: Real-time registration of volumetric brain MRI by biomechanical simulation of deformation during image guided surgery. *Computing and Visualization in Science* (2002) 5: p. 3-11
27. Flanagan, D.P. and Belytschko, T.: A uniform strain hexahedron and quadrilateral with orthogonal hourglass control. *International Journal for Numerical Methods in Engineering* (1981) 17: p. 679-706
28. Joldes, G.R., Wittek, A., and Miller, K.: Efficient Hourglass Control Implementation for the Uniform Strain Hexahedron Using the Total Lagrangian Formulation. *Communications in Numerical Methods in Engineering* (2007) Submitted, under review
29. Bonet, J. and Burton, A.J.: A simple averaged nodal pressure tetrahedral element for incompressible and nearly incompressible dynamic explicit applications. *Communications in Numerical Methods in Engineering* (1998) 14: p. 437-449
30. Bonet, J., Marriott, H., and Hassan, O.: An averaged nodal deformation gradient linear tetrahedral element for large strain explicit dynamic applications. *Communications in Numerical Methods in Engineering* (2001) 17: p. 551-561
31. Zienkiewicz, O.C., et al.: Triangles and Tetrahedra in Explicit Dynamic Codes for Solids. *International Journal for Numerical Methods in Engineering* (1998) 43: p. 565-583
32. Joldes, G.R., Wittek, A., and Miller, K.: Improved Linear Tetrahedral Element for Surgical Simulation. *Computational Biomechanics for Medicine - MICCAI 2006 Workshop Proceedings* (2006) Copenhagen, Denmark

33. Wittek, A., et al.: Computing reaction forces on surgical tools for robotic neurosurgery and surgical simulation. Australasian Conference on Robotics and Automation ACRA (2004) Canberra, Australia

Finite Element Simulation of the Beating Heart for Image-Guided Robotic Cardiac Surgery

Philip Pratt¹, Fernando Bello¹, Eddie Edwards¹, and Daniel Rueckert²

¹ Department of Biosurgery and Surgical Technology

² Visual Image Processing, Department of Computing
Imperial College of Science, Technology and Medicine
London, United Kingdom
p.pratt@imperial.ac.uk

Abstract. An interactive finite element simulation of the beating heart is described in which the intrinsic motion is implied from preoperative 4D tomographic scan data. The equations of motion are reversed, such that given changes in node displacements over time, the node forces that produce those changes are recovered. Subsequently, these forces are resolved from the global coordinate system into systems local to each mesh element. Therefore, at each simulation time step, the collection of node forces can be expressed as simple weighted sums of current node positions. Importantly, this facilitates the combination of extrinsic forces such as those due to tool-tissue interactions, changes in the relative direction in which gravity acts, and effects due to insufflation of the thoracic cavity and left lung deflation. Through the development of an image-guided coronary artery bypass application of the *da Vinci* tele-manipulator system, the method has been applied initially to volumetric images of a pneumatically-operated beating heart phantom.

1 Introduction

During the past decade, technical innovations have led to considerable advances in cardiac surgery. One such advance is the introduction of minimally invasive techniques. In particular, robotic tele-manipulator systems have made totally endoscopic coronary artery bypass (TECAB) a reality. But reductions in incision sizes and post-operative recovery times have seen corresponding increases in technical difficulty and durations of the procedures. This is partly because the surgeon cannot touch and palpate tissues directly, thereby making it more difficult to locate key anatomical structures beneath the tissue surface, and partly because of the limited field of vision afforded by the endoscopic cameras. Image guidance, using tomographic scan data acquired preoperatively, is one approach which addresses these limitations [1, 2]. A key requirement for image guidance in TECAB is the availability of 2D-3D registration techniques that can accurately process complex motion patterns and deformations. In addition to the intrinsic cardiac and respiratory motion, one must take into account tool-tissue interactions, effects due to left lung deflation, insufflation of the thoracic cavity, and the patient's position on the operating table relative to that adopted for preoperative scans.

This paper introduces a finite element model where the sequence of node forces required to produce intrinsic motion is recovered. These forces are resolved from the global coordinate system to be functions of their local geometry. In this way, their application is consistent with large-scale deformations and rotations and, more importantly, other extrinsic forces can be combined. The technique could be said to bring time-varying 3D scan data to life, such that it can be interacted with and not only observed.

Several relevant ideas have been put forward recently. In particular, this work builds on the image-guidance approach of Wang [3] and Wang *et al.* [4], who simulate tool-tissue interactions between a phantom heart and *da Vinci* endowrist instrument, integrating 3D geometry reconstructed from preoperative CT/MRI scans with a non-linear finite element model. The heart is nominally static, and has no intrinsic beating motion. Nakao [5] describes a model of the pulsating aorta, driven by cyclic changes in aortic blood pressure, which can be palpated by virtual tools in the context of a cardiac surgery simulation. Szalpa *et al.* [6] have developed an image guidance system for minimally-invasive cardiac surgery, whereby endoscopic images of a beating heart phantom are overlaid with CT images of the same. To reconstruct arbitrary phases of the cardiac cycle, non-linear free-form deformations are used to warp a high-quality image of the diastolic phase. Similarly, Wierzbicki *et al.* [7] create a dynamic epicardial surface mesh from a sequence of CT images. Chandrashekara *et al.* [8] have developed a non-rigid registration technique, based on Catmull-Clark subdivision lattices, to track the motion of the left ventricle in tagged MR images. Various finite element simulations have been described in which deformations are known *a priori*. Specifically, Kruggel and Tittgemeyer [9] use an inverse finite element model to derive a force field given an observed deformation of the brain. Kauer [10] also uses an inverse model to calibrate the properties of a visco-elastic material given experimental pressure data and resulting tissue deformations.

2 Methods

The tetrahedral finite element mesh comprises N nodes, and therefore has a maximum $3N$ degrees of freedom. In general, zero or more of these freedoms will be restrained. Following the notation in [11], restrained freedoms are specified by the node freedom array r_{jk} , where the coordinate axes are labelled $j = 0, 1, 2$ and the nodes are numbered $k = 0, 1, \dots, N-1$. Non-negative entries in the array label the D degrees of freedom in the mesh. The initial mesh geometry is specified by the global coordinate array g^{jk} . The equations of motion for a linear elastic solid are expressed in terms of the D displacements from the initial mesh configuration, i.e. $\mathbf{u} = [u^0 \ u^1 \ \dots \ u^{D-1}]^\top$. Thus, they can be written as

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{C}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f} \quad (1)$$

where \mathbf{M} is the mass matrix, \mathbf{C} is the viscous damping matrix, \mathbf{K} is the stiffness matrix, \mathbf{f} is the time-varying vector of applied forces, and \mathbf{u} , $\dot{\mathbf{u}}$ and $\ddot{\mathbf{u}}$ are the displacement, velocity and acceleration vectors, respectively. Rayleigh damping is assumed, i.e. $\mathbf{C} = \alpha_m \mathbf{M} + \alpha_k \mathbf{K}$.

2.1 Average Acceleration Integration

If Δt denotes the duration of each simulation time step, then using the Newmark family of integration methods [12], the discretised equations of motion take the following form. One can employ the implicit average acceleration method by setting $\beta = 1/4$ and $\gamma = 1/2$.

$$\begin{aligned}
\mathbf{a}_i &= \frac{1}{\beta\Delta t^2}\mathbf{u}_i + \frac{1}{\beta\Delta t}\dot{\mathbf{u}}_i + \left(\frac{1}{2\beta} - 1\right)\ddot{\mathbf{u}}_i \\
\mathbf{b}_i &= \frac{\gamma}{\beta\Delta t}\mathbf{u}_i - \left(1 - \frac{\gamma}{\beta}\right)\dot{\mathbf{u}}_i - \left(1 - \frac{\gamma}{2\beta}\right)\Delta t\ddot{\mathbf{u}}_i \\
\mathbf{u}_{i+1} &= \left[\mathbf{K} + \frac{\gamma}{\beta\Delta t}\mathbf{C} + \frac{1}{\beta\Delta t^2}\mathbf{M}\right]^{-1} (\mathbf{f}_i + \mathbf{M}\mathbf{a}_i + \mathbf{C}\mathbf{b}_i) \\
\dot{\mathbf{u}}_{i+1} &= \frac{\gamma}{\beta\Delta t}\mathbf{u}_{i+1} - \mathbf{b}_i \\
\ddot{\mathbf{u}}_{i+1} &= \frac{1}{\beta\Delta t^2}\mathbf{u}_{i+1} - \mathbf{a}_i
\end{aligned} \tag{2}$$

Given initial displacements, velocities and accelerations, the equations are iterated in the order listed above at each time step. The linear system is solved using Cholesky decomposition and back-substitution. The positions p_i^{jk} of each node k at any particular time step i are given by the following expression.

$$p_i^{jk} = \begin{cases} g^{jk} + u_i^{r_{jk}} & \text{if } r_{jk} \geq 0 \\ g^{jk} & \text{otherwise} \end{cases} \tag{3}$$

2.2 Recovering Forces from Displacements

The motion is assumed to be cyclic, spanning a total of T time steps. Successor and predecessor functions of the time index i are defined as follows.

$$next(i) = \begin{cases} i+1 & \text{if } i < T-1 \\ 0 & \text{otherwise} \end{cases} \quad prev(i) = \begin{cases} i-1 & \text{if } i > 0 \\ T-1 & \text{otherwise} \end{cases} \tag{4}$$

Using these definitions, two-sided finite difference approximations to the first and second-order derivatives of the displacement vector yield the following expressions for velocity and acceleration.

$$\dot{\mathbf{u}}_i = \frac{1}{2\Delta t} (\mathbf{u}_{next(i)} - \mathbf{u}_{prev(i)}) \quad \ddot{\mathbf{u}}_i = \frac{1}{\Delta t^2} (\mathbf{u}_{next(i)} - 2\mathbf{u}_i + \mathbf{u}_{prev(i)}) \tag{5}$$

These are used to compute the initial conditions $\dot{\mathbf{u}}_0$ and $\ddot{\mathbf{u}}_0$ from the known displacements. By inverting the displacement update step in (2) as follows, one can compute the discretised applied forces from known changes in displacement. Then by construction, if one were to solve the equations of motion and apply these forces at the appropriate times, one would recover the original cyclic motion exactly and indefinitely.

$$\mathbf{f}_i = \left[\mathbf{K} + \frac{\gamma}{\beta\Delta t}\mathbf{C} + \frac{1}{\beta\Delta t^2}\mathbf{M} \right] \mathbf{u}_{i+1} - \mathbf{M}\mathbf{a}_i - \mathbf{C}\mathbf{b}_i \tag{6}$$

2.3 Forces as Functions of Local Geometry

In order to combine intrinsic and extrinsic forces, the former must be expressed not in the global coordinate system, but for each node in terms of its local surrounding geometry. With the introduction of extrinsic forces, the geometry may deviate from the original cyclic motion through potentially large-scale deformations and rotations. By resolving intrinsic forces locally, they are made to act in the appropriate direction in conjunction with any extrinsically induced motion.

The intrinsic force acting on a particular node in the mesh is assumed to originate from the finite elements which contain that node. Indeed, an approximation is made whereby the force receives an equal contribution from each such element. For a given element at each point in time, the edge vectors from the node in question to the other three nodes in that element define a local basis in terms of which that element's fraction of the node force can be expressed. This amounts to equating the force to a weighted sum of those edge vectors and solving for the weights. Subsequently, the weights are further accumulated over the node's other parent elements, ultimately building a set of weights that links all the intrinsic forces to the geometry of the entire mesh.

Labelled with coordinate indices j , components h_i^{jk} of the node forces are mapped onto those applicable to each degree of freedom $\mathbf{f}_i = [f_i^0 \ f_i^1 \ \dots \ f_i^{D-1}]^\top$ by the node freedom array.

$$h_i^{jk} = \begin{cases} f_i^{r_{jk}} & \text{if } r_{jk} \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

Figure 1 (left) depicts at time step i a typical node k with its recovered force, and the five surrounding elements e_0, \dots, e_4 to which it belongs. In general, a node will be common to M_k elements. Figure 1 (middle) illustrates the first element e_0 , and the three edge vectors b_i^{j0} , b_i^{j1} and b_i^{j2} which, with the node itself, define the geometry of the element at that instant. In order to express the required fraction of the force in terms of local mesh geometry, its components in the global coordinate system are equated to weighted combinations of these edge vectors as follows.

$$\begin{bmatrix} b_i^{00} & b_i^{01} & b_i^{02} \\ b_i^{10} & b_i^{11} & b_i^{12} \\ b_i^{20} & b_i^{21} & b_i^{22} \end{bmatrix} \begin{bmatrix} w_i^0 \\ w_i^1 \\ w_i^2 \end{bmatrix} = \frac{1}{M_k} \begin{bmatrix} h_i^{j0} \\ h_i^{j1} \\ h_i^{j2} \end{bmatrix} \quad (8)$$

The weights are determined by direct inversion of the left-hand-side matrix. By accumulating contributions from each element containing the node, writing edge vectors in terms of node positions, incorporating the node freedom array, and considering all nodes, the total intrinsic force for each degree of freedom l can be expressed as

$$f_i^l = \sum_j \sum_k v_i^{jkl} p_i^{jk} \quad (9)$$

which lends itself particularly well to efficient implementation. Thus, in the absence of extrinsic forces and together with the finite element model, the sequence

of aggregate weights v_i^{jkl} completely determines the cyclic motion of the mesh. At each point in time, intrinsic forces are determined by local geometry only. The connectivity of the mesh dictates that many of the weights will be zero.

In order to ensure that the magnitudes of the intrinsic forces f_i^l , expressed as functions of position, do not give rise to simulation instability, the forces are normalised at each time step to have the same magnitude as those implied from the original motion where no extrinsic forces are present.

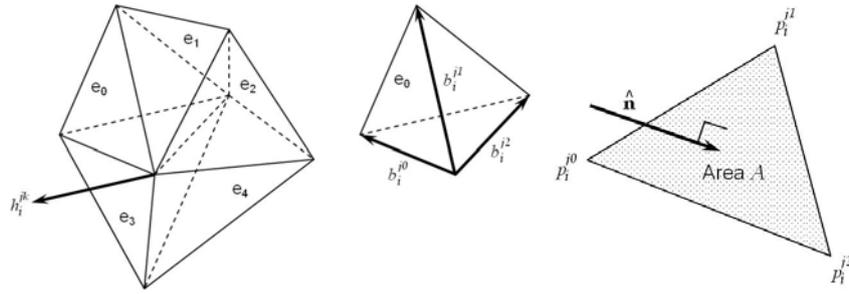


Fig. 1. Force resolution using local geometry and a typical external face

2.4 Tool-Tissue Interactions

Tool-tissue interactions are combined in the simulation by means of a simple instrument model, in which the end effector is taken to be a line segment. Barycentric coordinates returned by the collision detection algorithm upon detection of a face intersection are used to distribute an additional applied force appropriately over its three nodes. The magnitude of the force is proportional to the depth of segment penetration.

2.5 Insufflation

The mesh comprises a set of external triangular faces upon which pressurised insufflation gas can exert force. Figure 1 (right) shows one such face at a certain instant in time i , comprising nodes p_i^{j0} , p_i^{j1} and p_i^{j2} . It is assumed that the force due to insufflation pressure acts in a direction normal to the face, with a magnitude proportional to its area. The force is distributed uniformly over each of the face's three nodes. Viscous and frictional forces acting in other directions are assumed to be negligible. The face area A and unit normal $\hat{\mathbf{n}}$ are simple functions of the node positions. At each simulation time step, insufflation forces are accumulated over all external faces in the mesh to give the total additional force applied to each node.

2.6 Gravity

Let \mathbf{g} be the constant vector of node gravitational loads computed by integration of the element shape functions over all elements [11] during simulation initialisation. To account for changes in orientation relative to the position adopted during preoperative scans, the gravitational loads are subtracted from the original recovered intrinsic force vector (6) at each time step, to give gravity-adjusted intrinsic forces \mathbf{f}'_i , as follows.

$$\mathbf{f}'_i = \left[\mathbf{K} + \frac{\gamma}{\beta \Delta t} \mathbf{C} + \frac{1}{\beta \Delta t^2} \mathbf{M} \right] \mathbf{u}_{i+1} - \mathbf{M} \mathbf{a}_i - \mathbf{C} \mathbf{b}_i - \mathbf{g} \quad (10)$$

Trivially, the original motion is recovered if the gravitational loads are applied subsequently, unchanged, as extrinsic forces. However, it is now possible to rotate the finite element mesh and extrinsic gravitational loads, in an equal and opposite fashion, in order to simulate a change in patient orientation. This way, gravity will always take effect in the correct direction.

3 Results

The force recovery and resolution techniques were tested with data taken from scans of a beating heart phantom, using a simple linear elastic tissue model. The Chamberlain Group CABG phantom, illustrated in figure 2 (left), was scanned at 54 bpm with a Philips 64-slice CT scanner, producing 10 uniformly-spaced phases. The first of these was manually segmented and converted into a tetrahedral mesh of 709 elements and 747 degrees of freedom using the SimBio-Vgrid [13] mesh generator. Figure 2 (right) shows the interaction between the resulting mesh and a virtual tool. The Image Registration Toolkit [14, 15] was then used to create a sequence of 3D tensor product cubic B-spline deformations, mapping the initial mesh onto each phase in turn.

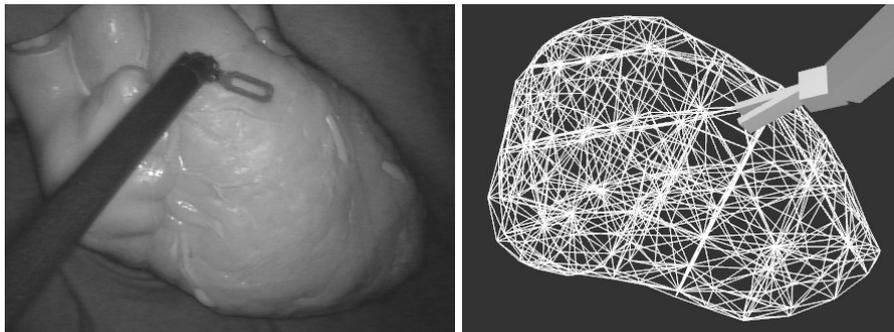


Fig. 2. Beating heart phantom and tetrahedral FEM mesh with virtual tool

To ensure stability of the FEM simulation by making the second time derivatives continuous, it was necessary to fit 1D cubic B-splines over time through each node. The basis functions were arranged in such a way that smoothness is maintained across the coincident start and end of the cardiac cycle. Six knots were used in this instance, implying one fewer basis functions to cover the entire cycle. Figure 3 (left) shows the passage of a typical node in the phantom mesh before and after cyclic smoothing, and Figure 3 (right) shows at each phase the maximum and mean errors resulting from the smoothing process. Their magnitude can be seen to be perfectly acceptable for image-guidance applications.

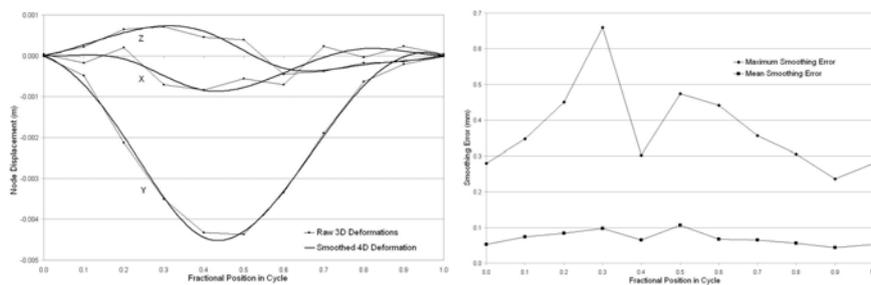


Fig. 3. Cyclic deformation smoothing and error measurements

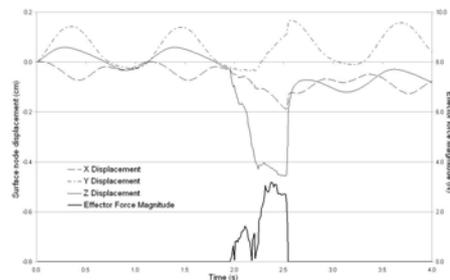


Fig. 4. Combined tool-tissue interactions

3.1 Simulation parameters

The following constants were chosen to approximate the material properties of the beating heart phantom: Young's modulus $E = 7.1E + 04$ Pa; Poisson's ratio $\nu = 0.285$; material density $\rho = 30.0$ kg/m³; Rayleigh mass damping coefficient $\alpha_m = 3.755E + 05$ and stiffness damping coefficient $\alpha_k = 0$; and acceleration due to gravity $g = 9.80665$ m/s². The equations of motion were integrated using a time step $\Delta t = 0.002314815$ seconds. The simulation uses an efficient sparse matrix implementation which makes real time interaction possible.

3.2 Tool-Tissue Interactions

The proprietary software written to test force recovery and resolution also simulates one or more virtual *da Vinci* endowrist instruments. Constrained by virtual trocar ports, each instrument has six degrees of freedom, and is controlled by a SensAble PHANTOM Omni haptic device. Figure 4 shows the displacements of a typical surface node close to the point of instrument contact, and the magnitude of the applied force. It is evident that the cyclic motion of the beating heart is combined with the inward motion induced by the instrument. Once the instrument is moved away from the surface, the mesh can be seen to relax into its original cyclic motion. The relative position of the node, and indeed the entire mesh, changes since there are no restrained degrees of freedom. The haptic device applies force feedback to the human operator and indeed, one can feel the pulsatile motion of the heart when contact is made with its surface. As such, the applied force magnitude exhibits an irregular increase and decrease itself in response to variations in the operator's application of the instrument.

3.3 Insufflation and Gravitational Effects

Figure 5 (left) shows the effect of an increase in insufflation gauge pressure. This results in an overall shrinking of the mesh, illustrated by a reduction in the distance between a typical node and the mesh centre. Observing the motion of another surface mesh node, figure 5 (right) shows how the intrinsic motion changes as the direction of gravity is rotated about the Z axis, increasing linearly from zero to some exaggerated target angle. Displacements in the Z direction remain unchanged whilst the other directions adapt slowly to the simulated change in patient position. Together with a detailed finite element model of the whole thorax including diaphragm and lungs, this is an important step towards accurate simulation of left lung deflation and its effect on heart motion.

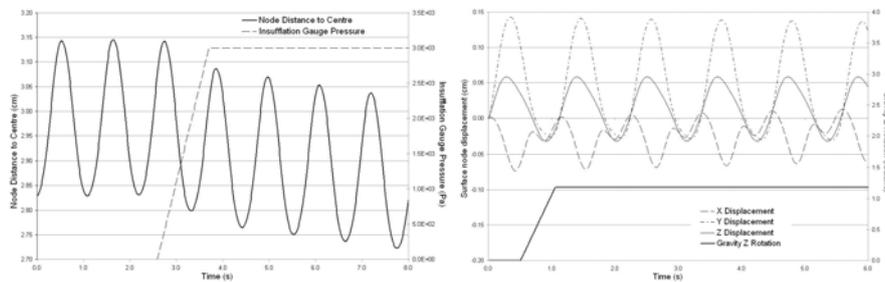


Fig. 5. Simulated insufflation and effects due to gravity direction change

4 Conclusion

This paper describes a novel technique for constructing finite element simulations with cyclical intrinsic motion recovered from 4D tomographic scan data, whereby extrinsic forces can be combined to produce realistic composite behaviour. In the context of an augmented reality application of the *da Vinci* robotic tele-manipulator system for TECAB, it offers several means by which the accuracy of overlay registration can be improved. Tool-tissue interactions, insufflation and effects due to gravity and change in patient orientation are combined with the beating heart. The technique is clearly also applicable to the field of surgical simulation and others where, for example, respiratory motion and patient inclination must be combined to produce accurate organ deformations, such as the treatment of liver tumours using robot-assisted RF ablation. Overall accuracy and validity of the technique will be determined by comparing simulation results to spatial data recovered from stereo endoscopic video sequences and recordings of effector positions.

It is also planned to extend the simulation to include non-linear finite element models. Clearly, the linear elastic model defined in (1) is not appropriate if the technique were applied to real patient data, but it suffices to illustrate the central ideas of force recovery and resolution in the heart phantom case. When non-linearities are present, the discretised equations of motion (2) are no longer applicable. Instead, an iterative scheme, such as the Newton-Raphson method for non-linear systems of equations, can be used to solve for the forces required to produce known displacements. Typically, the forward integration will also involve an iterative process [16], ensuring that dynamic equilibrium is restored at each simulation time step. At each such iteration, the internal forces and modified stiffness are re-calculated. Further refinements to the simulation will be made by introducing unequal weights and anisotropy during the local force resolution process, and by allowing weight interpolation over time, such that the frequency of cyclic motion can be modified.

Acknowledgements

The Image Registration Toolkit was used under licence from IXICO Ltd.. The authors would like to thank staff at St. Mary's and the Royal Brompton hospitals.

References

1. Chiu, A.M., Dey, D., Drangova, M., Boyd, W.D., Peters, T.M.: 3-D Image Guidance for Minimally Invasive Robotic Coronary Artery Bypass. *The Heart Surgery Forum* **3**(3) (2000) 224–231
2. Falk, V., Mourgues, F., Adhami, L., Jacobs, S., Thiele, H., Nitzsche, S., Mohr, F.W., Coste-Manière, E.: Cardio Navigation: Planning, Simulation, and Augmented Reality in Robotic Assisted Endoscopic Bypass Grafting. *Annals of Thoracic Surgery* **79**(6) (2005) 2040–2047

3. Wang, D.A.: Video-Based Augmented Reality For The da Vinci Tele-Manipulator System. PhD thesis, Department of Biosurgery and Surgical Technology, Imperial College of Science, Technology and Medicine, University of London (2005)
4. Wang, D.A., Faraci, A., Bello, F., Darzi, A.: Simulating Tele-Manipulator Controlled Tool-Tissue Interactions Using a Nonlinear FEM Deformable Model. In Westwood, J.D., et al., eds.: *Medicine Meets Virtual Reality 14*, IOS Press (2006) 565–567
5. Nakao, M.: Cardiac Surgery Simulation with Active Interaction and Adaptive Physics-Based Modeling. PhD thesis, Department of Social Informatics, Graduate School of Informatics, Kyoto University (2003)
6. Szpala, S., Guiraudon, G., Peters, T.: Cardiac Endoscopy Enhanced by Dynamic Organ Modeling for Minimally-Invasive Surgery Guidance. In Ellis, R.E., Peters, T.M., eds.: *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2003*, Springer (2003) 499–506
7. Wierzbicki, M., Drangova, M., Guiraudon, G., Peters, T.: Validation of dynamic heart models obtained using non-linear registration for virtual reality training, planning, and guidance of minimally invasive cardiac surgeries. *Medical Image Analysis* **8**(3) (2004) 387–401
8. Chandrashekhara, R., Razavi, R., Mohiaddin, R., Rueckert, D.: Nonrigid image registration with subdivision lattices: Application to cardiac MR image analysis. Submitted to *Medical Image Analysis* (2006)
9. Kruggel, F., Tittgemeyer, M.: Inverse Biomechanical Models of the Brain. In Kamm, R., Schmid-Schonbein, G., Athesian, G., Hefzy, M., eds.: *Proceedings of the 2001 Bioengineering Conference*, The American Society of Mechanical Engineers, New York (2001) 837–838
10. Kauer, M.: Inverse Finite Element Characterization of Soft Tissues with Aspiration Experiments. PhD thesis, Swiss Federal Institute of Technology, Zurich (2001)
11. Smith, I.M., Griffiths, D.V.: *Programming the finite element method*. 4th edn. John Wiley and Sons Ltd. (2004)
12. Hughes, T.J.R.: *The finite element method: linear static and dynamic finite element analysis*. Dover Publications, Inc. (2000)
13. Berti, G.: Image-based unstructured 3D mesh generation for medical applications. In Neittaanmäki, P., Rossi, T., Majava, K., Pironneau, O., eds.: *European Congress on Computational Methods in Applied Sciences and Engineering - ECCOMAS 2004*, University of Jyväskylä, Department of Mathematical Information Technology (2004)
14. Rueckert, D., Sonoda, L.I., Hayes, C., Hill, D.L.G., Leach, M.O., Hawkes, D.J.: Non-rigid registration using free-form deformations: Application to breast MR images. *IEEE Transactions on Medical Imaging* **18**(8) (1999) 712–721
15. Schnabel, J.A., Rueckert, D., Quist, M., Blackall, J.M., Castellano-Smith, A.D., Hartkens, T., Penney, G.P., Hall, W.A., Liu, H., Truwit, C.L., Gerritsen, F.A., Hill, D.L.G., Hawkes, D.J.: A generic framework for non-rigid registration based on non-uniform multi-level free-form deformations. In Niessen, W.J., Viergever, M.A., eds.: *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2001*, Springer (2001) 573–581
16. Reddy, J.N. In: *An Introduction to Nonlinear Finite Element Analysis*. Oxford University Press (2004) 304–306

Purely Evidence Based Multiscale Cardiac Tracking Using Optic Flow

Hans van Assen¹, Luc Florack^{1,*}, Avan Suinesiaputra², Jos Westenberg², and Bart ter Haar Romeny¹

¹ Biomedical Image Analysis, Dept. Biomedical Engineering, Technical University Eindhoven TU/e, Den Dolech 2, 5600 MB Eindhoven, Netherlands

h.c.v.assen@tue.nl <http://bmia.bmt.tue.nl>

² Div. Image Proc., Dept. Radiology, Leiden Univ. Med. Center, Leiden, Netherlands

Abstract. Heart disease can negatively influence cardiac pump function. To assess cardiac tissue function, a method based on classical optical flow theory applied in the spectral domain is presented. Assumption of pixel intensity conservation is replaced by assumption of spatial phase conservation. Simultaneous application to two independent observations of the same optical flow field removes the necessity of additional constraints (i.e. flow field smoothness, normal flow) to solve the optical flow constraint equation (OFCE). Using the 1st order Taylor expansion of the OFCE, our system yields not only pixel displacements, but also the 1st order differential structure of the displacements (i.e. strains), which otherwise should be calculated as a post-processing step.

Operation at pixel level obviates the need for interpolation of tag lines or sparse flow field representation. Experiments show coherent flow fields of a human cardiac systole. Comparison with velocity encoded MRI shows a good resemblance.

Key words: optical flow, tracking, heart, tagging, strain, HARP

1 Introduction

Many cardiac diseases have a large influence on the pump function of the cardiac left ventricle (LV). Valvular aorta stenosis, e.g., impedes blood flow from the left ventricle through the aortic valve into the aorta, thus serving as an obstruction. Furthermore, aorta stenosis influences systolic torsion and diastolic apical untwisting [1].

Measurements of muscle function are often used as prognostic indicator of success or failure of therapy. SV is one of the well-known indicators used in clinical practice for both diagnostic and prognostic purposes. Ejection fraction (EF), which is SV divided by end-diastolic volume, is the most often used functional parameter in both experimental and clinical studies. Due to a high afterload in aortic stenosis, EF may be reduced even while tissue function may be relatively

* The Netherlands Organisation for Scientific Research (NWO) is gratefully acknowledged for financial support.

normal [2]. This shows that EF as an indirect measure of tissue function may be misleading. Therefore, it would be better to measure tissue function itself.

A method is required that assesses cardiac tissue function both globally and locally and provides new means and criteria for assessment of mechanical dysfunction and to monitor response to treatment [3]. We propose to estimate cardiac function from tagged MR [4, 5] spatiotemporal image sequences by classification of the LV contraction pattern. Towards such an approach, we present a purely evidence based multiscale method for tissue function assessment from multiple tagged image sequences using neither interpolation of tags [6] or flow field [7], nor smoothness assumptions. Our approach follows the original rationale of Horn and Schunck [8], and extended to a multiscale framework by Florack et al. [9], Niessen et al. [10, 11] and Suinesiaputra et al. [12], and of spectral filtering introduced by Osman et al. [13]. As opposed to previous methods, however, we do not invoke any smoothness assumption or other hypothesis for resolving the ‘‘aperture problem’’ (intrinsic flow indeterminacy), but instead base ourselves entirely on data evidence.

In Section 2, we present general background information. In Section 3, we introduce optical flow and the way it is used in our proposed method. In Section 3.2, we explain our method, and in Section 4 we show some experimental results. Finally, in Section 5 we discuss results and future directions.

2 Background

2.1 Magnetic Resonance Tagging

In the late 1980s, Zerhouni et al. introduced a method (‘‘tagging’’) for visualising intramyocardial motion by MR imaging [14], later refined to (3D) (C)SPAMM [4, 5, 15]. The HARP technique, which employs tagging combined with spectral filtering in k-space [13, 16], overcomes tag fading by directly measuring *phase* information (Fig. 1(b)) of the MR signal (Fig. 1(a)). In 1999, another *phase*-based method for imaging myocardial function, called DENSE was introduced [17]. The principles behind HARP and DENSE, however, are the same [18]. A third well-known imaging protocol for dynamic analysis is Phase Contrast MRI (PCMRI) [19]. This technique directly measures velocity, which has to be integrated over time and spatially differentiated to obtain strains, which is far from trivial due to a limited temporal resolution. Moreover, PCMRI is subject to velocity errors, which propagate during displacement and strain analysis. For a review of motion analysis MRI protocols, cf. [18].

Patterns created with (C)SPAMM are inherent in the tissue, allowing analysis of the local dynamic behaviour of and strains in the LV, while imaging of the cardiac surfaces focuses on wall thickening. Strain analysis is superior to wall thickening analysis to detect wall motion abnormalities [20], since surface-based approaches are unable to display disturbances in, e.g., twisting during contraction and relaxation, which have been identified as early signs in pathologies [21].



Fig. 1. Left: vertically lined tagged image. Middle: Harmonic phase (HARP) image. Right: Sine of the HARP image.

2.2 Tag Tracking

Tracking with Optical Flow Tracking tags with optical flow methods has been presented before. Niessen et al. [10, 11] used a canine heart from MR cine images and Suinesiaputra et al. [12] tracked human hearts using grid tagging. Both applied the multiscale generalisation [9] of the OFCE by Horn and Schunck [8]. This method yields one equation and two unknowns in the zeroth-order approximation and four equations with eight unknowns in 1st-order approximation (Sec 3.1). To solve for all unknowns an additional constraint was required. Suinesiaputra et al. formalized the fact that flow components tangential to isosurfaces (“tangential flow”) cannot be retrieved from data evidence in a “normal flow constraint”. Constraining the flow to underlying image structure, the *calculated* optical flow does not necessarily correspond to the *true* optical flow. Dougherty et al. [22] also applied optical flow theory to analyze cardiac motion. They estimate global and local motion in a coarse-to-fine model-based technique. This technique encompasses a Laplacian filter to compensate for intensity and contrast loss in myocardial tags which violate the basic assumption of constant point brightness in optical flow techniques. This technique was compared with a semi-automatic method based on tracking individual tag intersection points [23]. The tag intersection points were found by manually thresholding the normalized cross-correlation result of the tagged MR image with a template of an idealized tag intersection, and served as input to an interactive active contour model. This technique yields the optical flow field at a sparse set of points in the image sequence, necessitating an interpolation step to obtain a dense field. Prince and McVeigh [24] developed an optical flow based method which requires extensive prior knowledge of the relaxation times T_1 , T_2 and the proton density D_0 of the myocardium. The adaptation of this technique by Gupta and Prince [25] still requires knowledge of T_1 over the image domain.

Tracking with Harmonic Phase (HARP) A different tracking method based on filtering in the Fourier domain (or *k-space*) is the Harmonic Phase (HARP) technique [13]. HARP involves spectral bandpass filtering and subsequently tracking phases in the angle image of the inverse Fourier transformed filter result. The HARP method serves as a basis for our optical flow based tracking technique, and is therefore further explained in Section 3.

3 Methods

3.1 Optical Flow

The classical approach to optical flow extraction [8] departs from an optical flow constraint equation (OFCE) that is intrinsically underdetermined, since flow components tangential to iso-surfaces (“tangential flow”) cannot be retrieved from data evidence. This indeterminacy is commonly referred to as the “aperture problem”. ‘Normal flow’, obtained by imposing the additional constraint that tangential flow vanishes identically, is useful in certain applications, e.g. for explaining the change of brightness over time *per se*, i.e. without the need to assign significance to point-to-point correspondences. However, as a representation of an underlying physical motion field normal flow is of little value.

Our goal is to extract the physically meaningful motion field for the LV contraction pattern. The usual way to circumvent the aperture problem is to complement data evidence with prior knowledge, or by stipulating some hypothesis about the true motion field. The former type of knowledge is rarely at hand, making the latter the typical ploy in practice. It usually assumes the form of a regularity or smoothness constraint. The rationale is that, due to inertia and coherence of physical objects, physical motion fields tend to be smooth. However, it is clear that there is no guarantee that a regularized solution is everywhere close to the underlying physical motion field. In fact, if the physical motion field exhibits strong variations at some locations, these will not be retrieved correctly, as they are precluded a priori. We therefore aim for a regularisation free solution, but one that is not hampered by missing data evidence (aperture problem).

Florack et al. [9], Niessen et al. [10,11] and Suinesiaputra et al. [12] have proposed a multiscale generalisation of the classical Horn & Schunck’s OFCE [8], emphasizing the intrinsic aspects. By imposing conditions reflecting known facts about simulated object/scene dynamics (using the standard “Translating” and “Diverging Tree” benchmark sequences) they were able to obtain very good performance (relative to standard error measures proposed by Barron et al. [26]) by virtue of exploiting the spatial and temporal scale degrees of freedom (d.o.f.s) of (Gaussian) derivative filters. The weakness of the proposed method is it only improves the way of handling the intrinsic d.o.f.s of the OFCE by incorporating scale in a slick way, but it does not handle the aperture problem in a realistic way. As a result, today’s methods of choice for optical flow extraction mostly follow the regularisation rationale, indeed usually one’s only option. For a state-of-the-art algorithm, cf. Brox et al. Bruhn et al. and Weickert et al. [27–29].

Nearly all existing paradigms use *extrinsic* models, i.e. priors complementing the *intrinsic* d.o.f.s captured by data evidence. As a result performance relies crucially on inherently uncertain assumptions, such as the validity of smoothness. It would be desirable if the missing d.o.f.’s (“tangential flow”, say) could be retrieved by adding *further intrinsic evidence* to the existing evidence, obviating the need for regularisation altogether. This is possible if one is in possession of a second, *independent recording* of the *same spatiotemporal region of interest*. In the present case this can be achieved with the help of suitably chosen MR tagging patterns.

Following this new rationale we exploit the strength of the multiscale OFCE by Florack et al. [9], while at the same time removing its shortcomings. The operational scheme for optical flow extraction makes use of a local polynomial expansion of the flow field (at each point). Here we take a 1st order scheme, in 2+1 spacetime dimensions,

$$U(x, y, t) = u + u_x x + u_y y + u_t t \quad \text{respectively} \quad V(x, y, t) = v + v_x x + v_y y + v_t t,$$

in which $u, u_x, u_y, u_t, v, v_x, v_y, v_t$ are eight local parameters of the horizontal, respectively vertical local optical flow field approximation $U(x, y, t)$ and $V(x, y, t)$. We furthermore assume that the source data constitute a scalar field (justified below), and use the appropriate paradigm for this case.

Let f be shorthand for $f(x, y, t; \sigma, \tau)$, the scalar spatiotemporal image sequence as a function of position (x, y) , time t , isotropic spatial scale $\sigma > 0$, and temporal scale $\tau > 0$. We denote its partial derivatives with respect to x, y , and t by self-explanatory subscripts. These are obtained by convolving the raw image sequence $f_0(x, y, t) = f(x, y, t; 0, 0)$ with a corresponding derivative of a normalized Gaussian,

$$\phi(x, y, t; \sigma, \tau) = \frac{1}{2\pi\sigma^2} \frac{1}{\sqrt{2\pi\tau^2}} \exp \left[-\frac{x^2 + y^2}{2\sigma^2} - \frac{t^2}{2\tau^2} \right].$$

The relevant 1st order OFCE is then given by the following linear system for the unknowns $u, u_x, u_y, u_t, v, v_x, v_y, v_t$. Collecting the unknowns in an 8-entry column vector \mathbf{v} , and indicating the 4×8 coefficient matrix by \mathbf{A} , and the inhomogeneous term by the 4-entry column vector \mathbf{a} , we have

$$\mathbf{A} \mathbf{v} = \mathbf{a}, \tag{1}$$

$$\mathbf{A} = \begin{bmatrix} f_x & f_y & f_{xt}\tau^2 & f_{yt}\tau^2 & f_{xx}\sigma^2 & f_{xy}\sigma^2 & f_{xy}\sigma^2 & f_{yy}\sigma^2 \\ f_{xt} & f_{yt} & f_x + f_{xtt}\tau^2 & f_y + f_{ytt}\tau^2 & f_{xxt}\sigma^2 & f_{xyt}\sigma^2 & f_{xyt}\sigma^2 & f_{yyt}\sigma^2 \\ f_{xx} & f_{xy} & f_{xxt}\tau^2 & f_{xyt}\tau^2 & f_x + f_{xxx}\sigma^2 & f_y + f_{xxy}\sigma^2 & f_{xxy}\sigma^2 & f_{xyy}\sigma^2 \\ f_{xy} & f_{yy} & f_{xyt}\tau^2 & f_{yyt}\tau^2 & f_{xxy}\sigma^2 & f_{yyy}\sigma^2 & f_x + f_{xyy}\sigma^2 & f_y + f_{yyy}\sigma^2 \end{bmatrix},$$

$$\mathbf{v} = [u \ v \ u_t \ v_t \ u_x \ v_x \ u_y \ v_y]^T \quad \text{and} \quad \mathbf{a} = -[f_t \ f_{tt} \ f_{xt} \ f_{yt}]^T.$$

For a detailed description and derivation of the general scheme for arbitrary dimensions and approximation orders, and including the case of density images, we refer to the literature, *loc. cit.*

Note that Eq. (1) captures only half³ of the required d.o.f.s (as it should), since no extrinsic model has been invoked. If we are in possession of a second image sequence g of the same spatiotemporal region of interest, then in addition to Eq. (1) we have

$$\mathbf{B} \mathbf{v} = \mathbf{b}, \tag{2}$$

³ In general: codimension of generic isosurface divided by spatial dimension.

6 H.C. van Assen et al.

with \mathbf{B} and \mathbf{b} the g -analogues of \mathbf{A} respectively \mathbf{a} . If the sequences f and g are truly independent, then (by definition) the combined system

$$\mathbf{C} \mathbf{v} = \mathbf{c}, \quad (3)$$

in which

$$\mathbf{C} = \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \end{bmatrix} \quad \text{and} \quad \mathbf{c} = \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix},$$

is non-singular, and establishes a unique solution \mathbf{v} based on data evidence only.

3.2 Applying OFCE to two image sequences simultaneously

This method utilizes the fact that at any point in the tagging pattern, besides an intensity tissue has a *spatial phase*. It is this spatial phase information from two perpendicularly tagged image sequences that is used as input to our intrinsic optical flow method. Schematically, our method is as follows:

1. Acquire two tagged image series with mutually perpendicular tag lines.
2. Calculate the Harmonic Phase representation [13] of the line tagged image sequences. To avoid spatial discontinuities in the input sequences, the sine function is applied to the HARP images (see Fig. 1(c)) before they are put into the solver of system (3). Note that the system (3) is solved for every pixel in every frame of the image sequence, yielding a dense optical flow field, removing the requirement of interpolation. After the sine function is applied to the HARP sequences, the resulting sequences are denoted $f_0(x, y, t)$ and $g_0(x, y, t)$ respectively (see Section 3.1).
3. Evaluate for every pixel at which combination of scales σ_f , σ_g , and τ the combined system (3) is most stable, i.e. yields the most reliable solution. This is done by application of the Frobenius norm [9–11] to system (3).
4. Solve systems (1), (2) for f and g simultaneously, and thus solve system (3)

We assume that the source data constitute a scalar field, because the intensities in the input sequences (the HARP images) represent a spatial phase, contrary to the MR images themselves which represent an accumulation of magnetisation over a volume, and thus constitute a density signal. Therefore, it is justified to apply the scalar paradigm presented in Section 3.1.

4 Experiments and examples

To demonstrate our method it was applied to a set of spatiotemporal images of a tagged human heart (Fig. 2). For comparison, a PCMRI acquisition of the same heart was performed in the same session, and with the same image orientations. Figure 2 shows four systolic frames of the respective acquisitions with the calculated flow field superposed in the left column. In the right column, the magnitudes plots of the corresponding time frames of the PCMRI acquisition are shown, on which the extracted velocities from the PCMRI phase equivalents were

superposed. In the vector plots the directions of the flows are color-encoded for more intuitive perception. This facilitates comparison between the left and right columns. The backgrounds in the left column are obtained by multiplication of $f_0(x, y, t)$ and $g_0(x, y, t)$. The backgrounds and the flow fields were masked based on the magnitude of the complex images computed from the tagged image sequences [13]. The masks in the right column were based on the PCMRI magnitude acquisitions.

Since the input functions $f(x, y, t, \sigma_f, \tau)$ and $g(x, y, t, \sigma_g, \tau)$ depend on the spatiotemporal scales σ_f , σ_g and τ , the appropriate scales have to be selected. In the results presented here, the best scale combinations were automatically selected (for each pixel separately) from 192 combinations of the following ranges (note that the logarithm of the scales have fixed increments):

- $\sigma_f, \sigma_g \in \{1.000, 1.219, 1.486, 1.811, 2.208, 2.692, 3.281, 4.000\}$
- $\tau \in \{1.000, 1.414, 2.000\}$

5 Discussion

Often a smoothness constraint is employed to solve the optical flow problem. This is also the case in many techniques that track only the tag stripes or intersections of perpendicular tag stripes. By interpolation of the found stripes, so-called “virtual tags” are defined, by means of which an initially sparse flow field is densified [6]. In our method, from the very beginning, the optical flow field is regarded at pixel resolution, and thus our method does not require smoothness constraints [8] or flow field interpolation [7] *per se*.

The size of the spectral filter used in the HARP technique, determines the remaining resolution in the spatial domain. By employing a 1-1 CSPAMM procedure, the central peak in k-space is cancelled, allowing for a larger filter, yielding a higher resolution of the flow field [30].

Many methods require regularisation in some sense. Most often regularisation is achieved by application of spline-based interpolation [23, 24]. In the results we have achieved thus far, regularisation has not been employed.

Because we apply our optical flow technique in the spectral domain, we are not restricted by the constant pixel brightness assumption, which is the basis of the optical flow theory in Section 3.1. Thus, the constant brightness assumption is replaced by a constant phase assumption, which seems to be more valid. Intensity modeling requiring knowledge of T_1 (and T_2 , D_0) [24, 25] is therefore not necessary in this method.

In this paper we focussed on the application of optical flow theory for tracking cardiac motion from a combined system of equations constructed from two input image sequences. We did not strive to present a full (quantitative) evaluation, but instead focussed on the theory and a few illustrative examples. However, the qualitative comparison to velocity-encoded MRI shows a striking resemblance between flow fields.

8 H.C. van Assen et al.

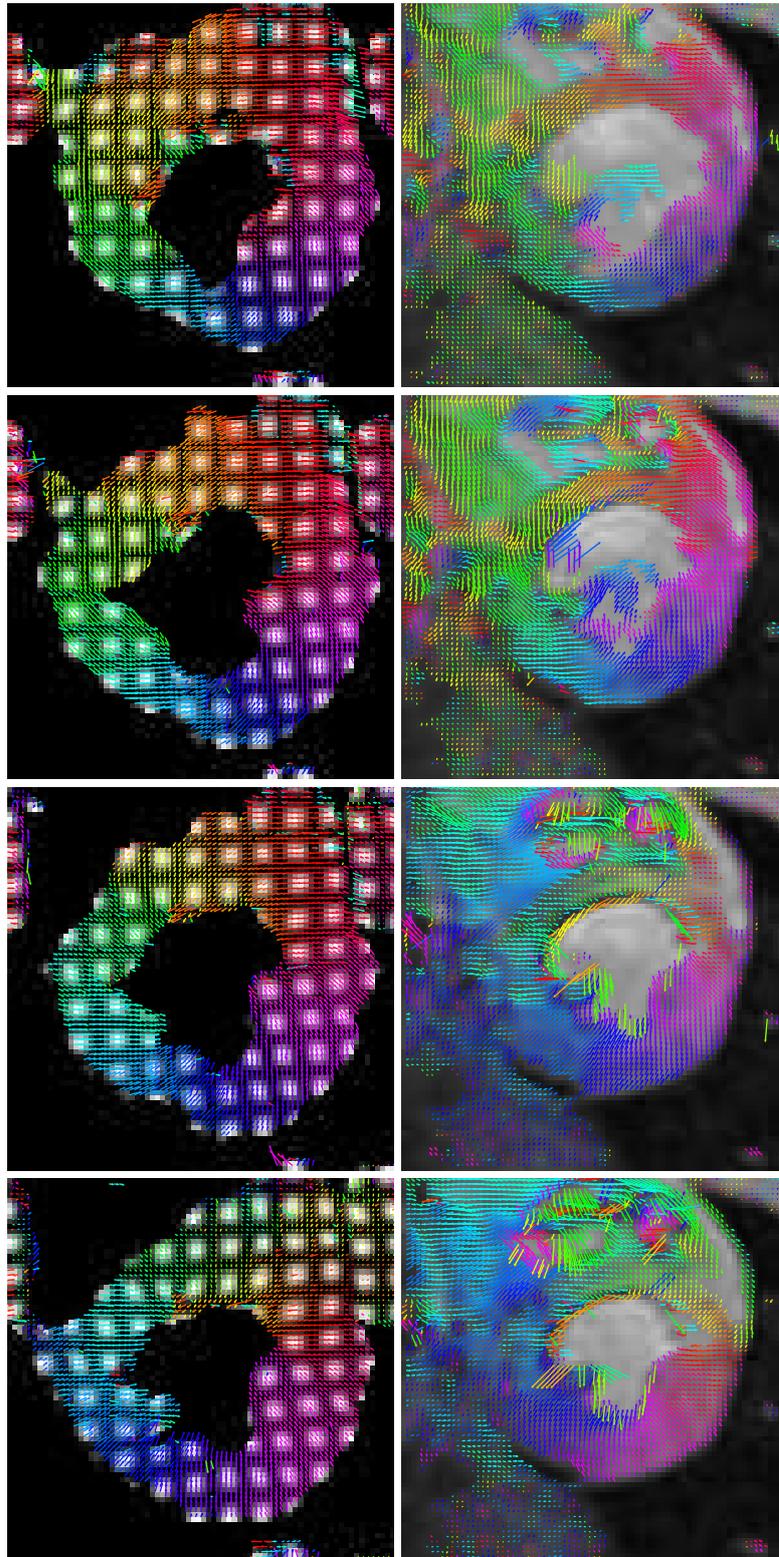


Fig. 2. Flow field of a part of the systole of a human heart (short-axis). The left column shows the flow field produced from the tagging sequences by our method, the right column shows the corresponding frames from the PCMRI sequences. The direction of the flow is color-encoded, i.e., homogeneous color means coherent flow field.

The input “functions” $f(x, y, t; \sigma, \tau)$ and $g(x, y, t; \sigma, \tau)$ depend on the spatial and temporal scales (σ, τ) . Since the size of the tags can vary locally, a.o. due to contraction, automatic scale selection per point was employed using the Frobenius norm. Also the “condition number” of the matrix \mathbf{C} in (3) can possibly serve as criterion for automatic scale selection. More research towards the selected scales and the variation of chosen scales in local neighborhoods should be conducted.

References

1. Stuber, M., Scheidegger, M.B., Fischer, S.E., Nagel, E., Steinemann, F., Hess, O.M., Boesiger, P.: Alterations in the Local Myocardial Motion Pattern in Patients Suffering From Pressure Overload Due to Aortic Stenosis. *Circulation* **100** (1999) 361–368
2. Gunther, S., Grossman, W.: Determinants of ventricular function in pressure-overload hypertrophy in man. *Circulation* **59** (1979) 679–688
3. MacGowan, G.A., Shapiro, E.P., Azhari, H., Siu, C.O., Hees, P.S., Hutchins, G.M., Weiss, J.L., Rademakers, F.E.: Noninvasive Measurement of Shortening in the Fiber and Cross-Fiber Directions in the Normal Human Left Ventricle and in Idiopathic Dilated Cardiomyopathy. *Circulation* **96** (1997) 535–541
4. Axel, L., Dougherty, L.: MR Imaging of Motion with Spatial Modulation of Magnetization. *Radiology* **171**(3) (1989) 841–845
5. Fischer, S.E., McKinnon, G.C., Maier, S.E., Boesiger, P.: Improved Myocardial Tagging Contrast. *Magnetic Resonance in Medicine* **30**(2) (1993) 191–200
6. Axel, L., Chen, T., Manglik, T.: Dense Myocardium Deformation Estimation for 2D Tagged MRI. In: *Proceedings of the Third International Workshop on Functional Imaging and Modeling of the Heart—FIMH 2005* (Barcelona, Spain, June 2005). Volume 3504 of LNCS., Berlin, Springer-Verlag (2005) 446–456
7. Pan, L., Prince, J.L., Lima, J.A.C., Osman, N.F.: Fast Tracking of Cardiac Motion Using 3D-HARP. *IEEE Transactions on Biomedical Engineering* **52**(8) (2005) 1425–1435
8. Horn, B.K.P., Schunk, B.G.: Determining Optical Flow. *Artificial Intelligence* **17** (1981) 185–203
9. Florack, L., Niessen, W., Nielsen, M.: The Intrinsic Structure of Optic Flow Incorporating Measurement Duality. *International Journal of Computer Vision* **27**(3) (1998) 263–286
10. Niessen, W.J., Duncan, J.S., ter Haar Romeny, B.M., Viergever, M.A.: Spatiotemporal analysis of left ventricular motion. In: *Medical Imaging 95*, San Diego, SPIE (February 1995) 192–203
11. Niessen, W.J., Duncan, J.S., Nielsen, M., Florack, L.M.J., ter Haar Romeny, B.M., Viergever, M.A.: A multiscale approach to image sequence analysis. *Computer Vision and Image Understanding* **65**(2) (February 1997) 259–268
12. Suinesiaputra, A., Florack, L.M.J., Westenberg, J.J.M., ter Haar Romeny, B.M., Reiber, J.H.C., Lelieveldt, B.P.F.: Optic flow computation from cardiac MR tagging using a multiscale differential method—a comparative study with velocity-encoded mri. In: *Proceedings of the Sixth International Conference on Medical Image Computing and Computer-Assisted Intervention—MICCAI 2003* (Montréal, Canada, November 2003). LNCS, Berlin, Springer-Verlag (2003) 483–490
13. Osman, N.F., Kerwin, W.S., McVeigh, E.R., Prince, J.L.: Cardiac Motion Tracking Using CINE Harmonic Phase (HARP) Magnetic Resonance Imaging. *Magnetic Resonance in Medicine* **42**(6) (1999) 1048–1060

10 H.C. van Assen et al.

14. Zerhouni, E.A., Parish, D.M., Rogers, W.J., Yang, A., Shapiro, E.P.: Human heart: tagging with MR imaging—a method for noninvasive assessment of myocardial motion. *Radiology* **169**(1) (1988) 59–63
15. Ryf, S., Tsao, J., Schwitter, J., Stuessi, A., Boesiger, P.: Peak-Combination in HARP: A Method to Correct for Phase Errors in HARP. *Journal of Magnetic Resonance Imaging* **20** (2004) 874–880
16. Sampath, S., Derbyshire, J.A., Atalar, E., Osman, N.F., Prince, J.L.: Real-Time Imaging of Two-Dimensional Cardiac Strain Using a Harmonic Phase Magnetic Resonance Imaging (HARP-MRI) Pulse Sequence. *Magnetic Resonance in Medicine* **50**(1) (2003) 154–163
17. Aletras, A.H., Ding, S., Balaban, R.S., Wen, H.: DENSE: Displacement Encoding with Stimulated Echoes in Cardiac Functional MRI. *Journal of Magnetic Resonance* **137** (1999) 247–252
18. Ozturk, C., Derbyshire, J.A., McVeigh, E.R.: Estimating Motion from MRI Data. *Proc. of the IEEE* **91**(10) (2003) 1627–1648
19. Bryant, D., Payne, J.A., Firmin, D.N., Longmore, D.B.: Measurement of flow with NMR imaging using gradient pulse and phase difference technique. *Journal of Computer Assisted Tomography* **8** (1984) 588–593
20. Gotte, M.J., van Rossum, A.C., Twisk, J.W.R., et al.: Quantification of regional contractile function after infarction: strain analysis superior to wall thickening analysis in discriminating infarct from remote myocardium. *Journal of the American College of Cardiology* **37** (2001) 808–817
21. Delhaas, T., Kotte, J., van der Toorn, A., Snoep, G., Prinzen, F.W., Arts, T.: Increase in Left Ventricular Torsion-to-Shortening Ratio in Children With Valvular Aorta Stenosis. *Magnetic Resonance in Medicine* **51** (2004) 135–139
22. Dougherty, L., Asmuth, J.C., Blom, A.S., Axel, L., Kumar, R.: Validation of an Optical Flow Method for Tag Displacement Estimation. *IEEE Transactions on Medical Imaging* **18**(4) (1999) 359–363
23. Kraitchman, D.L., Young, A.A., Chang, C.N., Axel, L.: Semi-automatic tracking of myocardial motion in MR tagged images. *IEEE Transactions on Medical Imaging* **14**(3) (1995) 422–433
24. Prince, J.L., McVeigh, E.R.: Motion estimation from tagged MR image sequences. *IEEE Transactions on Medical Imaging* **11**(2) (1992) 238–249
25. Gupta, S.N., Prince, J.L.: On Variable Brightness Optical Flow for Tagged MRI. In Bizais, Y., ed.: *Information Processing in Medical Imaging: 14th International Conference, Ile de Berder, France, June 1995*, Dordrecht, Kluwer, ISBN 0-7923-3593-7 (1995)
26. Barron, J.L., Fleet, D.J., Beauchemin, S.S.: Performance of optical flow techniques. *International Journal of Computer Vision* **12**(1) (1994) 43–77
27. Brox, T., Bruhn, A., Papenber, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In Pajdla, T., Matas, J., eds.: *Proc. ECCV. Volume IV of LNCS.*, Berlin, Springer-Verlag (May 2004) 25–36
28. Bruhn, A., Weickert, J., Feddern, C., Kohlberger, T., Schnörr, C.: Relations between regularization and diffusion filtering. *IEEE Transactions on Image Processing* **14**(5) (2005) 608–615
29. Weickert, J., Schnörr, C.: Variational optic flow computation with a spatio-temporal smoothness constraint. *Journal of Mathematical Imaging and Vision* **14**(3) (2001) 245–255
30. Kuijjer, J.P.A., Jansen, E., Marcus, J.T., van Rossum, A.C., Heethaar, R.M.: Improved harmonic phase myocardial strain maps. *Magnetic Resonance in Medicine* **46**(5) (2001) 993–999

3-D Non-Linear Finite Element Analysis of Normal Pressure Hydrocephalus

Tonmoy Dutta-Roy, Adam Wittek, Karol Miller

Intelligent Systems for Medicine Laboratory, School of Mechanical Engineering, M050,
University of Western Australia, 35 Stirling Highway, Crawley, WA, Australia 6009
{tduttar, kmiller, adwit}@mech.uwa.edu.au

Abstract. This paper presents the mechanics of Normal Pressure Hydrocephalus (NPH) growth using a computational approach. We generated a generic 3-D mesh of a healthy human brain and treated the brain parenchyma as single and biphasic continuum with non-linear constitutive law undergoing finite deformations. Contact boundary conditions constrained the brain which is enclosed in a skull. We loaded the brain using transmante pressure difference. Non-linear, implicit, Finite-Element (FE) procedures in time domain were used to obtain the deformations for the brain and ventricles. We propose that for modelling NPH, there is no significant advantage gained by using biphasic continuum to model brain parenchyma and that single phase continuum is adequate. We obtained almost equal ventricular volume for both single and biphasic treatment of brain parenchyma under same loading condition. The use of single phase continuum simplified the mathematical description for the model and resulted in large saving of computational time.

1 Introduction

Overlap of symptoms and diagnostic findings between Normal Pressure Hydrocephalus (NPH) and other neurodegenerative diseases (Alzheimer's etc) makes diagnosis of NPH a reoccurring problem for clinicians. Hakim and Adams [1] were the first to identify the condition of NPH. Currently, clinical and diagnostic findings of neurosurgeons in combination with engineering principles enhance the diagnosis of NPH [2, 3, 4 and 5], but these approaches offer no insight into NPH growth mechanics.

Hakim [6] proposed a "sponge" type model of brain parenchyma for NPH growth but without any mathematical formulations. Nagashima et al. [7] and Péna et al. [8] utilised coupled pore fluid diffusion and stress analysis (biphasic approach) for a linear elastic model of brain parenchyma (porous medium) undergoing infinitesimal deformations on a 2-D horizontal brain slice obtained from a brain atlas. Kaczmarek et al [9] used finite deformation biphasic theory on simplified brain geometry (cylindrical) and obtained an analytical solution. Taylor and Miller [10] utilised reassessed brain parenchyma elastic modulus and finite deformation biphasic theory on realistic 2-D brain geometry. Apart from NPH analysis, Miga et al. [11], Miga et al. [12], Paulsen et al. [13], Platenik et al. [14] and Lunn et al. [15] used biphasic

approach for intra-operative image registration of brain deformation during neurosurgery. They treated brain parenchyma as linear elastic and used infinitesimal deformation theory [11, 12, 13, 14 and 15].

In all these works [7, 8, 9, 10, 11, 12, 13, 14 and 15], the brain parenchyma was treated as linear elastic. The assumption regarding infinitesimal deformation [7, 8, 11, 12, 13, 14 and 15] was violated during NPH formation and brain deformation during neurosurgery, due to large deformations in the brain parenchyma. For correct understanding of NPH growth mechanics, finite deformation formulations and constitutive law (e.g. hyperelastic) which can handle large strains (> 20%) encountered during NPH is required. The outer surface of the brain parenchyma was assumed to be fixed to the skull [7, 8, 9 and 10]. As a result, displacement of the brain outer surface was not possible. This is an oversimplification of the brain-skull interaction. For complete understanding of NPH, proper boundary conditions between the brain and the skull should be included in the model [16]. We addressed the deficiencies pointed above by using fully non-linear (geometric, material and boundary) model for our simulations. To the best of our knowledge, this is the first 3-D, non-linear model investigating NPH growth mechanics.

Section 2 includes descriptions of the generic brain mesh as well as loading and boundary conditions for both single and biphasic cases used in our simulation. Element types and formulations for single and biphasic continuum is given in section 3. We detail the results in section 4. Comprehensive discussions and summary of our main findings is in section 5.

2 Biomechanical Model

2.1 Brain Mesh

The brain mesh is shown in Fig 1. We created the generic mesh of the healthy human brain by modifying person specific brain mesh [16] using Hypermesh (Altair Engineering, USA) pre-processing software. Table 1 presents the values for brain and ventricular volume of a healthy human brain [17]. Brain and ventricular volumes in our mesh were consistent with the values given in Table 1. As the brain is approximately symmetrical, half of the brain was simulated. Ventricular volume for a healthy human in our simulations was 14cm^3 . NPH was deemed developed when ventricular volume increased from 14cm^3 to more than 58cm^3 (Table 1).

Table 1. Brain and Ventricular Volumes for Healthy and NPH Cases (adapted from Matsumae et al. [17])

Case	Brain Volume (cubic cm)	Ventricular Volume (cubic cm)
Healthy Brain	1188±104	27±10
NPH	1163±129	116±42

2.1.1 Modified Hyperelastic Material Parameters

The stress-strain behaviour of the brain parenchyma is non-linear with stiffness in compression significantly higher than tension with strong stress-strain rate dependency [18, 19]. To account for these complexities in the brain parenchyma, we chose the hyper-viscoelastic constitutive model proposed by Miller and Chinzei [19].

The time required for NPH growth is relatively long (typically 4 days) [7] when compared to surgical interventions and loading of the brain occurs very slowly. Thus, the strain rate dependency of the brain parenchyma disappeared [10]. Hence, the brain parenchyma was modelled as hyperelastic (Ogden form [20]) given by:

$$W = \frac{2\mu}{\alpha^2} (\lambda_1^\alpha + \lambda_2^\alpha + \lambda_3^\alpha - 3) \quad (1)$$

where, W is the potential function, λ_i 's are the principal stretches, μ is the relaxed shear modulus and α is the material coefficient which can assume any real value without any restrictions. The value μ was 155.77 Pa [10] and the value of α was -4.7 [10]. We considered the brain parenchyma to be homogenous and isotropic for simulation purposes [21] as the brain tissue does not exhibit directional structure, unless the behaviour of very small tissue specimens is of interest [22].

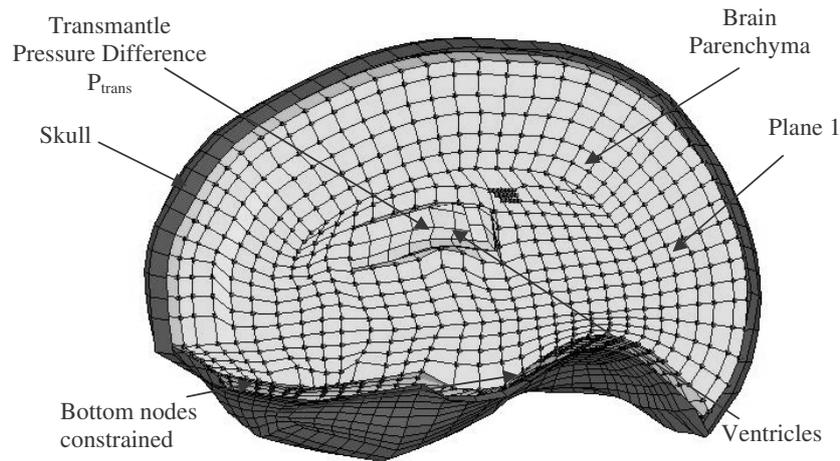


Fig. 1. Brain geometry, pressure loading and applied boundary conditions

2.1.1.1 Biphasic Continuum

Brain was considered to be a sponge like structure with the solid matrix corresponding to neurons and neuroglia and voids being extracellular space [1, 6] occupied by CSF. This is referred to as biphasic [7, 8, 9, 10, 11, 12, 13, 14, 15 and 22] continuum because of presence of two phases: brain parenchyma (solid or porous phase) and CSF (fluid phase). To understand the interaction between brain parenchyma and CSF when loaded by a given transmantle pressure difference, we performed a coupled pore fluid diffusion and stress analysis. The reader may refer to

the works of Nagashima et al. [7], Miller [23], Biot [24], and Bowen [25] for detailed discussions on the mathematical treatment of biphasic continuum.

In our model, brain parenchyma (solid phase) had a Poisson's ratio (ν) of 0.35 [7, 8, 9, 10, and 22] with relaxed hyperelastic material properties (section 2.1.1), initial void ratio of 0.2 [7, 26] and permeability of 1.59×10^{-7} m/sec [9]. It was fully saturated with CSF. CSF (fluid phase) was incompressible, non-viscous with mechanical properties of water.

2.1.1.2 Single Phase Continuum

Due to long development time for NPH, there existed possibility of Cerebrospinal Fluid (CSF) to be absorbed or evacuated in the brain parenchyma, resulting in CSF flow within it and change in brain and ventricular volume. We treated the brain parenchyma as compressible single phase continuum with non-linear constitutive law [19] (generalisation of Ogden rubber [23]), relaxed hyperelastic shear modulus (section 2.1.1) and a low Poisson's ratio of 0.35 [7, 8, 9, 10, and 22] and investigated this effect of compressibility.

2.2 Loading

Load was a transmante pressure difference (P_{trans}) in form of pressure on the ventricular surfaces as shown in Fig 1. There was no pressure acting on the outer surface of the brain. It is a widely held view that transmante pressure difference (P_{trans}) of 1mm of Hg (133.416 Pa) produced the clinical condition of NPH [27, 28] and the same was applied to the ventricular surface to investigate this claim.

2.2.1 Biphasic Continuum

Even though material strain rate effects were absent due to use of hyperelastic constitutive law for the brain parenchyma, rate effects were present because of relative motion between brain parenchyma (solid phase) and CSF (fluid phase). The time period of load application was of importance and transmante pressure difference (P_{trans}) was applied over the development time of NPH (4 days) using a polynomial which provided zero velocity and acceleration respectively at the beginning and end of the loading.

2.2.2 Single Phase Continuum

The time period of load application was not important as we seek a static solution for the single phase continuum and material strain rate effects were absent due to use of hyperelastic constitutive law for the brain parenchyma. Hence, time period of transmante pressure difference (P_{trans}) application was arbitrarily taken to be 10 seconds.

2.3 Boundary Conditions

As the brain is approximately symmetrical about the mid-sagittal axis, half of the brain for both single and biphasic continuum was simulated. The nodes on plane 1 (Fig 1) had symmetrical boundary conditions in YZ plane (no motion allowed for X translation) applied to them. As the brain was resting in the skull, we constrained the

brain bottom nodes in Y and Z translation (Fig 1). A skull enclosed the brain and frictionless, finite sliding; node-to-surface penalty contact between the brain and skull constrained the nodes on the brain outer surface. Following Wittek et al. [16], Sub-Arachnoid's Space (SAS) was accounted by a 3mm gap between the skull and the brain outer surface.

2.3.1 Biphasic Continuum

There exists a pressure gradient between the ventricles and Sub-Arachnoid's Space (SAS) resulting in flow of CSF from ventricles to SAS. We set the pore pressure on the ventricular surface equal to the transmante pressure difference (P_{trans}) and 0 Pa on the outer surface of the brain and implemented this pressure gradient.

3 Computational Model

3.1 Brain Mesh

3.1.1 Biphasic Continuum

5858 porohyperelastic type C3D20PH (20 node triquadratic displacement, trilinear pore pressure, mixed formulation with linear pressure, pore pressure) [29] and 89 type C3D10H (10 node quadratic tetrahedron, mixed formulation with linear pressure, stress displacement) [29] elements discretised the brain parenchyma. We used mixed formulation quadratic tetrahedrons to complete the brick dominated mesh. Volumetric locking was not shown by both C3D20PH and C3D10H type elements.

3.1.2 Single Phase Continuum

The brain parenchyma consisted of 5858 type C3D20H (20 node quadratic brick, mixed formulation with linear pressure, stress displacement) [29] and 89 type C3D10H (10 node quadratic tetrahedron, mixed formulation with linear pressure, stress displacement) [29] elements. As mentioned earlier, we used mixed formulation quadratic tetrahedrons to complete the brick dominated mesh. Type C3D20H did not exhibit volumetric locking for incompressible/nearly incompressible continuum (e.g. brain).

3.2 Skull

The skull consisted of 1006 type R3D4 (4 node, bilinear quadrilateral, 3-D rigid) [29] elements.

3.3 Finite Element Solver

We obtained the solution for NPH growth model using ABAQUS/Standard (Abaqus Inc, Providence, Rhode Island, USA) non-linear finite element code (ABAQUS/Standard, 2004) [32]. The code accounted for geometric, constitutive and contact non-linearities. STATIC (fully non-linear, finite deformation) procedure obtained solution for single phase continuum case and SOILS (fully non-linear, finite

deformation, porohyperelastic) procedure gave solution for the biphasic continuum case. Wu et al. [30] showed the validity of SOILS procedure for hydrated biphasic tissues.

4 Results

Table 2 gives the summary of ventricular cavity volume produced due to application of transmante pressure difference (P_{trans}) of 1mm of Hg (133.416 Pa).

Table 2. Volume of Ventricular Cavity Subjected to Transmante Pressure Difference (P_{trans}) of 1mm of Hg

Case	Poisson's Ratio (ν)	Ventricular Volume (cm ³)
Single Phase	0.35	37.2
Bi-Phase	0.35	36.6

5 Discussions and Conclusions

5.1 Ventricular Volume

Application of 1mm of Hg pressure load to the ventricular surface produced almost equal ventricular cavity volumes for both single and biphasic models (Table 2). The brain parenchyma had a Poisson's ratio (ν) of 0.35 which lead to equally low (467.31 Pa) bulk modulus for both cases. Due to this, the brain parenchyma was equally compressible for each. The long development time for NPH gave adequate time for the wetting fluid (liquid phase: CSF) to flow out of the interstitial voids and subsequently the pore pressure in the biphasic continuum which should have acted against the collapse of the solid phase (porous phase: brain parenchyma) did not do so. It could be convincingly argued from the results in Table 2 that there was no significant advantage gained by modelling brain parenchyma as a biphasic continuum for NPH. Furthermore, application of single phase model significantly reduced computational time. In this study, the computation time for single phase continuum was 160 minutes as compared to 1320 minutes for biphasic continuum.

5.2 Transmante Pressure Difference (P_{trans}) required to produce NPH

Penn et al. [27] and Czosnyka [28] reported that less than 1 mm of Hg (133.416 Pa) transmante pressure difference was adequate to produce the condition of NPH. As mentioned earlier, NPH was deemed developed when ventricular volume increased from 14cm³ to more than 58 cm³ (section 2.1) [17]. Our modelling results (Table 2) clearly showed that ventricular volume was significantly less than 58 cm³ for both single and biphasic continuum when 1mm of Hg transmante pressure difference (P_{trans}) was applied to the ventricular surface. A higher pressure would be required to produce NPH. Thus, if hypothesis of mechanical causes of NPH needs to be

sustained, measurement of transmantle pressure difference (P_{trans}) required to produce NPH should be reassessed.

5.4 Conclusions

Our work showed that application of 1mm of Hg transmantle pressure difference (P_{trans}) resulted in almost equal ventricular volume for both single and biphasic models. Hence, we recommend use of single phase continuum model for brain parenchyma. The use of single phase continuum simplified the mathematical description of the system and lead to shorter computational time. According to our modelling results, 1 mm of Hg transmantle pressure difference (P_{trans}) as reported by other authors was not adequate to produce NPH for both single and biphasic models This suggested that measurement of transmantle pressure difference (P_{trans}) required for producing NPH needed reassessment, if hypothesis of mechanical causes of NPH was to be sustained.

Acknowledgements

Financial support of the William and Marlene Schrader Trust in form of William and Marlene Schrader Post-Graduate Scholarship for the first author is gratefully acknowledged.

References

1. Hakim, S. and Adams, R.D.: The Special Clinical Problem of Symptomatic Hydrocephalus with Normal Cerebrospinal Fluid Pressure: Observations on Cerebrospinal Fluid Hydrodynamics, *Journal of the Neurological Sciences*. 2 (1965), 307-327.
2. Linninger, A. A., Tsakiris, C., Zhu, D. C., Xenos, M., Roycewicz, P., Danziger, Z. and Penn, R.: Pulsatile Cerebrospinal Fluid Dynamics in the Human Brain, *IEEE Transactions on Biomedical Engineering* 52 (2005) 557-565.
3. Shulman, K. and Marmarou, A.: Analysis of Intracranial Pressure in Hydrocephalus, *Developmental Medicine and Child Neurology Suppl* 16 (1968), 11-16.
4. Marmarou, A., Shulman, K. and Rosende, R.M.: A Non-linear Analysis of CSF System and Intracranial Pressure Dynamics, *Journal of Neurosurgery* 48 (1978), 332-344.
5. Czosnyka, M., and Pickard, J.D.: Monitoring and Interpretation of Intracranial Pressure, *Journal of Neurology, Neurosurgery and Psychiatry* 75 (2004), 813-821.
6. Hakim S.: Biomechanics of Hydrocephalus, *Acta Neurológica Latinoamericana* 1 Suppl 1 (1971), 169-194.
7. Nagashima, T., Tamaki, N., Matsumoto, S., Horwitz, B. and Seguchi, Y.: Biomechanics of Hydrocephalus: A New Theoretical Model, *Neurosurgery* 21 (1987), 898-904.
8. Pena, A., Bolton, M.D., Whitehouse, H. and Pickard, J.D.: Effects of Brain Ventricular Shape on Periventricular Biomechanics: A Finite Element Analysis, *Neurosurgery* 45 (1999), 107-118.
9. Kaczmarek, M., Subramaniam, R.P. and Neff, S.R.: The Hydromechanics of Hydrocephalus: Steady-State Solutions for Cylindrical Geometry, *Bulletin of Mathematical Biology* 59 (1997), 295-323.

10. Taylor, Z. and Miller, K.: Reassessment of Brain Elasticity for Analysis of Biomechanisms of Hydrocephalus, *Journal of Biomechanics* 37 (2004), 1263-1269.
11. Miga, M. I., Paulsen, K. D., Lemery, J. M., Eisner, S. D., Hartov, A., Kennedy, F. E. and Roberts, D. W.: Model-Updated Image Guidance: Initial Clinical Experiences with Gravity-Induced Brain Deformation. *IEEE Transactions on Biomedical Engineering*, 18 (1999), 866-874.
12. Miga, M.I., Paulsen, K.D., Hoopes, P.J., Kennedy, F.E. Jnr, Hartov, A., and Roberts, D.W.: In-vivo Quantification of a Homogeneous Brain Deformation Model for Updating Preoperative Image Guiding Surgery, *IEEE Transactions on Biomedical Engineering* 47 (2000), 266-273.
13. Paulsen, K. D., Miga, M. I., Kennedy, F. E., Hoops, P. J., Hartov, A. and Roberts, D. W.: A Computational Model for Tracking Subsurface Tissue Deformation during Stereotactic Neurosurgery, *IEEE Transactions on Biomedical Engineering*, 46 (1999), 213-225.
14. Platenik, L. A., Miga, M. I., Roberts, D. W., Lunn, K. E., Kennedy, F. E., Hartov, A. and Paulsen, K. D.: In Vivo Quantification of Retraction Deformation Modelling for Updated Image-Guidance during Neurosurgery. *IEEE Transactions on Biomedical Engineering*, 49 (2002), 823-835.
15. Lunn, K. E., Paulsen, K.D., Liu, F., Kennedy, F.E., Hartov, A. and Roberts, D.W.: Data-Guided Brain Deformation Modelling: Evaluation of a 3-D Adjoint Inversion Method in Porcine Studies. *IEEE Transactions on Bio-Medical Engineering*, 53 (2006), 1893-1900.
16. Wittek, A., Karol, M., Kikinis, R. and Warfield, S.K.: Patient-Specific Model of Brain Deformation: Application to Medical Image Registration, *Journal of Biomechanics* 40 (2007), 919-929.
17. Matsumae, M., Kikinis, R., Morocz, I., Lorenzo, A.V., Albert, M.S., Black, P. McL., and Jolesz, F.A.: Intracranial Compartment Volumes in Patients with Enlarged Ventricles Assessed by Magnetic Resonance-Based Image Processing, *Journal of Neurosurgery* 84 (1996), 972-981.
18. Miller K. and Chinzei K.: Constitutive Modelling of Brain Tissue: Experiment and Theory, *Journal of Biomechanics* 30 (1997), 1115-1121.
19. Miller, K. and Chinzei, K.: Mechanical Properties of Brain Tissue in Tension, *Journal of Biomechanics* 35 (2002), 483-490.
20. Ogden, R.W.: *Non-Linear Elastic Deformations*, Ellis Horwood Ltd., Chichester, Great Britain, (1984).
21. Ozawa, H., Matsumoto, S., Ohashi, T., Sato, M. and Kokubun, S.: Comparison of Spinal Cord Gray and White Matter Softness: Measurement of Pipette Aspiration Method, *Journal of Neurosurgery* 95 (2001), 221-224.
22. Miller, K., Taylor, Z. and Nowinski, W.L.: Towards Computing Brain Deformation for Diagnosis, Prognosis and Neurosurgical Simulations, *Journal of Mechanics in Medicine and Biology* 5 (2005), 105-121.
23. Miller, K.: Modelling Soft Tissues Using Biphasic Theory-A Word of Caution, *Computer Methods in Biomechanics and Biomedical Engineering* 1 (1998), 216-263.
24. Biot, M.A.: General Theory of Three-Dimensional Consolidation, *Journal of Applied Physics* 12 (1941), 1244-1258.
25. Bowen, R.M.: *Theory of Mixtures*, vol. III, Academic Press, New York, (1976).
26. Sykova, E: Diffusion Properties of the Brain in Health and Disease, *Neurochemistry International* 45 (2004), 453-466.
27. Penn, R.D., Max, C.L., Linninger, A. A., Miesel, K., Lu, S.N. and Stylos, L.: Pressure Gradients in the Brain in an Experimental Model of Hydrocephalus, *Journal of Neurosurgery* 102 (2005), 1069-1075.
28. Czosnyka, M.: *Telephonic Conversation*, (2006).
29. ABAQUS: *ABAQUS/Standard Manual*, Version 6.5.4, Hibbit, Karlsson and Sorenson, Inc. (2004).

30. Wu, J.Z., Herzog, W. and Epstein, M.: Evaluation of the Finite Element Software ABAQUS for Biomechanical Modeling of Biphasic Tissues, *Journal of Biomechanics* 31 (1998), 165-169.

Computational Biomechanics of the Breast: The Importance of the Reference State

Vijay Rajagopal¹, Angela Lee¹, Jae-Hoon Chung¹, Poul M.F. Nielsen¹, and Martyn P. Nash¹

Bioengineering Institute, University of Auckland, NZ, v.rajagopal@auckland.ac.nz

Abstract. Breast cancer diagnosis involves the analysis and interpretation of images of the breast acquired using a variety of imaging modalities such as ultrasound, x-ray mammography and magnetic resonance (MR). Biomechanical models of the breast could help track tissue movement and thereby assist in the interpretation of image patterns across multiple imaging modalities. To this end, we are developing a software system for the creation of subject-specific finite element models of the breast that is systematically validated and optimised for use in the clinical setting. In this paper, we describe a pilot study conducted in order to determine the sensitivity of deformation predictions to the accuracy of the representation of the reference state - an issue that has been overlooked in previous studies. Predictions of the supine gravity-loaded configuration of the breast of a volunteer from (i) a model that uses the prone gravity-loaded state as the reference state (without applying initial stresses to account for gravity), and (ii) a model that uses the breast configuration in neutral buoyancy (immersed in water, assuming the density of the breast is close to that of water) were compared.

The model using the prone gravity-loaded state as the reference configuration predicted the supine gravity-loaded anterior surface configuration with an RMS error of 11.4 mm, while the model using the neutral buoyancy state as the reference configuration predicted the supine configuration with an RMS error of 6.7 mm. These results indicate the selection of a reference state of the breast is an important aspect in the development of a reliable biomechanical model of the breast.

1 Introduction

Diagnostic procedures for breast cancer require interpretation of image patterns by experienced clinicians. Recent studies have highlighted the importance of experience and the examination of the breast using a variety of view-points for intra-modality (x-ray cranio-caudal and medio-lateral, for example) as well as inter-modality (x-ray, MRI, ultrasound) imaging [1, 2]. When imaging the breast using these different modalities, it undergoes significant changes in shape, and the displacements of the internal tissues are governed by the laws of physics.

Image registration techniques have been developed to account for these displacements so that a clinician may more easily track and assess a specific region

of tissue across different views. However, a number of these registration techniques have traditionally based their computations on image-intensity statistics such as mutual information and center of mass [3–5]. These techniques provide transformations between images that are typically heuristic and can prove to be unreliable.

For instance, Tanner *et al* [6] showed that the method proposed in [3] could be improved by introducing a volume-preserving constraint. The need for volume-preserving constraints indicated that non-rigid registration algorithms require further validation to measure the accuracy of the transformations in capturing the tissue deformations during imaging. An important aspect of these techniques that had not been addressed until recently, is the physical-plausibility of the transformations from these heuristic methods. Finite element models of the breast are now being developed to provide physics-based constraints to image registration algorithms [7–11]. Anatomically realistic biomechanical models of the breast can potentially assist clinicians in diagnosing breast cancer by providing them with an image visualisation and patient data management system.

Due to the non-linear nature of large deformations that breast tissue undergo during imaging procedures, large errors in the representation of a reference state will lead to large errors in model predictions. While the sensitivity of model predictions to the choice of boundary conditions and material parameters have been addressed [9, 12], most studies have overlooked the importance of correctly representing the reference state of the breast either: as stress-free in an unloaded configuration; or as initially stressed in a known loaded configuration by providing the three-dimensional stress distribution. Researchers have typically used a gravity-loaded configuration of the breast (such as the prone or supine configuration) without providing information on the three-dimensional stresses in the chosen state [10, 11]. To the authors' knowledge, this is the first study that investigates the use of the unloaded state as the reference configuration and the sensitivity of model predictions to errors in the representation of the reference state.

We first describe the methods used to create finite element models of the breast. We then compare the predictions of a particular gravity-loaded configuration of the breast of a volunteer from: (i) a biomechanical model that uses the prone gravity-loaded configuration as the reference state (with no initial stresses applied to account for gravity); and (ii) a biomechanical model that uses the breast configuration in neutral buoyancy (immersed in water, thus removing the effects of gravity loading) as the reference configuration.

2 Methods

We use our in-house modelling software, CMISS [13], to predict breast deformations. We use the finite element implementation of finite deformation elasticity theory to model breast biomechanics. The reader is referred to standard texts such as [14] and [15] for detailed descriptions of finite elasticity theory and the

finite element method. The following sections briefly describe the methods we use to generate and use subject-specific models of the breast.

2.1 Subject-specific breast geometry

We create subject-specific breast geometries by fitting faces of hexahedral, cubic-Hermite elements to skin and muscle surface data sets that have been segmented from MR images (see Fig. 1). The surfaces are fitted using a least squares approach, as described in [16]. The customisation process has been optimised to run in an automatic fashion (with minimal user intervention) in order to generate breast models rapidly as would be required in a clinical setting. The reader is directed to [7] and [17] for further details on the customisation process and its performance with different breast shapes. We found that geometric finite element models consisting of 24 tri-cubic Hermite elements (with 70 nodes and 1680 total geometric degrees of freedom) fitted the models with an average overall RMS error of 1.5 mm in representing the skin and muscle surfaces. For the mechanics study presented here, the breast was assumed to be homogeneous after [8], thus the geometrical model did not need to account for any structural differences between the internal tissues.

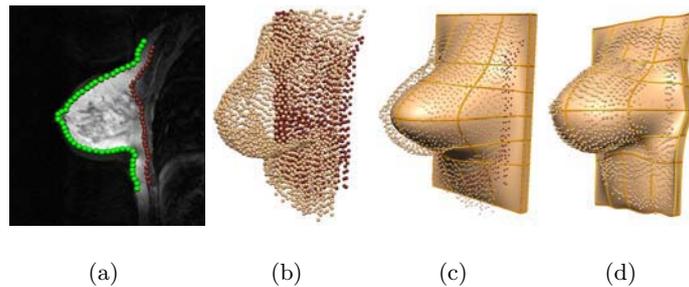


Fig. 1. Process of creating a subject-specific breast model. (a): Segmentation of tissue boundaries. Green points represent the skin surface, and red points represent muscle surface. (b): Dataset of skin (brown points), and muscle (red points) after segmenting an entire MR image set. (c): Initial finite element model whose external faces are to be fitted to the skin and muscle data sets. (d): Finite element model fitted to skin and muscle surfaces.

2.2 Loading and boundary Conditions

Gravity loading was applied as a body force to the reference state of the model (see Section 2.4 regarding the reference configuration). No initial stresses were

applied to the prone gravity-loaded reference configuration to account for gravity. For the purposes of this study, it was assumed that the breast tissues and muscle were firmly attached to the rib cage, thus we applied fixed-displacement boundary conditions at the posterior surface of the model. Previous models have applied displacement boundary conditions to nodes across the entire skin surface to ensure that they matched the outer shape of the breast in the deformed configuration [9, 8, 10]. In contrast, the simulations in the present study are driven by the gravity loading condition alone and thus provide a more robust way of assessing the ability of the model to predict breast deformations.

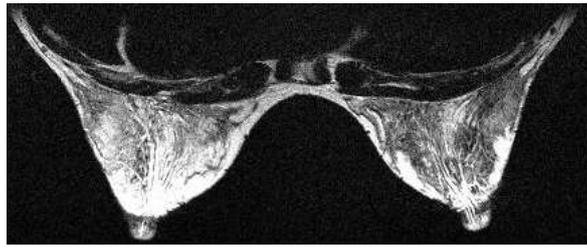
2.3 Subject-specific constitutive parameters

The breast was assumed to be incompressible, homogeneous and isotropic after [8]. A number of experimental studies have been conducted to characterise the mechanical behaviour of breast tissues [18, 19]. Previous researchers have typically used the experimental data from the literature to fit the material constant, c_1 , in the neo-Hookean constitutive equation $W = c_1(I_1 - 3)$, where I_1 is the first principal invariant of the Lagrangian strain tensor [14]. However, it is well known from experiments that breast tissue mechanical properties vary significantly between subjects [20]. Therefore, it is likely to be important to fit the material constant to each individual. We used our existing material parameter optimisation techniques [21] to estimate the value of c_1 using the model of the reference shape together with segmented data from images of different gravity-loaded configurations.

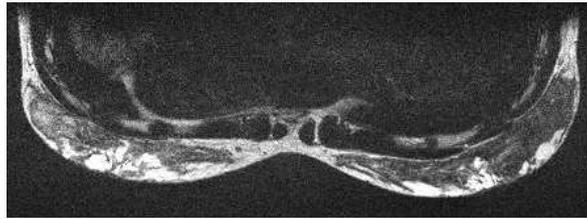
2.4 The reference configuration and its importance

In this paper, we investigate the importance of the selection of a reference state of the breast. We conducted this study by comparing the performances of a model in predicting gravity-loaded configurations when two different representations of the reference state of the breast were used.

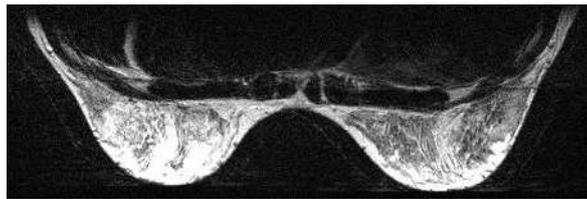
The breast of a volunteer was imaged in four different configurations (see Fig. 2): (i) prone gravity-loaded (ii) supine gravity-loaded (iii) 11.5 degrees ‘‘head up’’ (the volunteer is in a prone position with their back arched slightly upwards, such that the head is raised.) (iv) prone and neutrally buoyant by assuming that the density of the breast is close to that of water and submerging the breast in water. For this last case, we assume that neutral buoyancy offsets the effects of gravity, and more closely represents the stress-free reference configuration of the breast. These images are the first to capture the unloaded shape of the breast and allow us to quantify the importance of the representation of the reference configuration on the accuracy of breast mechanics modelling using clinical data.



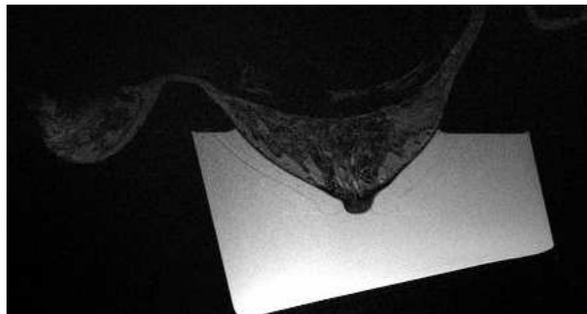
(a)



(b)



(c)



(d)

Fig. 2. MR images of the breast in four different orientations: (a): prone gravity-loaded configuration. (b): supine gravity-loaded configuration. (c): 11.5 degrees head up, where the volunteer is in a prone position with their back arched slightly upwards, such that the head is raised. (d): prone and neutrally buoyant configuration, obtained by assuming that the density of the breast is close to that of water and submerging the breast in water. Only the right breast was immersed in water in this study.

With these image sets, two different sets of simulations were conducted and are described as follows:

- **Using the neutral buoyant configuration as the reference state:** The neutral buoyancy configuration was assumed to represent the reference state of the breast and a model of this configuration was created using the customisation process described in Section 2.1. The model was then subjected to the “head up” gravity loading condition, and the value of c_1 in the neo-Hookean constitutive equation was estimated using non-linear optimisation, with the objective of minimising the error in predicting breast deformation in the “head up” orientation. The accuracy of the model prediction was quantified by comparing the manually segmented head up configuration of the breast skin to the model predicted anterior surface configuration. The segmented points were projected (using a closest orthogonal projection approach [16]) onto the deformed surface of the model, and an RMS error of the data projection vectors was calculated as a performance measure. The supine configuration of the breast was then predicted using the optimal material parameter value and compared to the skin data set that was segmented from the supine configuration MR images.
- **Using the prone gravity-loaded configuration as the reference state:** The same procedures for material parameter optimisation and supine configuration prediction were done as previously, but this time the prone gravity-loaded configuration (with no initial stresses due to gravity applied) was assumed to represent the reference state of the breast.

3 Results

3.1 The neutral buoyancy configuration

The geometric finite element model of the neutrally buoyant breast had an RMS error of 0.78 mm for fitting the skin surface and 1.2 mm for fitting the muscle surface of the volunteer’s breast.

Prior to characterising the mechanical properties of the breast, a displacement solution mesh convergence analysis was performed for the mechanics model by tracking material points in the model and recording their displacements for successive mesh refinements during the convergence analysis. The Euclidean displacements that each material point underwent during the deformation was calculated for each mesh resolution. The RMS errors between successive refinements were then recorded and the mesh corresponding to 112 tri-cubic Hermite elements, as illustrated in Fig. 3 (216 nodes; 5184 geometric degrees of freedom), was chosen as the most appropriate resolution for reliable model predictions.

This mesh was then used to characterise the mechanical behaviour of the breast using the “head up” configuration and the material parameter optimisation technique estimated a value of $c_1 = 0.08$ kPa for this model, which predicted the “head up” configuration with an RMS error of 5.4 mm. The supine configuration was then predicted using this subject-specific model and had an RMS

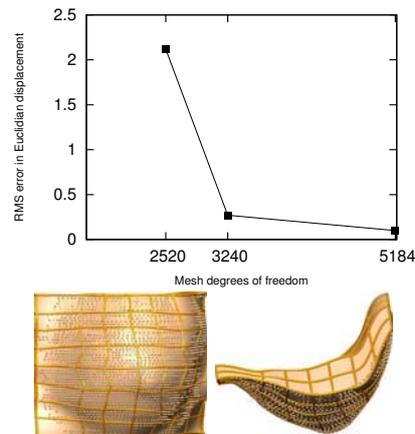


Fig. 3. Top: Convergence of RMS error in Euclidean displacements of selected material points inside the fitted neutral buoyancy model with increasing mesh resolution. Bottom: Chosen resolution for mechanics model predictions.

error of 6.7 mm in capturing the experimentally observed breast deformation (see Fig. 4).

3.2 The prone gravity-loaded configuration

The geometric finite element model of the prone gravity-loaded breast had an RMS error of 0.6 mm for fitting the skin surface and 1.0 mm for fitting the muscle surface of the volunteer’s breast. The mesh was refined in the same manner as in the neutral buoyancy simulation study to ensure reliable displacement converged model predictions. This mesh was then used to characterise the mechanical behaviour of the breast using the “head up” configuration. The material parameter optimisation technique estimated a value of $c_1 = 50$ kPa for this model, which predicted the “head up” configuration with an RMS error of 4.0 mm. The supine configuration was then predicted using this subject-specific model and had an RMS error of 11.4 mm in capturing the experimentally observed breast deformation (see Fig. 5).

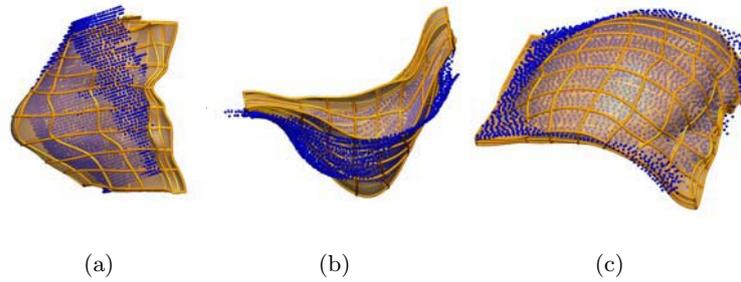


Fig. 4. Model predictions of the breast configuration using the neutral buoyancy state as the reference state. (a and b): Model prediction of the “head up” configuration of the breast compared to the experimentally obtained skin configuration (blue dots) segmented from MRI. (c): Model prediction of the supine configuration compared to the experimentally obtained skin configuration (blue dots) segmented from MRI.

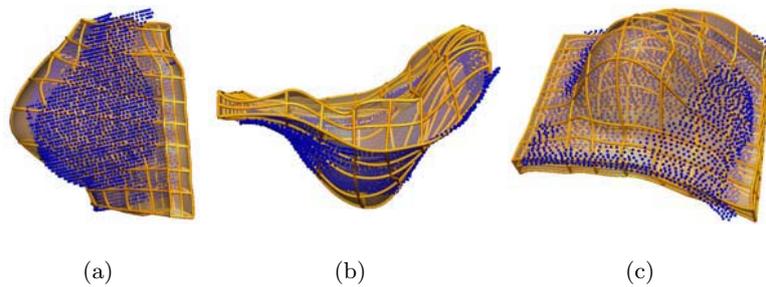


Fig. 5. Model predictions of the breast configuration using the prone gravity-loaded state as the reference state. (a and b): Model prediction of the “head up” configuration of the breast compared to the experimentally obtained skin configuration (blue dots) segmented from MRI. (c): Model prediction of the supine configuration compared to the experimentally obtained skin configuration (blue dots) segmented from MRI.

4 Discussion and Conclusions

A comparison of the RMS errors for predicting the supine configuration using the two different reference states (6.7 mm and 11.4 mm using neutrally buoyant and prone gravity-loaded states, respectively) and the corresponding figures (Fig. 4 and Fig. 5) clearly show that the neutrally buoyant state provides more reliable predictions of breast deformation. Note that this difference in model predictions exists even after adjusting the material parameter c_1 for each set of simulations to remove any bias in results.

The assumption of material homogeneity and the fixed-displacement boundary conditions at the posterior face could also be sources for errors in the model predictions. However, it is clear from this study that by keeping the boundary conditions and the material property assumptions the same and only changing the reference configuration (using the neutral buoyancy configuration instead of the prone gravity-loaded configuration), we reduce the RMS error significantly. Fig. 4 shows that the model captures gross surface deformation characteristics of both “head up” and supine configurations when using the neutral buoyancy state compared to when using the prone gravity-loaded state.

These results indicate that the neutral buoyancy configuration is a good representation of the reference state, but it is not necessarily the best. In future studies we will investigate the validity of the assumption that the density of the breast is close to that of water, along with techniques we have developed to calculate the reference state without experimentally acquiring them in the clinical setting [22]. The acquisition of a variety of breast configurations has given us the ability to systematically validate our modelling techniques and assumptions, and results such as Fig. 4 promise a reliable product that can assist clinicians in combating breast cancer.

References

1. General information on breast cancer. website article by Imaginis-The Breast Cancer Resource: <http://imaginis.com/breasthealth/statistics.asp> (2007)
2. Malur, S., Wurdinger, S., Moritz, A., Michels, W., Schneider, A.: Comparison of written reports of mammography, sonography and magnetic resonance mammography for preoperative evaluation of breast lesions, with special emphasis on magnetic resonance mammography. *Breast Cancer Research* **3** (2001) 55–60
3. Rueckert, D., Sonoda, L., Hayes, C., Hill, D., Leach, M., Hawkes, D.: Nonrigid registration using free-form deformations: application to breast MR images. *IEEE Transactions on Medical Imaging* **18** (1999) 712–721
4. Xiao, G., Brady, J., Noble, J., Burcher, M., English, R.: Nonrigid registration of 3D free hand ultrasound images of the breast. *IEEE Transactions on Medical Imaging* **21** (2002) 405–412
5. Engeland, S., Snoeren, P., Hendriks, J., Karssemeijer, N.: A comparison of methods for mammogram registration. *IEEE Transactions on Medical Imaging* **22**(11) (2003) 1436–1444
6. Tanner, C., Schnabel, J., Degenhard, A., Castellano Smith, A., Hayes, C., Leach, M., Rose, D., Hill, D., Hawkes, D.: Validation of volume-preserving non-rigid

- registration: Application to contrast-enhanced MR-mammography. In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI 2002)*. Lecture notes in Computer Science. Volume 2489-I. (2002) 307–314
7. Rajagopal, V., Chung, J., Warren, R., Highnam, R., Nash, M., Nielsen, P.: Finite element modelling of breast biomechanics: Predicting the effects of gravity. In Miller, K., Poulidakos, D., eds.: *Computational Biomechanics for Medicine. MICCAI 2006 Workshop* (2006) 94–103
 8. Ruiter, N., Stotzka, R.: Model-based registration of x-ray mammograms and MR images of the female breast. *IEEE Transactions on Nuclear Science* **53**(1) (2006) 204–211
 9. Tanner, C., Schnabel, J., Hill, D., Hawkes, D., Leach, M., Hose, D.: Factors influencing the accuracy of biomechanical breast models. *Medical Physics Journal* **33**(6) (2006) 1758–1769
 10. Azar, F., Metaxas, D., Schnall, M.: A deformable finite element model of the breast for predicting mechanical deformations under external perturbations. *Acad Radiol* (2001) 965–75
 11. Yu-Neifert, Q.: *A Three-Dimensional Finite Element Model to Predict The Applicability of Holographic Interferometry to Breast Tumour Detection*. PhD thesis, Graduate Faculty of the University of Akron (1995)
 12. Ruiter, N., Muller, T., Stotzka, R., Gemmeke, H., Reichenbach, J., Kaiser, W.: Automatic image matching for breast cancer diagnostics by a 3D deformation model of the mammary gland. *Biomed Tech (Berl)* **47**(2) (2002) 644–7
 13. CMISS. software program: <http://www.cmiss.org> (2007)
 14. Atkin, R., Fox, N.: *An Introduction to the Theory of Elasticity*. Longman Group (1980)
 15. Zienkiewicz, O., Taylor, R.: *The Finite Element Method: The Basis*. 5 edn. Volume 1. Butterworth-Heinemann (2000)
 16. Nielsen, P.: *The Anatomy of the Heart: A Finite Element Model*. PhD thesis, University of Auckland (1987)
 17. Rajagopal, V.: *Modelling Breast Tissue Mechanics Under Gravity Loading*. PhD thesis, University of Auckland (2007)
 18. Wellman, P.: *Tactile Imaging*. PhD thesis, Harvard Univeerity (1999)
 19. Krouskop, T., Wheeler, T., Kallel, K., Garra, B., Hall, T.: Elastic moduli of breast and prostate tissues under compression. *Ultrasonic Imaging* **20** (1998) 260–274
 20. Sarvazyan, A., Skovoroda, A., Emelianov, S., Fowlkes, J., Pipe, J., Adler, R., Buxton, R., Carson, P.: Biophysical bases of elasticity imaging. *Acoustic Imaging* **21** (1995) 223–240
 21. Chung, J., Rajagopal, V., Nielsen, P., Nash, M.: Computational modeling of the breast during mammography for tumor tracking. In: *Proceedings of SPIE (Medical Imaging)*. Volume 5746. (2005) 817–824
 22. Rajagopal, V., Chung, J., Bullivant, D., Nielsen, P., Nash, M.: Finite elasticity: Determining the reference state from a loaded configuration. *International Journal of Numerical Methods in Engineering* (2007) in press.

High Resolution Ultrasound Elastography: a Dynamic Programming Approach^{*}

Hassan Rivaz, Pezhman Foroughi, Emad Boctor, Richard Zellars,
Gabor Fichtinger and Gregory Hager

Johns Hopkins University, Baltimore, MD, USA

Abstract. This paper presents a 2D strain estimation technique that minimizes a cost function using dynamic programming (DP). The cost function incorporates similarity of echo amplitudes and displacement continuity. The method is capable of creating high quality elastograms at the same resolution as the original RF data. Since tissue deformations are smooth, the incorporation of the smoothness into the cost function results in reduced decorrelation noise. Freehand palpation elastography shows that the method is more robust to signal decorrelation (caused by scatterer motion in high axial compression and non-axial motions of the probe) compared to the correlation techniques. In-vitro experiments depict that the method is able to detect small hard lesions. The method is also suitable for real time elastography.

1 Introduction

Elastography, the display of the spatial variation of elastic modulus of tissue, is an emerging medical imaging method with various medical applications such as tumor detection [1] and ablation monitoring [2]. This paper focuses on static elastography, a well known technique that applies quasi-static compression of tissue and simultaneously images it with ultrasound. Through analysis of the ultrasound images, tissue displacement map can be obtained [3, 4]. Typically, a least squares technique is used to generate a low noise strain estimate (elastogram) from the displacement map.

Most elastography techniques estimate local displacements of tissue based on correlation analysis of radio-frequency (RF) echoes. The resolution of these techniques is low due to the size of the analysis window (approximately 80 samples or 3mm of tissue depth). Decreasing the window size does not improve elastogram resolution as it increases the variance of the least square strain estimator [5]. Large windows are also required to avoid ambiguity in time delay estimation especially when tracking a motion that exceeds a wavelength. At the same time, significant decorrelation within large windows limits the tolerable level of compression. To reduce signal decorrelation, stretching methods have been proposed [6, 7]. Moreover, large errors due to false peaks and smaller errors due to jitter [8] also limit the performance of correlation techniques.

^{*} Supported by Breast Cancer Research Foundation and NSF EEC-9731748.

In work closely related to this paper, Pellet-Barakat et al. [9] have proposed minimizing an energy function that combines constraints of conservation of echo amplitude and displacement continuity. An iterative multiscale approach is applied to minimize the energy function. The method is shown to generate accurate low noise displacement fields. The computation time is reported to be more than one minute for an elastogram that is less than half of the number of pixels in the elastograms generated in this paper. Hence the method is not suitable for real time elastography.

This paper describes an elastography technique based on dynamic programming (DP) for image matching. DP is an efficient method of global optimization [10] and has been extensively used in computer vision for finding correspondences between two images acquired from two cameras in stereo [11], matching deformable contours [12] and word recognition [13]. Devising a DP algorithm for an optimization problem involves the following steps.

1. View the choice of a feasible solution as a sequence of decisions occurring in stages so that the total cost is the sum of the costs of individual decisions.
2. Determine which decisions are possible at each stage.
3. Write a recursion on the optimal cost from the first stage to the final stage.

The remainder of this paper is summarized as follows. Section 2 discusses the methodology and details of the implementation of 1D displacement estimation, followed by subpixel and 2D displacement estimation. Experimental results on phantoms and chicken breast are presented in Section 3. Section 4 concludes the paper and presents the future work.

2 Dynamic Programming Elastography

2.1 1D displacement estimation

We first consider the problem of 1D strain estimation with 1D smoothness regularization. Consider two echo signals $g(i)$ and $g'(i)$ corresponding to two A-lines¹ acquired before and after compression (Figure 1 left), each signal sampled at $i = 1, 2 \dots m$. The distance between the two signals, Δ , can be quantified using sum of absolute differences (SAD), which is computationally inexpensive:

$$\Delta(i, d) = |g(i) - g'(i + d)| \quad (1)$$

where $d_{min} \leq d \leq d_{max}$ is the displacement at the sample i (Figure 1 left) and d_{min} and d_{max} specify the allowed displacement. The smoothness of the displacements is represented as S

$$S(d_i, d_{i-1}) = (d_i - d_{i-1})^k \quad (2)$$

¹ Typical ultrasound image, B-mode image, is composed of multiple vertical lines which are called A-lines.

where d_i is the displacement at the sample i and d_{i-1} is the displacement at the sample $i - 1$ of the $g(i)$. To avoid large jumps in the displacement estimation, it can be shown that the S function should be strictly convex; we choose $k = 2$. The cost function C at each point i and its associated displacement d_i is defined as a recursive function

$$C(i, d_i) = \min_{d_{i-1}} \{C(i-1, d_{i-1}) + wS(d_i, d_{i-1})\} + \Delta(i, d_i) \quad (3)$$

To keep the weight parameter w independent of the ultrasound time gain compensation (TGC), the Δ function was normalized by dividing it by its maximum value. The w value was then varied until a visually good strain image was generated from phantom data and was kept constant for all experiments. Its value, however, can be varied significantly without considerable effect on the strain image. The values of the C function are stored in a $(d_{max} - d_{min} + 1) \times m$ matrix (Figure 1 middle).

Generally, the optimum value of d_{i-1} should be sought in the entire $[d_{min}, d_{max}]$ range. However, since the strain value is low in elastography, it is expected that at each sample (among approximately 1000) of RF data, the change between the displacement of a sample and its previous sample is not more than 1. Therefore, the search range of optimum value for d_{i-1} is limited to the three values of $d_i - 1$, d_i and $d_i + 1$, which results in a significant gain in speed. The optimum value of d_{i-1} is also “memoized” [10] in a function M for later use.

$$M(i, d_i) = \arg \min_{d_{i-1}} \{C(i-1, d_{i-1}) + wS(d_i, d_{i-1})\} \quad (4)$$

The cost function C is calculated for $i = 1 \dots m$. The minimum cost at $i = m$ gives the displacement of this point, which is traced back to $i = 1$ using the memoization function to calculate all the displacements (D).

$$\begin{aligned} D(i) &= \arg \min_{d_i} \{C(i, d_i)\}, \quad i = m \\ D(i) &= M(i+1, D(i+1)), \quad i = 1 \dots m-1 \end{aligned} \quad (5)$$

The displacement map of all A-lines is calculated using the same procedure independently. In Section 2.3, we present a method for coupling adjacent A-lines.

2.2 Subpixel displacement estimation

The displacement function D obtained in the previous section only takes integer values. This means that at a typical 1% strain rate, the D function takes a constant value for 99 samples followed by a changes of 1 in the 100th sample (on average). A smooth strain map can be obtained from D using postprocessing techniques like least squares, sacrificing image resolution. Here, we extend the DP algorithm to refine the displacement estimate to subpixel level.

At each displacement estimate $D(i)$, the pre-compression signal is interpolated around i and is upsampled in the $[i-1, i+1]$ interval by a factor

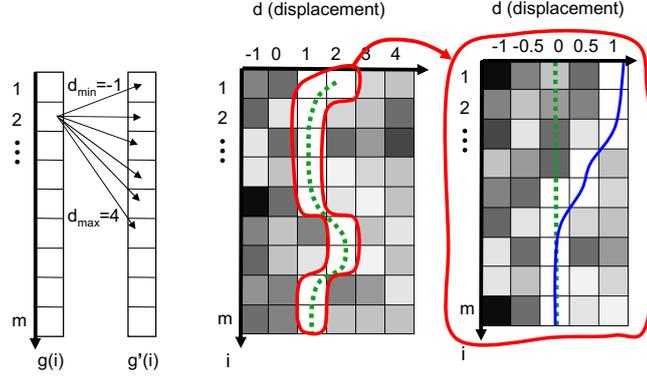


Fig. 1. In the left, values of $g(i)$ and $g'(i+d)$ corresponding to pre and post compression RF data are compared. Middle shows the cost function C of equation 3, with white and black representing low and high cost values respectively. In right, a new cost function around the optimum path of the first stage's cost function (the dashed line) is created, giving a $1/\gamma = 1/2$ pixel displacement accuracy.

of γ . The post-compression signal is also interpolated and upsampled in the $[i + D(i) - 1, i + D(i) + 1]$ interval by the same factor. The DP algorithm is now performed on the two upsampled signals to achieve a subpixel displacement estimation of $1/\gamma$ (Figure 1 right). Repeating the refinement procedure n times results in a refinement factor of $1/\gamma^n$.

2.3 2D displacement estimation

Until now, we have assumed pure axial compression independently estimated on each A-line. However, lateral displacement in a soft material is inevitable even when it undergoes pure axial compression. This displacement is related to the Poisson's ratio, which describes the material compressibility. Also, freehand palpation is rarely a pure compression and thus also results in non-axial tissue motion. As a result, a 2D smoothness regularization that considers the displacements between adjacent A-lines is more natural. The DP algorithm of Section 2.1 is modified here to allow for 2D displacement estimation and 2D smoothness.

Assuming that ultrasound images consist of n A-lines, the distance between the pre and post compression signals is

$$\Delta(i, j, d_a, d_l) = |g_j(i) - g'_{j+d_l}(i + d_a)| \quad (6)$$

where $d_{a,min} \leq d_a \leq d_{a,max}$ and $d_{l,min} \leq d_l \leq d_{l,max}$ are the axial and lateral displacements respectively and $j = 1 \dots n$ refers to j^{th} A-line and $i = 1 \dots m$.

$$S(d_{a_i}, d_{l_i}, d_{a_{i-1}}, d_{l_{i-1}}) = (d_{a_i} - d_{a_{i-1}})^2 + (d_{l_i} - d_{l_{i-1}})^2 \quad (7)$$

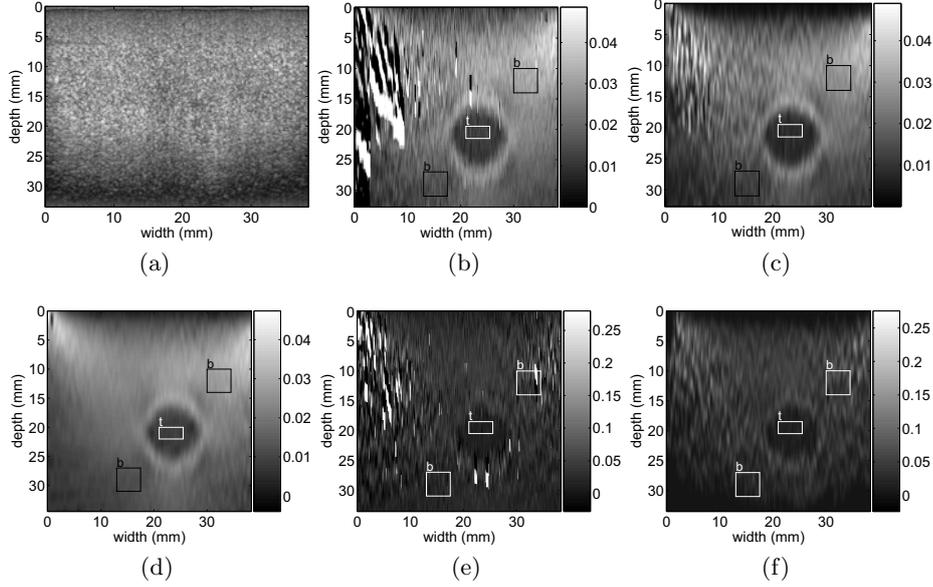


Fig. 2. B-mode image (a) and elastograms obtained from freehand palpation of the breast elastography phantom using cross correlation (b), 1D DP (c), 2D DP (d), cross correlation with B-mode data (e) and 1D DP with B-mode data (f). The rectangles show CNR calculation windows (t and b refer to target and background).

is the smoothness regularization with subscripts a and l referring to axial and lateral. The cost function at the i th sample of the j th A-line is

$$C_j(d_a, d_l, i) = \min_{\delta_a, \delta_l} \left\{ \frac{C_j(\delta_a, \delta_l, i-1) + C_{j-1}(\delta_a, \delta_l, i)}{2} + wS(d_a, d_l, \delta_a, \delta_l) \right\} + \Delta(d_a, d_l, i) \quad (8)$$

For memoization, δ_a and δ_l values that minimize the cost function are stored for all d_a , d_l and i values. The specific form of the cost function allows the calculation of the displacement of each A-line using the cost values of the previous A-line. The cost function of the j th line, $C_j(d_a, d_l, i)$, is calculated and is minimized, resulting in its displacement map. The $C_j(d_a, d_l, i)$ function is also used for the calculation of the next cost function $C_{j+1}(d_a, d_l, i)$ and is deleted from the memory afterwards. This makes the amount of memory required to store the cost function values independent of the number of A-lines.

3 Results and Discussion

RF data was acquired from an Antares Siemens system (Issaquah, WA) with a 7.27MHz linear array at a sampling rate of 40MHz. For comparison, strain im-

ages were also calculated using a standard cross correlation method with a 3mm window size and 80% overlap and a three point parabolic interpolation to find the subsample location of the correlation peak [14, 15]. Linear regression with a 30-sample window is performed on the displacement field to calculate strain. Normalization was performed on the elastogram obtained from cross correlation to eliminate outliers in the strain estimation (DP does not require normalization because of global optimization). The signal to noise ratio (SNR) and contrast to noise ratio (CNR) at the specified target and background windows were calculated according to [3] as follows

$$\text{CNR} = \frac{C}{N} = \sqrt{\frac{2(\bar{s}_b - \bar{s}_t)^2}{\sigma_b^2 + \sigma_t^2}}, \quad \text{SNR} = \frac{\bar{s}}{\sigma} \quad (9)$$

where \bar{s}_b and \bar{s}_t are the spatial strain average of the target and background, σ_b^2 and σ_t^2 are the spatial strain variance of the target and background, and \bar{s} and σ are the spatial average and variance of a window in the strain image respectively.

In the first experiment, a breast elastography phantom (CIRS, Norfolk, VA) with a lesion of 10mm diameter and three times stiffer than the background was palpated freehand. In consecutive images, where axial compression is low and there is little non-axial motion, both methods perform well. However, as the axial compression and non-axial motion increase, the DP method outperforms the cross correlation method. Figures 2 (a)-(d) show the B-mode and three strain images obtained with cross correlation, 1D DP and 2D DP. A high level of lateral motion, approximately 2 A-lines, at the top left of the image and high axial strain cause the cross correlation method to fail. The CNR value at the specified target and background windows of the cross correlation method, 1D DP and 2D DP were 2.12, 1.78 and 3.14 respectively. The CNR value only considers the strain data in the three windows, which were carefully selected not to contain the noisy parts of the cross correlation elastogram. Therefore, the cross correlation technique gives a better CNR than the 1D DP although its strain image is noisier.

The performance of the DP method using B-mode data was also studied in this experiment. To simulate B-mode data, the envelope of the RF data was calculated using the Hilbert transform and the data was downsampled by a factor of two. To generate elastograms with higher quality the B-mode data was then upsampled by a factor of two for both cross correlation and DP. The Δ estimate in equation 6 was slightly modified for the B-mode analysis, averaging the differences in a 3x3 window around the sample i, j .

$$\Delta(i, j, d_a, d_t) = \frac{1}{9} \sum_{s=-1}^{s=1} \sum_{t=-1}^{t=1} |g_{j+t}(i+s) - g'_{j+t+d_t}(i+s+d)| \quad (10)$$

Figures 2 (e) & (f) illustrate the elastograms obtained from B-mode data using cross correlation and 1D DP respectively showing that the 1D DP method successfully detects the hard lesion. The CNR values for the cross correlation technique and the 1D DP method are 0.16 and 1.02 respectively.

To examine the ability of the method to detect small hard targets, a needle with the diameter of 1.27mm was inserted into chicken breast and the tissue

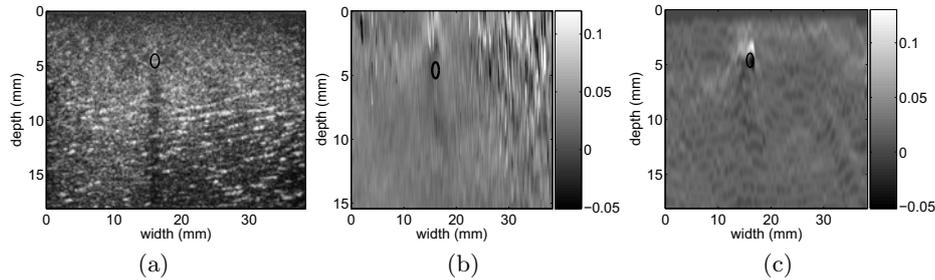


Fig. 3. B-mode image (a) and elastograms obtained from freehand palpation of the in vitro chicken breast with an inserted needle using cross correlation (b), and 2D DP (c). The needle is marked by a circle (appearing as an ellipse because of the scaling ratios). The large strain value over the needle indicates stress concentration.

was palpated freehand. In Figure 3(a), the B-mode image of the tissue is shown; the circle in the three images of Figure 3, whose center is located manually from the B-mode image, marks the needle. Elastograms obtained using the cross correlation and 2D DP techniques are shown in Figures 3 (b) & (c) respectively. High values of strain (white regions) can be observed over the needle in the 2D DP elastogram indicating stress concentration caused by the hard needle.

A CIRS elasticity QA phantom with a 10mm diameter lesion and with the lesion and background Young's elasticity module of 25 kPa and 33kPa respectively was compressed in 24 steps (each step 0.005in). The strain map between the first frame and all other frame was calculated using the cross correlation and DP methods. The SNR metric was calculated in a small window located at the top center of the image, where strain is approximately constant. Figure 4 shows that the DP methods have higher dynamic range, an important elastography performance metric [3]. Since the 1D DP method calculates the displacement at subpixel level, it gives higher SNR at small strains than the 2D DP. This indicates the importance of subpixel displacement estimation for small compression. Implementation of the subpixel displacement estimation for the 2D DP method is underway.

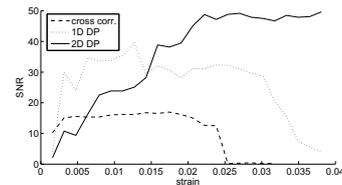


Fig. 4. SNR values of the cross correlation, 1D DP and 2D DP techniques (dashed, dotted and solid curves respectively).

Substituting other computationally more expensive similarity measures like sum of square difference and normalized cross correlation in the Δ function resulted in no significant difference in the performance. Currently, an elastogram of 1000x100 pixels with maximum axial displacement of 10 samples (1% strain) and maximum lateral displacement of ± 1 A-lines takes 1.4sec to generate; 0.25sec for calculating the Δ function and 1.1sec for the DP optimization (on a 3.8GHz P4 CPU). Current implementation is in MATLAB with the Δ function calcu-

lation and the DP optimization in mex functions. If the organ is palpated with the probe, the displacement of the parts of the image close to the probe is very small. Taking advantage of this will result in increasing the speed by a factor of 2. Also a hierarchical search, similar to the subpixel displacement estimation of Section 2.2, and exploiting hardware parallelism can significantly speed the algorithm.

4 Conclusion and Future Work

An elastography algorithm that is significantly more robust to the signal decorrelation (caused by scatterer motion in high axial compression, and lateral and out of plane motions of the probe) than cross correlation techniques is presented. This allows higher axial compression, increasing the dynamic range of the elastogram which is crucial for lesion detection. The CNR and SNR metrics seem to indicate that the regularization creates smooth elastograms while preserving contrast. The regularization also does not appear to damage the ability of the algorithm to detect small targets, like the small Brachytherapy needle. Further work is required to study the effect of regularization on the contrast. Future work will study the effect of uncontrolled tissue motion such as blood flow and respiration on the DP algorithm. The 2D algorithm can be extended to 2D+t, to exploit the cost function in previous time as well as incorporating a 2D+t smoothness regularization.

References

1. Garra, B., Céspedes, E., Ophir, J., Spratt, S., Zuurbier, R., Magnant, C., Pennanen, M.: Elastography of breast lesions: initial clinical results. *Radiology* **202** (1997) 79–86
2. Varghese, T., Zagzebski, J., Lee, F.: Elastographic imaging of thermal lesions in the liver in vivo following radiofrequency ablation: preliminary results. *UMB* **28** (2002) 1467–1473
3. Ophir, J., Alam, S., Garra, B., Kallel, F., Konofagou, E., Krouskop, T., Varghese, T.: Elastography: ultrasonic estimation and imaging of the elastic properties of tissues. *Annu. Rev. Biomed. Eng.* **213** (1999) 203–233
4. Greenleaf, J., Fatemi, M., Insana, M.: Selected methods for imaging elastic properties of biological tissues. *Annu. Rev. Biomed. Eng.* **5** (2003) 5778
5. Lindop, J., Treece, G., Gee, A., Prager, R.: Dynamic resolution selection in ultrasonic strain imaging. *Univ. Cambridge Tech. Rep.* (2006) 1–25
6. Varghese, T., Ophir, J.: Enhancement of echo-signal correlation in elastography using temporal stretching. *IEEE T. UFFC* **44** (1997) 173–180
7. Chaturvedi, P., Insana, M., Hall, T.: 2-d companding for noise reduction in strain imaging. *IEEE T. UFFC* **45** (1998) 179–191
8. Walker, W., Trahey, G.: A fundamental limit on delay estimation using partially correlated speckle signals. *IEEE T. UFFC* **42** (1995) 301–308
9. Pellot-Barakat, C., Frouin, F., Insana, M., Herment, A.: Ultrasound elastography based on multiscale estimations of regularized displacement fields. *IEEE TMI* (04)
10. Bellman, R.: *Applied dynamic programming*. Princeton Univ. Press (1962)

11. Ohta, Y., Kanade, T.: Stereo by intra- and inter-scanline search using dynamic programming. *IEEE PAMI* **7** (1985) 139–154
12. Geiger, D., Gupta, A., Costa, L., Vlontzos, J.: Dynamic programming for detecting, tracking, and matching deformable contours. *IEEE PAMI* **17** (1995) 294–302
13. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. *IEEE T. Sig. Proc* **26** (1978) 43–49
14. Céspedes, I., Huang, Y., Ophir, J., Spratt, S.: Methods for estimation of subsample time delays of digitized echo signals. *Ultrason. Imag.* **17** (1995) 142–171
15. Zahiri, R., Salcudean, S.: Motion estimation in ultrasound images using time domain cross correlation with prior estimates. *IEEE T. Biomed. Eng.* **53** (2006)

Multi-resolution 3D Nonrigid Registration via Optimal Mass Transport on the GPU

Tauseef ur Rehman¹, Gallagher Pryor², John Melonakos¹, and Allen Tannenbaum¹

¹ School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA [tauseef](mailto:tauseef@gatech.edu), [jmelonak](mailto:jmelonak@gatech.edu), tanneba@ece.gatech.edu,

² College of Computing, Georgia Institute of Technology, Atlanta, GA donovang@cc.gatech.edu

Abstract. In this paper we present computationally efficient implementation of the minimizing flow approach for optimal mass transport (OMT) with applications to non-rigid 3D image registration. Our implementation solves the OMT problem via multi-resolution, multigrid, and parallel methodologies on a consumer graphics processing unit (GPU). Although computing the optimal map has shown to be computationally expensive in the past, we show that our approach is almost two orders magnitude faster than previous work and is capable of finding transport maps with optimality measures (mean curl) previously unattainable by other works (which directly influences the accuracy of registration). We give results where the algorithm was used to compute non-rigid registrations of 3D synthetic data as well as intra-patient pre-operative and post-operative 3D brain MRI datasets.

1 Introduction

Image registration and morphing are amongst the most common image processing problems. Registration is the process of establishing a common geometric reference frame between two or more image data sets and is necessary in order to compare or integrate image data obtained from different measurements. A vast amount of literature exists on image registration techniques and we refer the reader to [1, 2] for an overview of this field. In this paper, we approach the registration task by treating it as an optimal mass transport problem. As with other registration techniques, the computational burden associated with this problem is high. We propose a multi-resolution approach for the solution of this problem on the GPU to alleviate this difficulty.

The optimal mass transport problem was first formulated by a French mathematician Gaspard Monge in 1781, and was given a modern formulation in the work of Kantorovich [3] and, therefore, is now known as the Monge-Kantorovich problem. The original problem concerned finding the optimal way to move a pile of soil from one site to another in the sense of minimal transportation cost. Hence, the Kantorovich-Wasserstein distance is also commonly referred to as the Earth Mover's Distance (EMD).

Recently, Haker *et al.* [4, 5] have applied the optimal mass transport approach to certain medical image registration problems. Rigorous mathematical details for their algorithm are given by Angenent *et al.* [6]. Although there have been a number of algorithms in the literature for computing an optimal mass transport, the method by Haker *et al.* computes the optimal warp from a first order partial differential equation, which is a computational improvement over earlier proposed higher order methods and computationally complex discrete methods based on linear programming. However, at large grid sizes and especially for 3D registration the computational cost of even this method is significant.

Though computationally expensive, the OMT method has a number of distinguishing characteristics: **(1)** it is a parameter free method and no landmarks need be specified, **(2)** it is symmetrical (the mapping from image A to image B is the inverse of the mapping from B to A), **(3)** its solution is unique (no local minima), **(4)** it can register images where brightness constancy is an invalid assumption, and **(5)** OMT is specifically designed to take into account changes in densities that result from changes in area or volume.

Contribution. In this paper we extend our previous work [7] and implement the more general formulation of the OMT problem for 3D non-rigid registration based on multi-resolution techniques and using the parallel architecture of the GPU. Although multi-resolution methods have served as critical pieces of registration algorithms in the past, it had yet to be shown that the Optimal Mass Transport problem could be solved in the same manner. Our experimental results show that this is indeed the case, a result which has implications for many fields beyond imaging due to the ubiquitous nature of the OMT problem. We also show that the PDE-based solution to the OMT problem is greatly enhanced by our approach to such an extent that it becomes practical for use on large 3D datasets both in terms of speed and accuracy. Overall, these results show that OMT-based image registration is practical on medical imagery and, thus, merits further investigation as an elastic registration technique without the need of smoothness priors or brightness constancy assumptions.

2 Optimal Mass Transport for Registration

2.1 Formulation of the Problem

We will briefly provide an introduction to the modern formulation of the Monge-Kantorovich problem. We assume we are given, a priori, two sub-domains Ω_0 and Ω_1 of R^d with smooth boundaries, and a pair of positive density functions, μ_0 and μ_1 defined on Ω_0 and Ω_1 respectively. We assume that,

$$\int_{\Omega_0} \mu_0 = \int_{\Omega_1} \mu_1 \quad (1)$$

This ensures that we have same total mass in both the domains. The functions μ_0 and μ_1 in this formulation can be the same as the source and target images, respectively, or a smooth version of them. They can also be scalar fields that are

appropriate for the underlying physical model. We now consider *diffeomorphisms* \tilde{u} from Ω_0 to Ω_1 which map one density to other in the sense that,

$$\mu_0 = |D\tilde{u}|\mu_1 \circ \tilde{u} \quad (2)$$

which we call the mass preservation (MP) property, and write $\tilde{u} \in MP$. Equation (2) is called the *Jacobian equation*. Here, $|D\tilde{u}|$ denotes the determinant of the Jacobian map $D\tilde{u}$, and \circ denotes composition of functions. It basically implies that if a small region in Ω_0 is mapped to a larger region in Ω_1 , then there must be a corresponding decrease in density in order for the mass to be preserved. There may be many such mappings, and we want to pick an optimal one in some sense. Accordingly, we define the squared L^2 Monge-Kantorovich distance as following:

$$d_2^2(\mu_0, \mu_1) = \inf_{\tilde{u} \in MP} \int_{\Omega_0} \|\tilde{u}(x) - x\|^2 \mu_0(x) dx \quad (3)$$

The *optimal MP map* is a map which minimizes this integral while satisfying the constraint given by Equation (2). The Monge-Kantorovich functional, Equation (3), is seen to place a penalty on the distance the map \tilde{u} moves each bit of material, weighted by the material's mass. A fundamental theoretical result [8, 9], is that there is a unique optimal $\tilde{u} \in MP$ transporting μ_0 to μ_1 , and that \tilde{u} is characterized as the gradient of a convex function ω , i.e., $\tilde{u} = \nabla\omega$. This theory translates into a practical advantage, since it means that there are no non-global minima to stall our solution process.

2.2 Computing the Transport Map

We will describe here only the algorithm for finding the optimal mapping \tilde{u} . The details of this method can be found in [4]. The basic idea for finding the optimal warping function is first to find an initial MP mapping u^0 and update it iteratively to decrease an energy functional. When the pseudo time t goes to ∞ , the optimal u will be found, which is \tilde{u} . Basically there are two steps. The first step in this algorithm is to find an initial mass preserving mapping. This can be done for general domains using the method of Moser [10] or the algorithm proposed in [4]. The later method can simply be interpreted as the solution of a one-dimensional Monge-Kantorovich problem in the x-direction followed by the solution of a family of one-dimensional Monge-Kantorovich problems in y-direction and finally solve a family of 2D Monge-Kantorovich problems in the z-direction. The second step is to adjust the initial mapping found above iteratively using gradient descent in order to minimize the functional defined in Equation (3), while constraining u so that it continues to satisfy Equation (2). This process iteratively removes the curl from the initial mapping u and, thereby, finds the polar factorization of u . For details on this technique, please refer to [4]. The overall algorithm is summarized graphically in Figure 1. This same algorithm can be used to compute transport map in arbitrary dimensions the only difference being that in R^2 the problem is a bit simpler where you solve

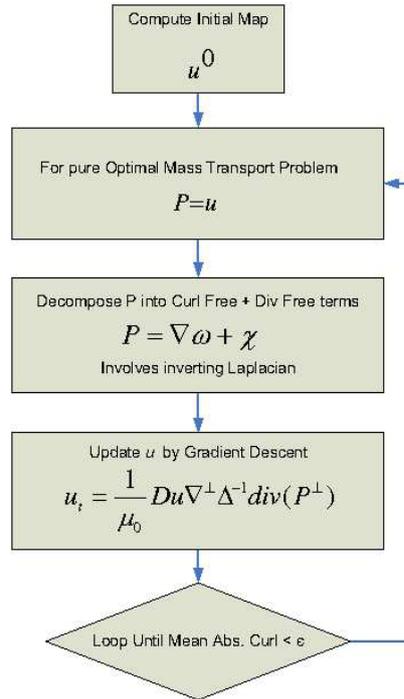


Fig. 1. Optimal Mass Transport Algorithm

the Laplace equation with Dirichlet boundary conditions as compared to solving a Poisson equation with Neumann boundary conditions in higher dimensions. These computations are done in our implementation using Multigrid methods.

3 Implementation

3.1 Multi-resolution Warping

Performing image registration using a multi-resolution approach is widely used to improve speed, accuracy, and robustness. The basic idea is that registration is first performed at a coarse scale. The spatial mapping determined at the coarse level is then used to initialize registration at the next finer level. This process is repeated until it reaches the finest scale. This *coarse-to-fine* strategy greatly improves the registration success rate and also increases robustness by eliminating local optima at coarse scales [11]. Our coarse to fine hierarchy is comprised of three levels (Figure 2).

In our experiments, we found that the coarse-to-fine strategy converges at least twice as fast as the single-resolution solution. Additionally, we found that the coarse to fine method converges to solutions with accuracy (low error metric: mean curl) unattainable by single-resolution methods.

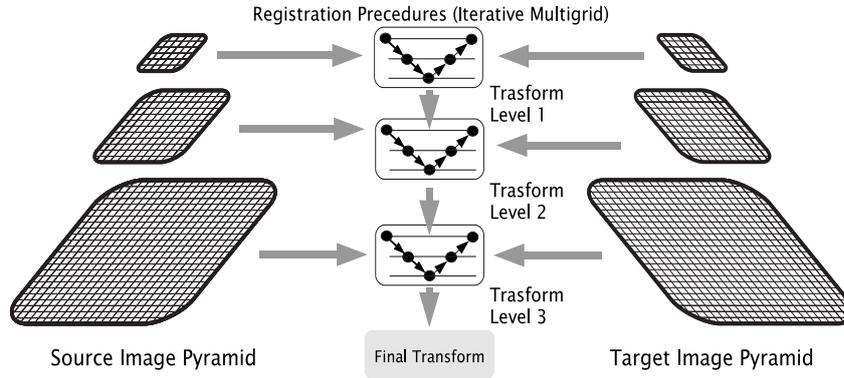


Fig. 2. A Multi-Resolution Registration Scheme. We employ a coarse to fine hierarchy three levels deep with which we solve for an optimal mapping from source to destination data. This method has shown to speed convergence and realize more accurate solutions.

3.2 3D Multigrid Laplacian Inversion

We inverted the Laplacian (a key component of the OMT algorithm) using a 3D multigrid solver. The multigrid idea is very fundamental, it takes advantage of the smoothing properties of the classical iteration methods at high frequencies (Jacobi, Gauss Siedel, SOR etc) and the error smoothing at low frequencies by restriction to coarse grids. The essential multigrid principle is to approximate the smooth (low frequency) part of the error on coarser grids. The non-smooth or rough part is reduced with a small number of iterations with a basic iterative method on the fine grid.

The basic components of multigrid algorithm are discretization, intergrid transfer operators (interpolation & restriction), relaxation scheme and the iterative cycling structure. We used an explicit finite difference scheme for approximating the 3D Poisson equation. This approach uses a 19-point formula on the uniform cubic grid. Relaxation was performed using a parallelizable four-color Gauss-Seidel relaxation scheme. This increases robustness and efficiency and is especially suited for the implementation on the GPU. We used tri-linear interpolation operator for transferring coarse grid correction to fine grids. The residual restriction operator for projecting residual from the fine to coarse grids is the full-weighting scheme. A multigrid $V(2,2)$ -cycle algorithm was used to iterate for solution (Residual max norm $\approx 10^{-5}$). Interested readers are referred to [12–14] for details on implementation of the multigrid methods.

3.3 GPU Implementation

An advantage of our solution to the OMT problem is that it is particularly suited for implementation on parallel computing architectures. Over the past few years, it has been shown that graphics processing units (GPUs; now standard in most

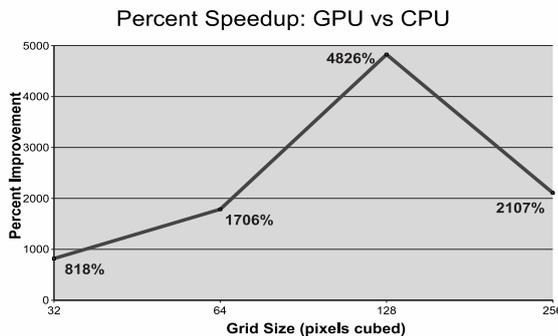


Fig. 3. The GPU realizes an increasing advantage in solving the OMT problem over the CPU as grid size increases up to 128^3 sized grids. Past this point, there is still a large advantage, but a sharp drop is due to memory bandwidth limitations.

consumer-level computers), which are naturally massively parallel, are well suited for these types of parallelizable problems [15, 16].

Taking advantage of these two facts, we implemented our OMT multigrid algorithm on the GPU. The GPU can be considered a massively parallel co-processor and dedicated memory interfacing to the CPU over a standard bus. Modern GPUs are comprised of up to 128 symmetric processors running up to speeds of 1.35Ghz. Their advantage over the CPU in this sense is that while the CPU can execute only one or two threads of computation at a time, the GPU can execute over two orders of magnitude more. Thus, instead of sequentially computing updates on data grids one element at a time, the GPU computes updates on entire grids on each render pass, significantly improving performance (Figure 3). For instance, on a modest Dual Xeon 1.6Ghz machine with an nVidia GeForce 8800 GX GPU (3DMark score of 7200), improvements in speed over our CPU OMT implementation reached 4826 percent on a 128^3 volume data. Presently available GPUs only allow single precision computations, however, this did not affect the stability of the OMT algorithm.

4 Results

We illustrate our registration method using two examples. In the first case, we register a synthetically generated 3D sphere to a deformed (dented) counterpart (Figure 4). In the second case, two 3D brain MRI datasets were registered. The first data set was pre-operative and while the second data set was acquired during surgery and craniotomy and opening of the dura (Figure 5,6). Both were resampled to 256^3 voxels and preprocessed to remove the skull.

In both cases, mean curl of the transport map was reduced to approximately 10^{-3} indicating convergence. However, our coarse-to-fine multigrid implementation on the GPU solves for the optimal transport maps in practical computation times. For instance, registration of the first data (size 128^3) set required 800

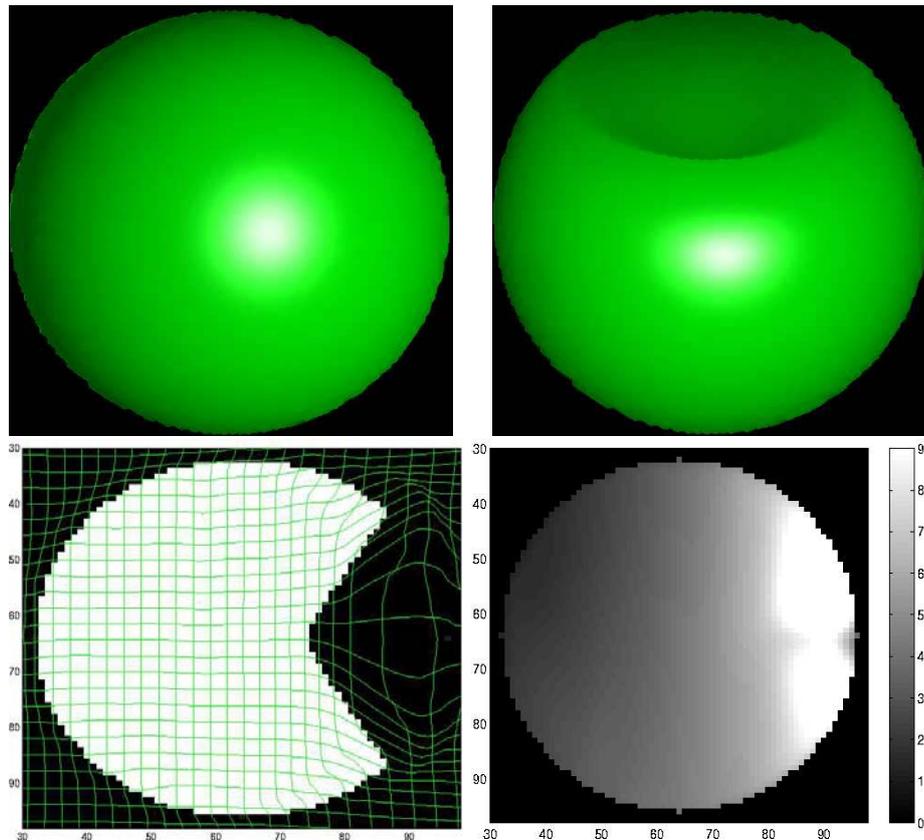


Fig. 4. Synthetic Imagery Results. A sphere is mapped to its deformed counterpart. In the bottom row, the left figure shows the optimal mass transport mapping as a deformation grid overlaid on the destination data. And the right figure shows the magnitude of deformation. Data size 128^3)

iterations of the solver (most at the coarsest scale) requiring less than a minute of computation time. In the second case, (size 256^3) 3600 iterations of the solver were run, requiring less than 15 minutes of computation time.

5 Conclusions

In this paper, we presented a computationally efficient method for 3D image registration based on the classical problem of optimal mass transportation. Many times, global registration methods similar to that presented here are computation intensive making them impractical. However, we have shown that the optimal mass transport is, in fact, a viable solution for elastic registration by achieving low run times on commonly sized 3D datasets on standard desktop computing platforms.

Acknowledgements

This work was supported in part by grants from NSF, AFOSR, ARO, MURI, MRI-HEL as well as by a grant from NIH (NAC P41 RR-13218) through Brigham and Women's Hospital. This work is part of the National Alliance for Medical Image Computing (NAMIC), funded by the National Institutes of Health through the NIH Roadmap for Medical Research, Grant U54 EB005149. Information on the National Centers for Biomedical Computing can be obtained from <http://nihroadmap.nih.gov/bioinformatics>.

References

1. Maintz, J.A., Viergever, M.A.: A survey of medical image registration. *Medical Image Analysis* **2** (1998) 1–57
2. Brown, L.G.: A survey of medical image registration. *ACM Computing Surveys* **24** (1992) 325–376
3. Kantorovich, L.V.: On a problem of monge. *Uspekhi Matematicheskikh Nauk.* **3** (1948) 225–226
4. S.Haker, Zhu, L., Tannenbaum, A., Angenent, S.: Optimal mass transport for registration and warping. *International Journal of Computer Vision* **60**(3) (2004) 225–240
5. S.Haker, Tannenbaum, A., Kikinis, R.: Mass preserving mappings and image registration. In: MICCAI. (2001) 120–127
6. Angenent, S., Haker, S., Tannenbaum, A.: Minimizing flows for the monge-kantorovich problem. *SIAM Journal of Mathematical Analysis* **36** (2003) 61–97
7. Rehman, T., Tannenbaum, A.: Multigrid optimal mass transport for image registration and morphing. In: Proceedings of SPIE Conference on Computational Imaging V. Volume 6498. (2007)
8. Brenier, Y.: Polar factorization and monotone rearrangement of vector-valued functions. *Communications on Pure and Applied Mathematics* **64** (1991) 375–417
9. Gangbo, W., McCann, R.: The geometry of optimal transportation. *Acta Mathematica* **177** (1996) 113–161
10. Moser, J.: On the volume elements on a manifold. *Transactions of the American Mathematical Society* **120** (1965) 286–294
11. Yoo, T.S.: *Insight into Images, Principles and Practice for Segmentation, Registration and Image Analysis*. A. K. Peters Ltd. 2004 (2004)
12. Gupta, M.M., Zhang, J.: High accuracy multigrid solution of the 3d convection-diffusion equation. *Applied Mathematics and Computation* **113** (2000) 249–274
13. W.L.Briggs, Hensen, V., McCormick, S.: *A Multigrid Tutorial*. SIAM (2000)
14. Trottenberg, U., Oosterlee, C., Schüller, A.: *A Multigrid Tutorial*. SIAM (2000)
15. Bolz, J., et al.: Sparse matrix solvers on the GPU: Conjugate gradients and multigrid. In: Proceedings of SIGGRAPH. Volume 22. (2003) 917–924
16. Nolan, G., et al.: A multigrid solver for boundary value problems using programmable graphics hardware. In: Proceedings of SIGGRAPH. (2003) 102–111

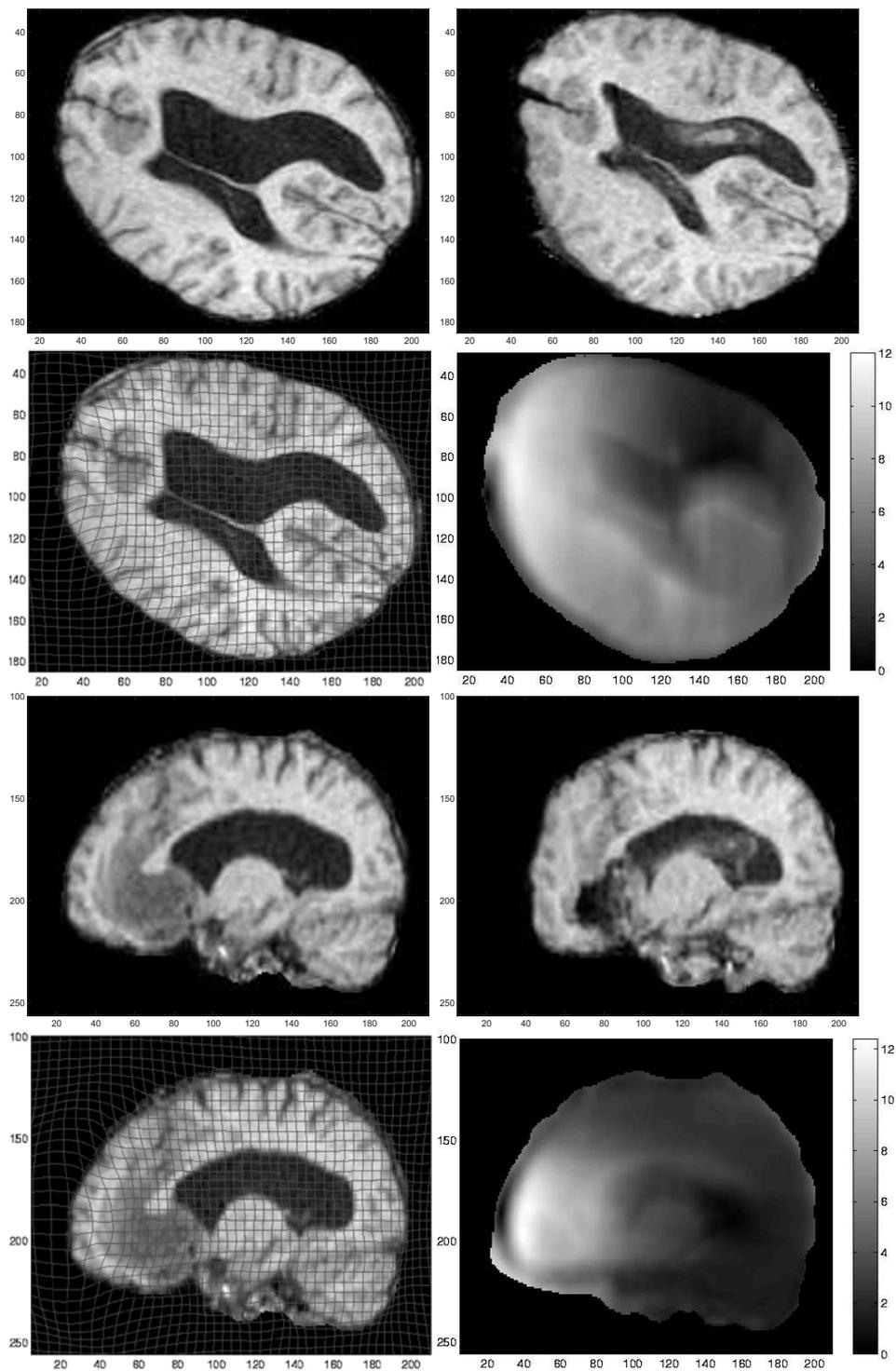


Fig. 5. Brain Sag Registration. The top four figures show the registration results on an axial slice and the bottom four show results for a sagittal slice from the 3D volume. The deformation due to the brain sag after craniotomy and opening of the dura is clearly visible in both the deformation grid and the magnitude of deformation plots. The gravity vector is parallel to the horizontal axis. A rigid shift can also be noticed due to slight displacement of the head during surgery.

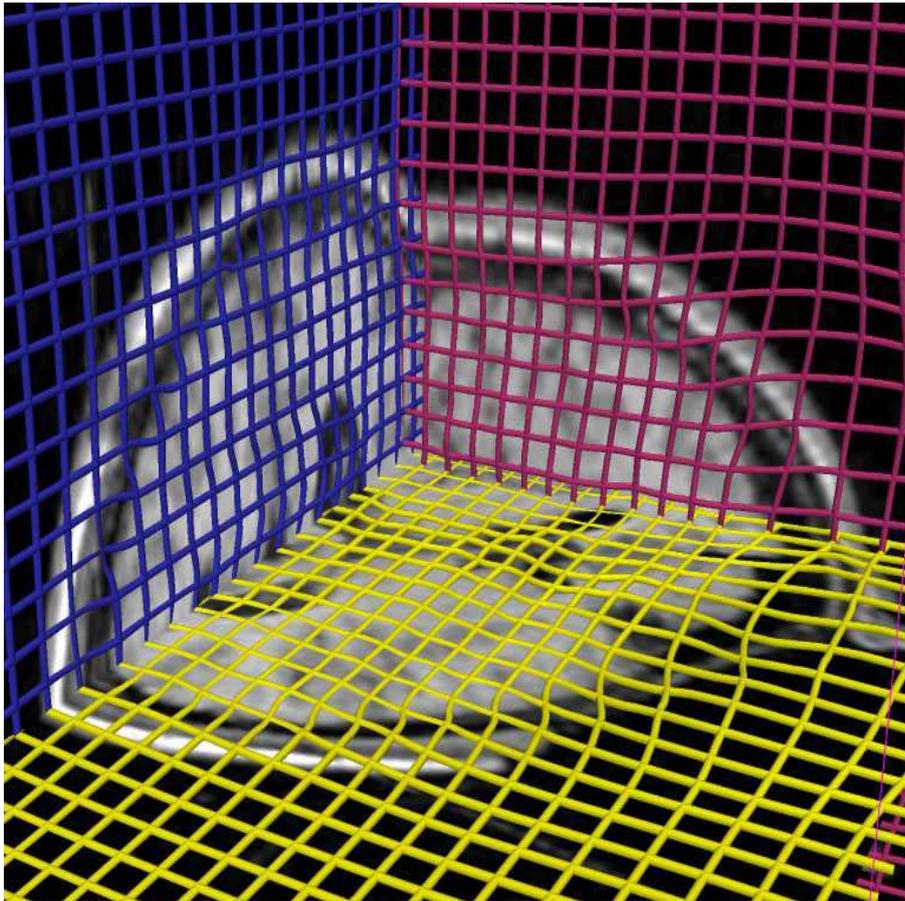


Fig. 6. Brain Sag Registration(3D View). The brain sag is visible in the anterior portion of the brain. (Data size 256^3).

Part 2

Computational Fluid Mechanics and Thermodynamics

Hardware-accelerated Bleeding Simulation for Virtual Surgery

Jing Qin¹, Wai-Man Pang¹, Yim-Pan Chui¹, Yong-Ming Xie¹,
Tien-Tsin Wong¹, Wai-Sang Poon², Kwok Sui Leung³, Pheng-Ann Heng^{1,4}

¹ Dept. of Computer Science and Engineering, The Chinese University of Hong Kong

² Department of Surgery, The Chinese University of Hong Kong

³ Department of Orthopaedics and Traumatology,
The Chinese University of Hong Kong

⁴ Shun Hing Institute of Advanced Engineering,
The Chinese University of Hong Kong

Abstract. In this paper, a novel hardware-accelerated framework is proposed for real-time and realistic simulation of bleeding. In the framework, computational advantage of newly released Physics Processing Unit (PPU) is exploited for simulating the mechanical behavior of blood based on Smoothed Particle Hydrodynamics (SPH). The realism and flexibility of the framework are achieved by adjusting the biomechanical properties of blood and providing proper collision detection between blood and soft tissues such as human skin. A GPU-based marching cubes algorithm is also developed to accelerate the rendering process. Experimental results demonstrate our framework can greatly improve the time performance of bleeding simulation with satisfying realism and has potential to be integrated into many interactive simulators.

1 Introduction

Considerable efforts have been dedicated to improving the performance of computer-assisted surgical simulation. While many simulators have been developed, most of them focus on providing interactive deformable models of soft tissues and rapid haptic rendering. However, real-time and realistic simulation of bleeding, which is an important component of virtual surgery, is given relatively little attention. In reality, bleeding simulation profoundly influences the quality of surgical training and planning, since the handling of various bleeding phenomena requires specific skills from the surgeons. For example, in hysteroscopic simulators, some work has been done to simulate the diffusion of intra-uterine bleeding, which should obscure the view of the surgeons in real surgery. The added realism can help surgeons learn how to perform correct action in this situation[1]. Moreover, bleeding simulation can provide crucial information for surgical training and designing. For example, related information on aneurysm hemodynamics and pathology can be obtained from bleeding simulation for designing treatment of aneurysms. Finally, modeling blood flow is not only essential for surgical simulation but also in diagnosis and treatment planning of individual patient data[2].

The key challenge of real-time bleeding simulation is imitating the behaviors of blood interactively and accurately with sophisticated viscous fluid models and complicated biomechanical properties, but under limitation of computational power. In this paper, a novel framework for real-time bleeding simulation with hardware acceleration is proposed. In the framework, the computational advantage of newly released Physics Processing Unit (PPU) is exploited for simulating the various bleeding phenomena based on Smoothed Particle Hydrodynamics (SPH). To ensure the realism, parameters of vessels are extracted from Chinese Visible Human(CVH) dataset, together with biomechanics of human blood, to generate the particle-based blood model. Collision detection is implemented by constructing proper models based on the built-in algorithm of the PPU. A GPU-based marching cubes algorithm is also developed to accelerate the rendering process.

2 Previous Work

Different approaches for bleeding simulation have been proposed, which can be summarized into two major categories: temporal texture synthesis and physically based imitation. Although the former can mimic certain phenomena such as effusion[3], this technique does not have the full capability of synthesizing heterogeneous texture for complicated bleeding phenomena such as flowing and emitting.

In the past, the adoption of physically based methods is limited by the performance of computational power. The development of computational techniques in fluid dynamics[4], together with the increasing capability of the hardware, found many promising approaches in vascular and bleeding simulation[5][6]. In general, these approaches can be categorized into two classes: grid-based methods and particle-based methods. Although some speedup algorithms have been proposed[7][8], these methods are still not suitable for most real-time and interactive surgical simulation, especially when complicated models are used to ensure the realism. Smoothed Particle Hydrodynamics (SPH) is one of the efficient particle based methods initially proposed for the simulation of astrophysical problems such as fission and stars. This technique was first introduced to the computer graphics community to depict fire and other gaseous phenomena[9]. Müller adopted this method to simulate bleeding for virtual surgery with models of up to 3000 particles[10]. Recently, Daenzer proposed a method to simulate bleeding and smoke in virtual surgery based on simple particle system[11].

3 The Framework

3.1 The Design of Overall Framework

General work-flow of the framework is illustrated in Fig.1. After identifying some necessary parameters such as the central lines, the radius and the vascular wall thickness of the vessels from the CVH dataset, a particle system is created by a module named Particles Distributor(PD). Integrating this particle system with biomechanical properties of blood, a complete blood model represented by an

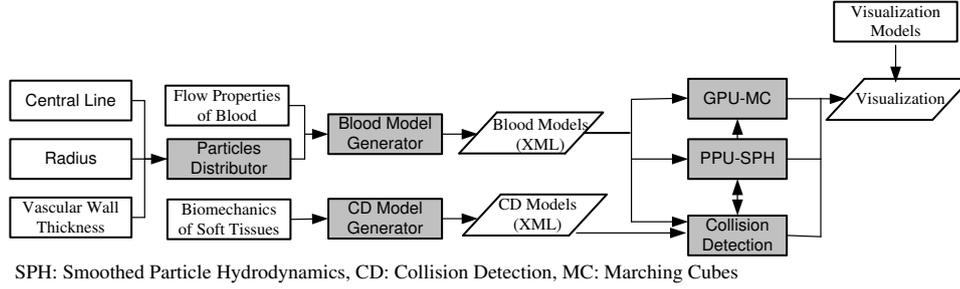


Fig. 1. General work-flow of the framework

XML file for the convenience of data exchange is created. Based on the model, the dynamics of the blood is simulated by PPU-based SPH, while the appearance of the blood is rendered using a GPU-based marching cubes algorithm. Meanwhile, the collision detection models and visualization models of soft tissues such as human skin are also included to achieve better realism.

3.2 Modeling Blood with Blood Biomechanics

In general, the basic equations of fluid motion are obtained by imposing that the fluid conserves mass, momentum and energy. We assume the temperature of our blood model is nearly constant and there is no heat input into the system and no heat source in the region. Therefore, the equation for balance of energy is the same as balance of mechanical energy and work and the energy conservation leads to no new independent equation. In addition, the particle system can inherently conserve mass by setting that both the number of particles and the mass of each particles are constant. Thus we can focus on how to solve the Navier-Stokes which formulates conservation of momentum:

$$\rho \frac{\partial \mathbf{V}}{\partial t} = -\rho \mathbf{V} \cdot \nabla \mathbf{V} - \nabla p + \mu \nabla^2 \mathbf{V} + \rho \mathbf{g} \quad (1)$$

Here ρ is the density; \mathbf{V} is the vector velocity field; the vector operator ∇ is define as $\nabla \equiv \mathbf{i} \frac{\partial}{\partial x} + \mathbf{j} \frac{\partial}{\partial y} + \mathbf{k} \frac{\partial}{\partial z}$; p is the pressure; \mathbf{g} means the body forces such as gravitational forces; μ is viscosity coefficient.

Compared to other fluid simulation approaches, SPH is relatively simple in implementation and fast in computation [12] [13]. Moreover, it can be accelerated with PPU so that sophisticated blood effects can run at a higher frame rate. The basic mechanism of SPH is based on the following equation:

$$A_S(\mathbf{x}) = \sum_j m_j \frac{A_j}{\rho_j} W(\mathbf{r}, h) \quad (2)$$

where a scalar quantity A at position \mathbf{x} is a weighted sum of contributions from neighboring particles. m_j and ρ_j are the mass and density of particle j , respectively, and $\mathbf{r} = \mathbf{x} - \mathbf{x}_j$. A_j is the field quantity at \mathbf{x}_j . The function $W(\mathbf{r}, h)$

is referred to as the smoothing kernel with the properties $\int W(\mathbf{r}, h)d\mathbf{r} = 1$ and $\lim_{h \rightarrow 0} W(\mathbf{r}, h) = \delta(\mathbf{x})$, where h is the support radius and δ is the Dirac-function.

Initially, particles have three quantities including mass, position and velocity. In every time step of the simulation, SPH is used to evaluate the density (ρ_i) and the force density (f_i^p, f_i^v) caused by pressure and viscosity at particle i based on the following equations derived in[14]:

$$\rho_i = \sum_j m_j W(\mathbf{r}, h) \quad (3)$$

$$f_i^p = -\nabla p(r_i) = -\sum_j m_j \left(\frac{p_j + p_i}{2\rho_j} \right) \nabla W(\mathbf{r}, h) \quad (4)$$

and

$$f_i^v = \mu \nabla^2 v(r_i) = \mu \sum_j m_j \frac{\mathbf{V}_j - \mathbf{V}_i}{\rho_j} \nabla^2 W(\mathbf{r}, h) \quad (5)$$

where $\mathbf{r} = \mathbf{x}_i - \mathbf{x}_j$. Adding the gravitational force, we can get the total force densities. Thus based on Newton's second law, we can compute acceleration, velocity and position of every particle. In the computation, we use built-in smoothing kernels ($W(\mathbf{r}, h)$) in the PPU to ensure the stability of the simulation.

By centrifugation, the blood is separated into plasma and cells. The cellular contents are essentially all erythrocytes or red cells. Normally, the red cells occupy about 45%-50% of the blood volume[15]. To simplify the blood model, two kinds of particles are defined. One represents plasma and the other represents red cells. They are randomly distributed in the particle model in a pre-defined ratio. Based on research of blood biomechanics, typical values of the density of blood plasma and blood cells are $1025kg/m^3$ and $1125kg/m^3$ respectively, while the whole blood has a specific density between $1.056kg/m^3$ and $1.066kg/m^3$. More important, blood is a non-Newtonian fluid. In general, the constitutive equation of an isotropic incompressible Newtonian fluid is:

$$\gamma_{ij} = -p\phi_{ij} + 2\mu T_{ij} \quad (6)$$

where

$$T_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right), \quad (7)$$

$$T_{ii} = T_{11} + T_{22} + T_{33} = 0. \quad (8)$$

Here γ_{ij} is the stress tensor, p is the hydrostatic pressure, ϕ_{ij} is the isotropic tensor or Kronecker delta, T_{ij} is the strain-stress tensor, v_i is the velocity component, and μ is a constant called the coefficient of viscosity. However, blood does not perform according to Equation (1) because the μ is not a constant. In our framework, we approximate the non-Newtonian behavior of blood by dynamically adjusting the value of viscosity in regular time steps. Some initial parameters such as the position and number of particles can be set based on CVH datasets.

3.3 Collision Detection between Blood and Soft Tissues

In PPU, Continuous Collision Detection(CCD) mechanism is provided for fast moving objects, which is suitable for simulating interactions between blood and

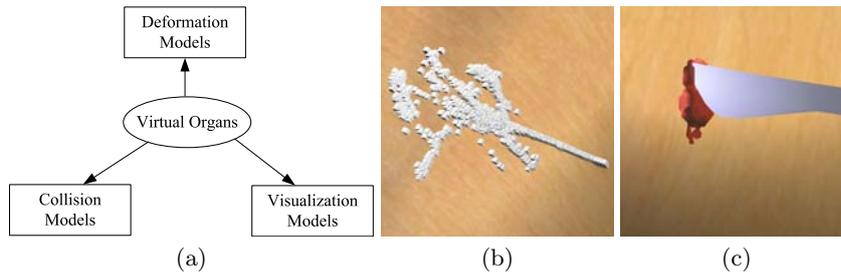


Fig. 2. Multi-model representation and collision detection: (a) Multi-model representation, (b) Particles moving on skin, (c) Collisions between blood, skin, and virtual instrument.

soft tissues such as skin. However, CCD requires that a skeleton mesh with triangles and vertices is embedded in the objects involved. Therefore, soft tissue collision with blood is simultaneously represented with a deformation model, a collision model and a visualization model, which all have different data structures but consistent geometrical positions (Fig. 2(a)). Fig. 2(b) shows the particles moving on the human skin based on the CCD mechanism. Fig. 2(c) illustrates the collisions between blood, skin and virtual instrument. Two built-in key parameters in PPU, Coefficient of Restitution and Adhesion Factor, are allowed to be defined and dynamically adjusted to get different effects. The former controls how much the particles bounce when they collide with soft bodies while the latter determines how easily particles slide along a surface.

3.4 GPU-Accelerated Marching Cubes for Blood Surface Rendering

A GPU-accelerated marching cubes algorithm is intergraded into the framework to avoid having the acceleration achieved by PPU being neutralized by any slow rendering process occurring when the number of the particles is relatively large. Each particle in SPH represents part of the fluid. Therefore, the particles form a color field inside the space (or volume). Direct rendering of the volume is slow and does not result in a convincing appearance. Usually, the surface of the fluid, which is defined by a user-specified isovalue, is first extracted. Then, rendering is performed using the surface elements or mesh. In our system, the marching cube method is employed because it is fast and more easily accelerated by current GPU hardware.

Marching cubes is an algorithm for rendering isosurfaces in volumetric data. The method is performed on each grid cell, which is formed by every eight neighboring voxels in a volume. Details of isosurfaces extraction can be referred to [16]. After determining which edges of the grid intersect the isosurface, triangular patches are created to divide the grid. By connecting the patches from all grids on the isosurface boundary, we have the final surface mesh representation.

Because of the highly parallelizable nature of Marching Cubes algorithm, it has been accelerated using GPU [17][18]. However, due to limitations in GPU capabilities at that time, either some parts of the algorithm need to remain

running on CPU or several passes are required to complete the whole algorithm. Both scenarios introduce overheads and consequently degrade performance.

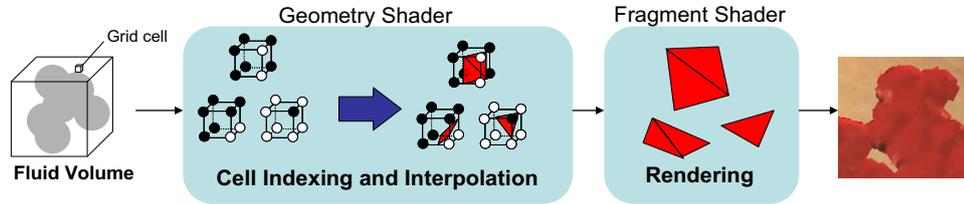


Fig. 3. Blood rendering using GPU accelerated Marching Cube.

Utilizing the new Geometry Shader feature in the newest GPU supporting shader model 4.0, the marching cubes method can be performed entirely inside the GPU in a single pass. Fig. 3. illustrates the whole process divided among different shaders to complete the marching cubes implementation. First, marching cubes grid cells are encoded as point primitives and sent to the geometry shader. The geometry shader operates on each point in parallel and generates a set of triangles used to cut the cells according to the marching cubes code table. Next, the positions of constructed vertices are adjusted according to the voxel values in the grid cells. Finally, the isosurface is illuminated in the fragment shader for display.

4 Experiment Results

4.1 Time Performance

The proposed framework has been implemented on a PC with the following configuration: Pentium 4 Dual Core 3.2GHz CPU, 4GB RAM, NVIDIA GeForce 8800 display adapter and AGEIA PhysXP1 physics processing unit. A series of experiments have been done to compare the time performance of CPU-based bleeding simulation and our hardware-accelerated framework. Tables 1. and 2. show the experimental results. A significant speed improvement is observed when PPU and GPU are used in bleeding simulation. The average frame rate of GPU-base MC is $29.7F/S$ at a volume size of $64 \times 64 \times 64$, which can fulfill most real-time surgical simulation. The corresponding average frame rate of the CPU-based MC is only $5.6F/S$. With the increase of the number of the particles, the advantage of PPU-accelerated SPH is clearly revealed in Table 2. The frame rate of PPU-based SPH stays above $50F/S$ while the CPU-based SPH is under $10F/S$ when the number of particles is more than 5000. A dramatic decrease of time performance of PPU-based SPH occurs when the number of particles is more than 6000. This is maybe due to the PPU using relatively slow PCI bus interface to transfer data. Although this need to be verified through more experiments, in most cases, 6000 or fewer particles are enough to achieve realistic effects.

Table 1. The comparison between GPU and CPU-based MC for blood rendering

Volume Size	Number of Triangles	CPU(F/S)	GPU(F/S)
32×32×32	15568	62.3	129.5
64×64×64	103822	5.6	29.7
128×128×128	967383	0.55	5.3

Table 2. The comparison between CPU and PPU-based SPH for bleeding simulation

Num. of Particles	CPU(F/S)	PPU(F/S)	Num. of Particles	CPU(F/S)	PPU(F/S)
500	63.8	64.1	3000	16.4	55
1000	45.4	63.8	4000	12.3	53
1500	33.6	63.8	5000	9.3	49
2000	24.5	60	6000	7.1	7.1
2500	20.7	60	7000	6.2	6.3

4.2 Visualization

Fig. 4. shows the rendering results of our hardware-accelerated bleeding simulation. It demonstrates that our framework can simulate various bleeding phenomena with acceptable realism. The observed interaction between blood and skin validated the collision detection models. Fig. 5(a) and (b) show the bleeding effects when integrating our framework to a virtual-reality based Orthopedics surgery trainer (refer to attached video). Real-time and realistic bleeding simulation can help trainers learn how to handle various bleeding phenomena during the surgery and therefore improve their experience and skills.

5 Conclusion and Future Work

In this paper, a novel framework for real-time bleeding simulation with hardware acceleration is presented. The relative computational advantage of the Physics Processing Unit (PPU) is exploited for simulating bleeding based on SPH. The realism is achieved by adjusting the biomechanical properties of the simulated blood and providing proper collision detection between the simulated blood and soft tissues. Furthermore, a GPU-based marching cubes algorithm is developed to accelerate the rendering process. Experimental results demonstrate that the framework achieves realistic and real-time simulation of bleeding. Future work includes optimizing the marching cubes algorithm, integrating cutting algorithms and haptic devices and developing simulators based on this bleeding simulation framework.

Acknowledgement The work described in this paper was supported in part by research grants from the Research Grants Council of the Hong Kong Special Administrative Region (Project No. CUHK 4223/04E and CUHK 4453/06M). This work is affiliated with the Virtual Reality, Visualization and Imaging Research Center at The Chinese University of Hong Kong as well as the Microsoft-CUHK Joint Laboratory for Human-Centric Computing and Interface Technologies.

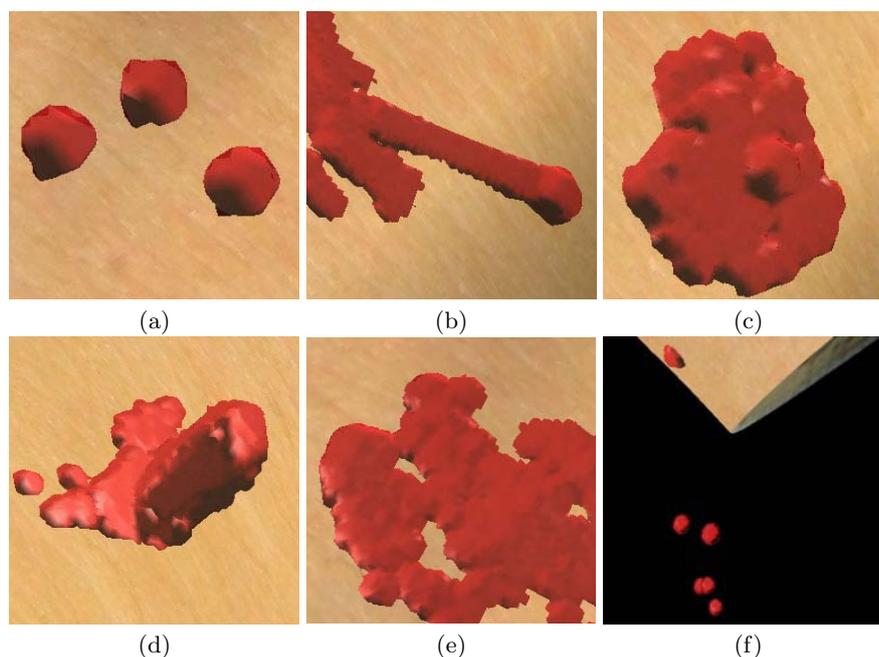


Fig. 4. The visualization results of various bleeding phenomena with collision between blood and human skin: (a) trickling, (b) flowing, (c) pooling, (d) gushing, (e) pouring, and (f) dropping.

References

1. Ztonyi, J., Paget, R., Szkely, G., Bajka, M.: Real-time synthesis of bleeding for virtual hysteroscopy. In: Proceedings of Medical Image Computing and Computer-Assisted Intervention MICCAI 2003. Volume 1. (November 2003) 67–74
2. Castro, M.A., Putman, C.M., Cebal, J.R.: Computational fluid dynamics modeling of intracranial aneurysms: effects of parent artery segmentation on intraaneurysmal hemodynamics. *American Journal of Neuroradiology* (27) (2006) 1703–1709
3. Bar-Joseph, Z., El-Yaniv, R., Lischinski, D., Werman, M.: Texture mixing and texture movie synthesis using statistical learning. *IEEE Transactions on Visualization and Computer Graphics* 7(2) (2001) 120–135
4. Quarteroni, A., Tuveri, M., Veneziani, A.: Computational vascular fluid dynamics: problems, models and methods. *Computing and Visualization in Science* 2(4) (2000) 163–197
5. Kurtcuoglu, V., Poulidakos, D.: Principles and challenges of computational fluid dynamics in medicine. In: 9th MICCAI Workshop on Computational Biomechanics For Medicine. (October 2006) 14–22
6. Svensson, J., Gårdhagen, R., Heiberg, E., Ebbers, T., Loyd, D., Länne, T., Karlsson, M.: Feasibility of patient specific aortic blood flow cfd simulation. In: Proceedings of Medical Image Computing and Computer-Assisted Intervention - MICCAI 2006 Proceedings, Part I. (2006) 257–263

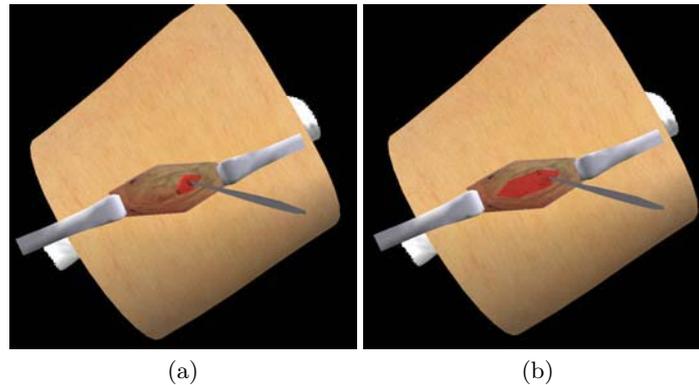


Fig. 5. Integrating our hardware-accelerated bleeding framework to a virtual-reality based Orthopedics surgery trainer.(a)accidental cut at blood vessel (b)bleeding from blood vessel.

7. Carlson, M., Mucha, P.J., R.B., V.H.I., Turk, G.: Melting and flowing. In: Proceedings of the 2002 ACM SIGGRAPH Symposium on Computer Animation. (July 2002) 167–174
8. Carlson, M., Mucha, P.J., Turk, G.: Rigid fluid: animating the interplay between rigid bodies and fluid. *ACM Transaction on Graphics* **23**(3) (Aug 2004) 377–384
9. Stam, J., Fiume, E.: Depicting fire and other gaseous phenomena using diffusion processes. In: *Computer Graphics Proceedings, Annual Conference Series, 1995 (ACM SIGGRAPH '95 Proceedings)*. (1995) 129–136
10. Müller, M., Schirm, S., Teschner, M.: Interactive blood simulation for virtual surgery based on smoothed particle hydrodynamics. *Journal of Technology and Health Care* **12**(1) (2004) 25–31
11. Daenzer, S., Montgomery, K., Dillmann, R., Unterhinninghofen, R.: Real-time smoke and bleeding simulation in virtual surgery. In: *Proceedings of Medicine Meets Virtual Reality*. (2007) 94–99
12. Libersky, L.D., Petschek, A.G., Carney, T.C., Hipp, J.R., Allahdadi, F.A.: High strain lagrangian hydrodynamics: a three-dimensional sph code for dynamic material response. *Journal of Computational Physics* **109** (1993) 67–75
13. Monaghan, J.J.: Simulating free surface flows with sph. *Journal of Computational Physics* **110** (1994) 399–406
14. Müller, M., Charypar, D., Gross, M.: Particle-based fluid simulation for interactive applications. In: *in Proceedings of ACM SIGGRAPH Symposium on Computer Animation (SCA)*. (2003) 154–159
15. Fung, Y.C.: *Biomechanics: Mechanical Properties of Living Tissues*. Second edn. Springer, Berlin Germany (1993)
16. Lorensen, W.E., Cline, H.E.: Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH 87 Proceedings* **21**(4) (July 1987)
17. Goetz, F., Junklewitz, T., Domik, G.: Real-time marching cubes on the vertex shader. In: *Eurographics 2005 Short Presentation*. (2005)
18. Xie, Y., Heng, P.A., Wang, G., Wong, T.T.: Gpu-friendly marching cubes for visualizing translucent isosurfaces. In: *Proceedings of Medicine Meets Virtual Reality*. (2007) 500–503

Modeling Air-flow in the Tracheobronchial tree using Computational Fluid Dynamics

Ilhan Kaya¹, Anand P. Santhanam², Celina Imielinska³ and Jannick Rolland^{1,2}

¹ School of Computer Science and Engineering

² CREOL, College of Optics and Photonics
University of Central Florida

4000 Central Florida Blvd Orlando, FL 32816

ikaya@creol.ucf.edu

³ Department of Biomedical Informatics, Columbia University,
701 West 168th Street
New York, NY 10032

Abstract. In this paper, we present a biomechanical framework to model air-flow inside the bronchus and deformations across the tracheobronchial tree, pipeline for the simulator, theory and initial steps to realize this framework on a highly parallel graphical processing unit (GPU). We discuss the main challenges expected and encountered to date. By using computational fluid dynamics (CFD) and computational solid dynamics (CSD) principles, we propose a numerical simulation framework that includes a biomechanical model of the tracheobronchial tree to simulate air flow inside the tree, on GPU in real-time. The proposed 3D biomechanical model to simulate the air inside the lungs coupled with a deformation model of the tracheobronchial tree, expressed through fluid-structure interaction, can be used to predict the transformations of the voxels from a 4D computed tomography (4DCT) dataset. Additionally, the proposed multi-functional CFD and CSD based framework is suitable for clinical applications such as adaptive lung radiotherapy, and a regional alveolar ventilation estimation.

1 Introduction

Lung cancer and various other ongoing or chronic lung diseases, e.g. closure of airways and emphysema, if lumped together, it is the number three killer in the United States. There has been a significant amount of research in analyzing the tissues of the lung through advanced imaging modalities such as Positron Emission Tomography (PET), Computerized Tomography (CT), and Magnetic Resonance Imaging (MRI). By applying various forms of medical image processing algorithms to CT, MRI, and PET images, anatomical structures and their material properties can be explored. Tawhai et al. make use of various forms of image segmentation algorithms to generate subject-specific computational meshes of the human bronchial tree [1, 2]. Hoffman et al. used deformable registration algorithms on CT scans to estimate the regional ventilation in ovine lungs [3, 4]. In the past few years, researchers all over the

world have been able to simulate natural phenomena such as fluids, gases, fire behavior on GPUs accurately and faster than CPUs. Three different approaches to simulate fluids in the field of computational fluid dynamics exist, fixed grid-based (Euler) or particle motion (Lagrange) and a hybrid Arbitrary Lagrangian-Eulerian (ALE). Eulerian methods evaluate material properties at stationary grid points, whereas Lagrangian methods employ moving grids with the material particles to solve the governing equations [5]. Kruger et al. built up a million Lagrangian particles to simulate and visualize 3D flow fields on non-uniform grids [6]. Dynamically regenerated meshes are used in [7] to simulate fluids. Viscoelastic material simulation based upon Eulerian methods is shown in [8]. Harris et al. make use of GPU to simulate cloud dynamics whose state is governed by incompressible Navier Stokes (NS) equations [9, 10].

In this paper, we propose a simulation framework to simulate the air inside the tracheobronchial tree on GPU. As part of this framework, we propose a 3D biomechanical model which accounts for the air motion inside the tracheobronchial tree, and the deformations on the tracheobronchial tree. This work is the first of its kind to simulate elasticity theory and fluid dynamics to predict the displacements and deformations of the human tracheobronchial tree on GPU, which provides opportunity to run the simulation in real-time and on customer PCs. The framework and biomechanical model proposed might be used to account for the alveoli ventilation when it is combined with an image registration algorithm, and it can also account for transport of aerosol contained in cigarette smoking in human lung.

2 GPU Pipeline

GPUs are very powerful, highly parallel stream processing processors. Since they are specifically built for vector processing, they have a parallel computer architecture model. Their architecture is built upon single instruction multiple data (SIMD) paradigm. In SIMD, a single node dispatches the instruction to other processing nodes that compute the instruction with their own local data. SIMD computers provide superiority over instruction driven processors, CPUs, due to their parallel execution of the instruction.

In Fig. 1 there is an illustration of floating point computational power of GPUs with respect to CPUs. This superiority makes GPUs an ideal platform to solve data-driven numerical problems in parallel. In a typical GPU, which is available on almost any PC sold today in market; there are one vertex and one fragment processor. Each processor is realized through several pipelines, working in parallel. In a NVIDIA GTX8800, which we use for our framework, there are 128 vertex and fragment processors pipelines. In Fig. 2, the GPU pipeline that realizes the stream processing nature of these parallel machines is illustrated.

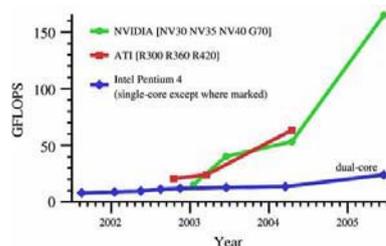


Fig. 1. Illustration of 32-bit floating point GPU comparison courtesy of Engel et al [11].

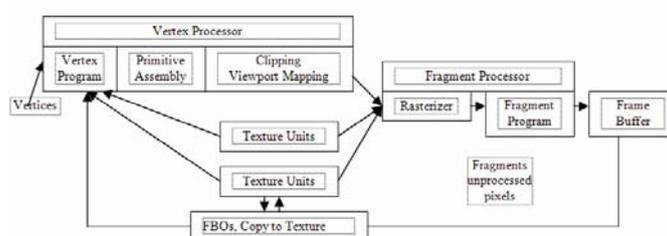


Fig. 2. Graphical Processing Pipeline, GPU Architecture

3 Biomechanical Modeling of Air inside the Tracheobronchial Tree and its Deformations

The tracheobronchial tree geometry, shape, measurements and dimensions change for each individual. Average tracheal length ranges from 5.4 cm to 13.1 cm, as the age of the subjects changes from 2 to 20 [12]. Generally, air flow through the terminal bronchioles follows a laminar pattern, whereas air flow inside the trachea can be turbulent with high ventilation rates in abnormal conditions [13]. Hence our biomechanical model consists of two parts to be able to correctly model the physics of the tracheobronchial tree: Modeling air inside the tracheobronchial tree, and modeling the deformations on the tracheobronchial tree.

3.1 Modeling air trough computational fluid dynamics (airway gas dynamics)

We model air inside the tracheobronchial tree as a fluid defined as any material which deforms continuously under shear stress. We make use of the continuum assumption of fluid mechanics, which states that fluid can be treated as an infinitely divisible substance. We divide the solution domain into 3D infinitesimal cells, where the velocity and the density of the fluid are defined as average properties. Our modeling assumptions for fluid is being incompressible, i.e. each cell has constant volume over time, homogeneous, density of the fluid stays constant in space, viscous, rate of deforma-

tion is same under the same stress for all fluid cells. Fluid cells might experience deformation, rotation, translation in space. The state of the fluid is determined through velocity of the each fluid cell, being a function of time and space, effectively represented as a vector field, and pressure of each fluid cell, which is function of space and time.

3.1.1 Equations of Fluid State

Governing equations for fluid motion are the Navier-Stokes equations (NS). The state of the fluid, given by velocity and pressure, can be determined through solutions of NS.

$$\frac{\partial \mathbf{u}}{\partial t} = -(\mathbf{u} \cdot \nabla) \mathbf{u} - \frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{F} \quad (1)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (2)$$

In the above equations, \mathbf{u} represents velocity, a vector quantity, having 3 components in 3D. If $\mathbf{x}(x,y,z)$ represents the position of the fluid cell, the velocity is $\mathbf{u}(\mathbf{x},t)$ and the pressure is $p(\mathbf{x},t)$. ρ represents the density of the fluid, ν represents the kinematics viscosity. $\mathbf{F}(\mathbf{x},t)$ represent the external forces acting on the fluid including the gravitational force. The $(\mathbf{u} \cdot \nabla) \mathbf{u}$ term is called the advective term and it is the source of nonlinearity. It resembles the propagation of any disturbance caused by the external forces acting on the fluid. The $\frac{1}{\rho} \nabla p$ term is the effect of pressure on the

rate of change of the velocity. The $\nu \nabla^2 \mathbf{u}$ term represents viscous diffusion. Due to viscosity, fluids are resistive to move, this resistance results in the distribution of momentum. Eq. 2 states the mass conservation since the fluid is assumed to be incompressible. As for the boundary conditions to solve NS, we use no-slip velocity and pure Neumann pressure condition. No-slip velocity states that velocity is zero in all 3 dimensions at the boundaries. Pure-Neumann boundary conditions state that the rate of change of pressure is zero along the normal direction on the surface boundaries.

3.1.2 A Solution Method for Equations of Fluid State

The solution of NS, the set of Partial Differential Equations (PDE), can be found in many texts. We follow the lead of Griebel et al. [14] and Marsden et al. [15] Marsden et al. show the solution of the NS through a theorem in vector calculus. This theorem is called the Helmholtz-Hodge Decomposition Theorem. It states that in the same way a vector is decomposed into two components, a vector field can be decomposed into two vector fields. A divergent vector field \mathbf{w} can be decomposed into a non-divergent vector field \mathbf{u} and a vector field composed by the gradient of the scalar field. The Helmholtz Hodge decomposition theorem is shown in Eq. 3.

$$\mathbf{w} = \mathbf{u} + \nabla p \quad (3)$$

Since the application of external forces, advection and viscous diffusion, results in a divergent velocity field, \mathbf{w} , at the end of each time step, we must satisfy the continuity requirement by subtracting the gradient of pressure field from the divergent velocity field.

$$\mathbf{u} = \mathbf{w} - \nabla p \quad (4)$$

If we apply divergence to both sides of Eq 3, then we have

$$\begin{aligned} \nabla \bullet \mathbf{w} &= \nabla \bullet \mathbf{u} + \nabla \bullet \nabla p \\ \nabla \bullet \mathbf{w} &= \nabla^2 p \end{aligned} \quad (5)$$

Since \mathbf{u} is a divergent-free vector field, $\nabla \bullet \mathbf{u}$ drops out. Eq. 5 is known as the Poisson-pressure equation. Once we have the pressure distribution, Eq. 4 gives the velocity at each time t , for any point in the fluid domain. We can further simplify Eq. 1 by defining a projection operator, Ω , which projects the divergent vector field \mathbf{w} into its non-divergent vector field, \mathbf{u} . If we apply Ω to both sides of Eq. 3,

$$\Omega(\mathbf{w}) = \Omega(\mathbf{u}) + \Omega(\nabla p) \quad (6)$$

By definition of Ω

$$\Omega(\mathbf{w}) = \Omega(\mathbf{u}) = \mathbf{u} \quad (7)$$

If we substitute Eq. 7 into Eq. 6

$$0 = \Omega(\nabla p) \quad (8)$$

Applying the projection operator Ω to Eq. 1 and substituting Eq. 8 into the resulting formulation will yield a drop out of the pressure term from Eq. 1.

$$\frac{\partial \mathbf{u}}{\partial t} = \Omega \left[-(\mathbf{u} \bullet \nabla) \mathbf{u} + \nu \nabla^2 \mathbf{u} + \mathbf{F} \right] \quad (9)$$

3.2 Modeling tracheobronchial wall dynamics using linear elasticity theory

An important characteristic of the tracheobronchial tree is that it deforms constantly under shear stress due to its elastic nature. When the fluid starts to move inside the tracheobronchial tree, its pressure distribution causes displacements among other forces acting on the tracheobronchial tree. During an inhalation expiration cycle, changes in the transmural pressure inside and outside of the tracheobronchial tree causes deformations on the walls. These deformations furthermore change the boundaries for the fluid motion, resulting in displaced boundaries for the fluid simulation. This type of coupling between fluid and an elastic wall is modeled through Fluid-Structure Interaction (FSI). Fluid pressure distribution gives the force exerted upon the elastic wall, whereas elastic displacements calculated through elastic body theory of solid mechanics, give the new boundaries for fluid. Holzhauser et al. developed a mathematical model to estimate the change in cross-sectional area of the trachea based upon a coupling between fluid flow and elastic theory [16]. FSI models

are being also used in the simulation of blood flow in highly deformable arteries. In [17], authors developed a method to couple elastic arteries wall deformation as a moving boundary condition for blood flow. Equations for the incompressible fluid flow will be the same as given in Eq. 1. and Eq. 2. However the boundaries for the simulation for each time step will be redefined with the displacement calculated through the elasticity theory. Based on a staggered partition approach, we will divide the problem domain into two steps as shown by Griebel et al. [18]. At each time step, the fluid flow will be solved as if it has fixed boundaries, and the pressure distribution calculated by the fluid flow will be an input to the elasticity problem. Then newly calculated displacements through elasticity will be fed into fluid flow as new boundaries. Fig 3 gives an overview of FSI.

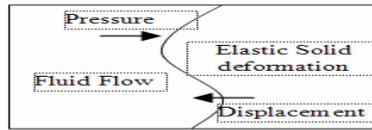


Fig. 3. A schematic of Fluid-Structure Interaction Modeling

In the elastic body deformation domain, time dependent Lamé Equation serve as the mathematical formulation of the elasticity problem. It is given as

$$\rho_s \frac{\partial^2 u}{\partial t^2} - \mu \Delta u - (\lambda + \mu) \nabla \nabla \cdot u = \rho_s b \quad (10)$$

where u represents the displacement of elastic solid, ρ_s is the density of elastic solid, b are the external forces acting on the elastic solid, μ and λ are material specific Lamé constants. It is to be noted that the b includes the force caused by the fluid flow. Finite methods have been extensively investigated for computing the solution of the elastic equation.

3.3 Biomechanical Model

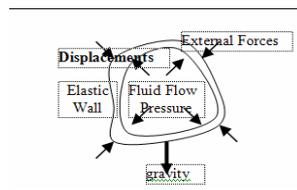


Fig. 4. 2D cross-section of a 3D biomechanical model of the human tracheobronchial tree.

In Fig. 4, a 2D cross-section of our biomechanical model for the tracheobronchial tree is shown. This model utilizes a fluid flow model of CFD with an elasticity theory of CSD in a FSI. Inputs to this model is gravitational force, external forces acting on the tracheobronchial tree, volume-time curve of the subject inhalation exhalation cycle, computational geometry of tracheobronchial tree, inhaled air parameters, i.e viscosity,

density, tissue properties for elasticity, density of the tissue, lame constants. As an output of this model, tracheobronchial deformations can be predicted as displacements.

4 Framework Pipeline

To implement the proposed framework, a simulation pipeline was developed, which is illustrated in Fig. 5. Acquisition of the geometry for the tracheobronchial tree is done through segmentation of the images obtained through CT. Fluid simulation is mostly done on GPU with an Eulerian grid approach where advection is done through a semi-Lagrangian method based on [19]. Elastic body simulation will be done through the Arbitrary Lagrangian-Eulerian (ALE) method, in process of implementation. Once we have the deformations of the tracheobronchial tree, 4DCT registration algorithm can run to estimate alveoli ventilation.

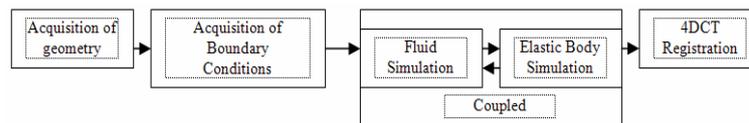


Fig. 5. Framework Pipeline to realize our 3D biomechanical model

5 Initial Implementation & Results

We have implemented initial operators which form the basis functions for the biomechanical framework proposed. The operators, addition, multiplication, gradient, divergence, and Jacobian are implemented on GPU. We use OPENGL and GLSL shading language to implement our framework. 32-bit floating point 3D textures are used to represent 3D uniform grid that we need to use for our discretization of the partial differential equation domain. As a discretization method, we use finite differences. 32-bit floating point textures are necessary to do any floating point arithmetic on GPU. In each channel of the 3D RGBA or luminance texture, the texture contains a true single precision floating point number that makes them suitable to represent 3D vector fields, i.e. velocity of fluid $w(x,t)$, $u(x,t)$, or scalar fluid pressure field, $p(x,t)$. To implement the general purpose calculation on GPU with 32-bit 3D textures, we render into 3D texture with frame buffer objects to have a feedback loop between framebuffer and texture units illustrated in Fig. 2. To compute each operation, we need to be able to generate as many fragments as the grid dimension. We render a dummy quadrilateral of filling the viewport for each slice of the 3D texture and run the computation on each fragment. Results are stored back on the 3D texture voxels through framebuffer objects. Multi-texturing is used when there is more than one operand as in the case of addition. The gradient operator takes a 3D luminance texture as input and gives a 3D RGBA texture as output, since gradient operates on a scalar field and gives the greatest rate of change on that scalar field as a vector field. The gradient of the pressure field, $p(x,t)$, is calculated through the finite difference formula given in

Eq. 11. We calculate the gradient of the pressure field to guarantee that the fluid obeys the divergent-free velocity assumption (incompressibility). Once the fluid is perturbed by an external force, the result is a divergent velocity field. This divergence on velocity field can be removed by subtracting the gradient of the pressure field at that time instant from the divergent velocity field according to the HelmHoltz Hodge theorem (Eq. 4).

$$\nabla p(\mathbf{x}, t) = \left[\frac{\partial p}{\partial x}, \frac{\partial p}{\partial y}, \frac{\partial p}{\partial z} \right] = \left[\frac{p_{i+1,j,k} - p_{i-1,j,k}}{2\delta x}, \frac{p_{i,j+1,k} - p_{i,j-1,k}}{2\delta y}, \frac{p_{i,j,k+1} - p_{i,j,k-1}}{2\delta z} \right] \quad (11)$$

The divergence operator takes a 3D velocity field, and gives a scalar field, so the input is a 3D RGBA texture, and the output is a luminance texture. Divergence of the velocity field is calculated through the following finite difference formula.

$$\nabla \cdot \mathbf{u} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = \frac{u_{i+1,j,k} - u_{i-1,j,k}}{2\delta x} + \frac{v_{i,j+1,k} - v_{i,j-1,k}}{2\delta y} + \frac{w_{i,j,k+1} - w_{i,j,k-1}}{2\delta z} \quad (12)$$

For both the gradient and the divergence operators, we sample neighboring top, bottom, front, back, left, right voxels at a position in the grid in a GLSL shader and calculate the formulas presented with uniform grid dimension assumptions. Two Poisson equations arise in the solution of fluid equations, the viscous diffusion and the Poisson pressure equations. Once the fluid has been perturbed by an external force, the result is a divergent velocity field. We can obtain the pressure distribution by solving the Poisson pressure equation given in Eq. 5. In order to account for the viscosity of the fluid, we need to solve viscous diffusion equation, given in Eq. 13.

$$\frac{\partial \mathbf{w}}{\partial t} = \nu \nabla^2 \mathbf{w} \quad (13)$$

Poisson equations can be solved by many numerical iteration methods i.e conjugate gradient, Gauss-Seidel, multi-grid, and Jacobian which are generally referred to as relaxation methods. We use the Jacobian due to its relatively easy implementation. By using the finite difference method, we expand the Laplacian operator, as it is given in Eq. 14.

$$\nabla^2 p(\mathbf{x}, t) = \left[\frac{\partial^2 p}{\partial x^2} + \frac{\partial^2 p}{\partial y^2} + \frac{\partial^2 p}{\partial z^2} \right] = \frac{p_{i+1,j,k} - 2p_{i,j,k} + p_{i-1,j,k}}{(\delta x)^2} + \frac{p_{i,j+1,k} - 2p_{i,j,k} + p_{i,j-1,k}}{(\delta y)^2} + \frac{p_{i,j,k+1} - 2p_{i,j,k} + p_{i,j,k-1}}{(\delta z)^2} \quad (14)$$

Once we substitute Eq. 14 into Eq. 5, then we obtain the iteration equation for the Jacobi algorithm, which gives us the pressure distribution across the fluid.

$$p_{i,j,k}^{n+1} = \frac{p_{i+1,j,k}^n + p_{i-1,j,k}^n + p_{i,j+1,k}^n + p_{i,j-1,k}^n + p_{i,j,k+1}^n + p_{i,j,k-1}^n - (\delta x)^2 \nabla \cdot \mathbf{w}}{6} \quad (15)$$

We implemented these operators both on GPU and CPU, and plotted the frame per second (FPS) values versus the grid size as a comparison metric in Fig. 6. As it can be seen from Fig. 6, initially, when the grid dimension is small, CPU outperforms GPU, since it is designed to do general purpose computation, and setting up the fragment

pipeline on GPU takes up time. Once the grid dimension becomes 32 or higher, GPU outperforms CPU 10-30 times, being similar to the plot given in Fig. 1.

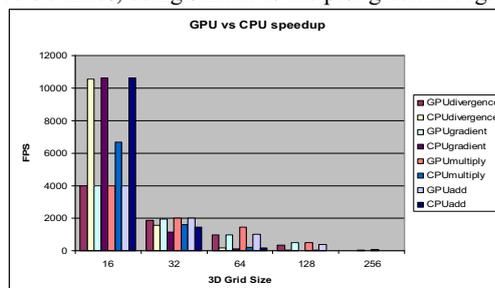


Fig. 6. GPU vs. CPU FPS comparison of the framework operators

6 Challenges

Poulikakos et al. have shown general challenges encountered while applying CFD methods to simulate fluids inside the human body, i.e. blood, cerebrospinal fluid [20]. In a similar way, we now list the challenges in implementing our biomechanical model. The complex geometry of the tracheobronchial tree makes the image acquisition and processing steps tedious. Due to the coupling of the elasticity problem with the fluid flow, fluid boundaries are modified at each time step, so traction of the boundaries is required. Hence boundary conditions are the Achilles's hill of the problem. As for the elasticity problem, material properties i.e. lame constants, density of tracheobronchial wall tissue, need to be specified. For the external forces acting on the exterior of the elastic tracheobronchial tree, an inverse computational dynamics approach needs to be specified. Numerical simulation of the problem requires discretization. For the fluid flow we use a finite-difference scheme. This is simple yet not suitable for fluid-structure problem. In order to solve Poisson Equations, Jacobi's iteration is used. Convergence and stability of numerical methods used in implementation should be verified.

7 Conclusion

In this paper we present a biomechanical framework to simulate air motion inside the tracheo-bronchial tree and estimate the deformations of the tracheobronchial tree caused by this fluid flow. This framework is being realized on GPU. The biomechanical model combines a fluid flow model based upon CFD with an elastic model of CSD. Furthermore, a simulator pipeline is given in this paper. Initial implementation results have been shown and challenges discussed.

Acknowledgements. This work is supported by US Army Medical Research and Material Command.

References

- [1] Tawhai M. H., Hunter P.,Tschirren J., Reinhardt J.,McLennan G., Hoffman E.A.:CT-based Geometry Analysis and Finite Element Models of the Human and Ovine Bronchial Tree. *J. Appl. Physiol.* Vol. 97 (2004) 2310-2321.
- [2] Tawhai M. H., Pullan A.J., Hunter P.J.: Generation of an Anatomically-Based Three Dimensional Model of the Conducting Airways. *Annals of Biomechanical Engineering.* Vol 28 (2000) 793-802.
- [3] Pan Y., Kumar D., Hoffman E.A., Christensen G.E., McLennan G., Song H.J., Ross A., Simon B.A., Reinhardt J.: Estimation of Regional Lung Expansion via 3D image Registration. *Proceedings of SPIE* Vol. 5746. Bellingham, USA (2005).
- [4] Tawhai M. H., Nash M.P., Hoffman E.A.:A Imaging Based Computational Approach to Model Ventilation Distribution and Soft-tissue Deformation in the Ovine Lung. *Academic Radiology.* Vol. 13 (2006) 113-120.
- [5] Feldman B.E., O'Brian J.F., Klingner B.M.,Goktekin G.E.: Fluids in Deforming Meshes. *Proceedings of 2005 ACM SIGGRAPH/Eurographics Symposium. on Computer Animation.* Los Angeles, USA (2005).
- [6] Kruger J., Kipfer P., Kondratieva P.,Westermann R.: A particle System for Interactive Visualization of 3D Flows. *IEEE T. on Visualization and Computer Graphics.* Vol. 11 (2005) 744-756.
- [7] Klingner B.M., Feldman B.E., Nuttapong C., O'Brian J.F.: Fluid Animation with Dynamic Meshes. *Proceedings of 2006 ACM SIGGRAPH.* Boston, USA (2006).
- [8] Goktekin G. T., Barteil W. A., O'Brain J.F.: A Method for Animating Viscoelastic Fluids. *Proceedings of 2004 ACM SIGGRAPH.* Los Angeles, USA (2004).
- [9] Harris J. M., Baxter V. W., Scheuermann T.,Lastra A.: Simulation of Cloud Dynamics on Graphics Hardware. *Proceedings of 2003 ACM SIGGRAPH / Eurographics Conference on Graphics Hardware.* San Diego, USA (2003).
- [10] Harris J.M.: Fast Fluid Dynamics Simulation on the GPU. In *GPU Gems Programming Techniques, Tips, Tricks for Real-time Graphics*, Fernando R. (Ed) Addison-Wesley Professional Boston (2004) 637-665.
- [11] Engel K., Hadwiger M., Kniss M.J., Rezk-Salama C., Weiskopf D.:Real-Time Volume Graphics. A. K. Peters Ltd, Wellesley, USA (2006).
- [12] Griscom N.T., and Wohl M.E.: Dimensions of the Growing Trachea Related to Age and Gender. *American Journal of Roentgenology.* Vol 146 (1986) 233-237.
- [13] West J.B.: *Respiratory Physiology- The Essentials* (7th Ed). Lippincott Williams & Wilkins (2005).
- [14] Griebel M., Dornseifer T., NeunHoeffer T.: *Numerical Simulation in Fluid Dynamics- A practical Introduction.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, USA (1998).
- [15] Chorin A.J., and Marsden J.E.: *A Mathematical Introduction to Fluid Mechanics* (4th Ed). Springer, (2000)
- [16] Holzhauser U. and Lambert R.K.: Analysis of Tracheal Mechanics and Applications. *J. Appl. Physiol.* Vol. 91 (2006) 290-297.
- [17] Figueroa A., Vignon-Clementel I., Jansen K.E., Hughes J.R.T., Taylor C.A.: A Coupled Momentum Method for Modeling Blood Flow in Three-Dimensional Deformable Arteries. *Computer Methods in Applied Mechanics and Engineering.* Vol. 195 (2006) 5685-5706
- [18] Engel M., and Griebel M.: Flow Simulation on Moving Boundary-fitted Grids and Application to Fluid-Structure Interaction Problems. *Int. J. Numerical Meth. In Fluids.* Vol. 50 (2006) 437-468
- [19] Stam J.:Stable Fluids. *Proceedings of 1999 ACM SIGGRAPH.* Los Angeles, USA (1999).
- [20] Paulikakos D., Kurtcuoglu V.: Principles and Challenges of Computational Fluid Dynamics in Medicine. In 9th International MICCAI Conference, Copenhagen, Denmark (2006).

Imaging Facial Signs of Neuro-Physiological Responses

Dvijesh Shastri¹, Arcangelo Merla², Panagiotis Tsiamyrtzis³, Ioannis Pavlidis^{1*}

Computational Physiology Lab¹,
Department of Computer Science,
University of Houston, Houston, TX, USA,
djshastr@mail.uh.edu, ipavlidis@uh.edu

ITAB - Institute of Advanced Biomedical Technologies²,
Department of Clinical Sciences and Bioimaging,
University "G. D'Annunzio",
Chieti, Italy
a.merla@itab.unich.it

Department of Statistics³,
Athens University of Economics and Business
Athens, Greece
pt@aueb.gr

Abstract. In the present paper, we introduce an integrated framework for detecting peripheral sympathetic responses through purely imaging means. The measurements are performed on three facial areas of sympathetic importance, that is, periorbital, supraorbital, and maxillary. To the best of our knowledge, this is the first time that the sympathetic importance of the maxillary area is analyzed. Because the imaging measurements are thermal in nature and are composed of multiple components of variable frequency (i.e., blood flow, sweat gland activation, and breathing), we chose wavelets as the image analysis framework. The image analysis is grounded on GSR signals, which are still considered the golden standard in peripheral neurophysiological and psychophysiological studies. The experimental results show that monitoring of the facial channels yields similar detecting power to GSR's.

1 Introduction

The Autonomic Nervous System (ANS) and particularly its sympathetic division has been the object of intense study in neurophysiology and psychophysiology. The sympathetic division readies the body for a crisis that may require sudden, intense physical activity. It is a primal survival mechanism. Therefore, interest on methodologies that scrutinize sympathetic responses is well founded and has many applications.

* To whom all correspondence should be addressed

2

When sympathetic activation occurs, an individual experiences increased activity in the cardiovascular and respiratory centers of the pons and medulla oblongata, leading to elevations in blood pressure, heart rate, breathing rate, and depth of respiration. These vital sign changes are mediated through adrenergic postganglionic fibers. Determination of sympathetic activation through vital sign monitoring is not always straightforward.

As an alternative, researchers focused their efforts on sympathetic manifestations effected through cholinergic postganglionic fibers. These fibers innervate sweat glands of the skin and the blood vessels to skeletal muscles and the brain. They provide a pathway to stimulating sweat gland secretion and selectively enhancing blood flow to muscles.

In this context, Electro-Dermal Activity (EDA) has been the gold standard for peripheral monitoring of sympathetic responses. EDA is measured through the Galvanic Skin Response (GSR), which is a simple and reproducible method for quantifying sweat gland activation in the palm. Alternatively, EDA can be captured through a palm thermistor, which registers the full thermoregulatory phenomenon including changes both in blood flow and sweat gland activation. In our case, this is a useful yardstick, as it provides palm information similar to the one thermal imaging provides for the face.

Indeed, in recent years, we have demonstrated that during arousal additional physiological signs materialize on the face. Specifically, we have shown that increased blood flow in the periorbital [1][2] and supraorbital [3] areas are ubiquitous manifestations of stress. We have also developed a thermal imaging methodology to extract both the periorbital and supraorbital signals.

In the present paper, we link traditional probe-based with the newer image-based neurophysiological methodologies. We study comparatively the periorbital, supraorbital, and palm channels (GSR and thermistor) within a classic repeated arousal experiment. First, we introduce a novel modeling methodology to quantify the GSR signal and validate the arousal experiment. In addition, we demonstrate that concomitantly to the palm area, strong sweat gland activation is manifested in the maxillary area. This is one more sympathetic thermoregulatory phenomenon manifested on the face. Therefore, it can be sensed and computed through thermal imaging. We apply a wavelets analysis method for all channels (periorbital, supraorbital, maxillary, GSR, and palm thermistor). The results reveal tonic (baseline) and phasic (event related) affinity of the three imaging channels to the GSR and palm thermistor channels. We also identify the presence of a breathing component in the maxillary signal.

Our research brings to the fore the pivotal role of facial physiology in the manifestation of stress and establishes the methodological framework for sensing peripheral sympathetic responses through imaging means.

In the rest of the paper we unveil our new imaging methodology for modeling and analyzing the facial sympathetic channels. Specifically, in section 2, we describe the method itself. In section 3, we report and discuss its experimental validation.

2 Methodology for Signal Modeling and Analysis

2.1 Modeling of GSR Signal

The first goal is to model the GSR signal and be able to draw inferences about the repeated arousal effect on each subject. This is very important, as GSR is the ground-truth in our studies and validation of ground-truth is a prerequisite for meaningful comparative studies. Specifically, our modeling scheme needs to show that individuals tend to habituate and therefore, GSR amplitudes tend to reduce, latencies tend to increase, and wave-shapes tend to remain unaltered. These well-established and understood patterns of repeated arousals in normal subjects, if quantified here, they will validate our experimental design and execution.

As we will discuss in the Experimentation Section, we stimulate the subjects with 3 auditory startles spaced at least 1 min apart. For this reason, we choose to split the GSR signal in three non-overlapping segments:

- *S1*: 2 sec before first startle until 2 sec before second startle
- *S2*: End of *S1* to 2 sec before third startle
- *S3*: End of *S2* to end of experiment

We divide each of the segments *S1*, *S2*, and *S3* into three subsegments:

- *LS*: The Left Subsegment, which spans from the beginning of the segment till the maximum value (shortly after startle).
- *RS*: The Right Subsegment, which spans from the maximum value till the end of the segment.
- *LSOS*: The Left Stimulus Onset Subsegment, which starts at the time of the startle and lasts until the maximum value is reached. It is portion of *LS* and is useful in estimating the habituation effect.

The GSR signal around the stimulus is formed by the charging and discharging of an RC circuit, which closes on the palm skin during emotional sweat gland activation. Charging corresponds to arousal (*LS*) and it is characterized by an exponential increase. Discharging corresponds to arousal waning (*RS*) and it follows an exponential decay. For this reason, we choose the Laplace distribution to model the GSR signal. The probability density function is given by:

$$f(t|\mu, \beta) = \frac{1}{2\beta} \exp\left(-\frac{|t - \mu|}{\beta}\right), \quad (1)$$

where μ (mean) denotes the time parameter, while $\beta > 0$ is the scale parameter.

Although, the GSR signal is not symmetric around the local maximum value, the Laplace distribution is. This led us to model separately *LS* and *RS* for every segment (see Fig. 1). For *LS* we fit a truncated Laplace distribution where the μ parameter is assumed to be known (location of the maximum) and the distribution is censored to the right of the maximum. Similarly, for *RS* we use a truncated Laplace distribution where the values at the left of the maximum are censored. The goal then is to estimate the scale parameters of the left and right

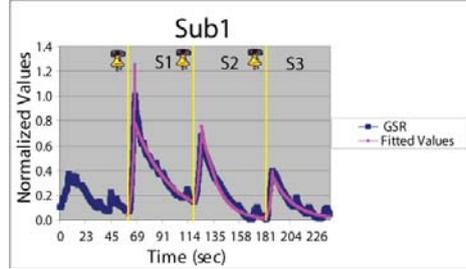


Fig. 1. GSR segments $S1$, $S2$, and $S3$ along with the fitted Laplace values for subject Sub1. The stimuli occurrences have been marked appropriately.

distributions (i.e., β_L and β_R). This estimation is done through the Ordinary Least Squares (OLS) method.

For LS where $t \leq \mu$, we have:

$$y = f(t) = \frac{1}{2\beta_L} \exp\left(-\frac{\mu - t}{\beta_L}\right) \Rightarrow \ln(y) = \left[-\frac{\mu}{\beta_L} - \ln(2\beta_L)\right] + \frac{1}{\beta_L}t, \quad (2)$$

so that time t and logarithmic scale $\ln(y)$ are linearly related. We use OLS to estimate the slope, whose inverse is the parameter of interest β_L .

For RS where $t \geq \mu$, we have:

$$y = f(t) = \frac{1}{2\beta_R} \exp\left(-\frac{t - \mu}{\beta_R}\right) \Rightarrow \ln(y) = \left[\frac{\mu}{\beta_R} - \ln(2\beta_R)\right] - \frac{1}{\beta_R}t, \quad (3)$$

so that time t and logarithmic scale $\ln(y)$ are linearly related. We use OLS to estimate the slope, whose negative inverse is the parameter of interest β_R .

For $LSOS$ we apply linear (versus exponential) fitting, as these subsegments are nearly impulsive.

2.2 Wavelets Analysis of Sympathetic Signals

We extract thermal signals from three facial areas: periorbital, supraorbital, and maxillary. In all three cases the regions of interest are tracked using the coalitional tracking method we reported in [4]. In the periorbital area, the extracted signal is formed from the evolution of the mean thermal footprint of the facial arteriovenous complex. This footprint is segmented via a fuzzy segmentation algorithm, which is seeded in the initial frame with two points in the inner orbital areas (see Fig. 2). On each subsequent frame, the seeds are adjusted with help from the coalitional tracker. In the supraorbital area, the extracted signal is formed from the evolution of the mean thermal footprint of the entire region of interest. In the maxillary area, the extracted signal is formed from the evolution of the mean thermal footprint of the entire region of interest.

The periorbital thermal signal is a correlate of the blood supply to the orbital muscle. The supraorbital thermal signal is a correlate of the blood supply to

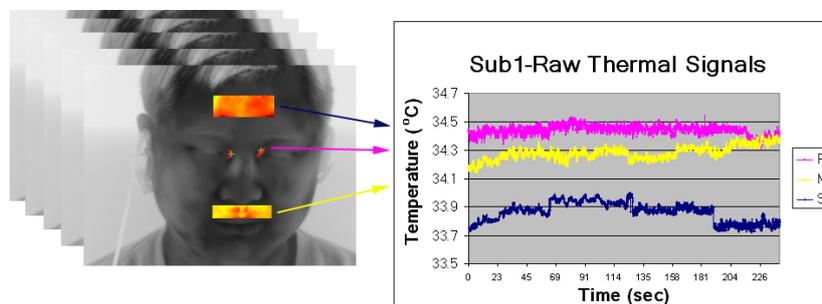


Fig. 2. Periorbital, supraorbital, and maxillary regions of interest and the respective mean thermal signals along the timeline. The periorbital measurement is strictly localized on the thermal footprints of the facial artery.

the corrugator muscle. It may also be slightly modulated from the activation of sweat glands in the forehead. The maxillary thermal signal is a correlate of the blood perfusion in the respective area. Based on our observations, it is also heavily modulated from the activation of local sweat glands. Finally, the maxillary signal is periodically modulated from the thermal effect of breathing, due to the proximity of nostrils.

Concomitantly with the three facial imaging signals, we extract palm perspiratory and thermal signals through GSR and thermistor sensors respectively. We also extract the breathing signal through a piezo-respiratory belt transducer. All probe signals (GSR, palm thermistor, and respiratory belt) are synchronized with the thermal imager through an electronic circuit.

The stress content of the GSR signal has been documented in the literature exhaustively [5][6][7]. To associate this content to the facial imaging signals, we use a multi-resolution wavelets approach. The typically noisy profile of facial signals (see Fig. 2) and the confounding phenomena that form them, do not allow direct modeling of their raw waveforms, as in the case of GSR (see Fig. 1). Therefore, component isolation and noise reduction are necessary.

Specifically, we consider that all signals if they are of sympathetic importance they have either a strong phasic or tonic component [8]. The phasic component should be at a scale that matches the inter-stimulus interval of the experiment, while the tonic component will reside at an even higher scale that spans the entire experimental time-line. Any strong extraneous modulation (e.g., breathing) in some signals should be evident in a lower scale (i.e., higher frequency), far away from the phasic and tonic scales.

To quantify the contribution of phasic, tonic, and other components in the signals we apply a Continuous Wavelet Transform (CWT) with a Daubechies-10 mother wavelet. We then compute the energy of each signal in all scales. The energy curves feature global and local maxima. We analyze these maxima to understand if they correspond to phasic or tonic responses. We also compare their relative contributions in each signal.

3 Experimentation and Discussion

We used a high quality Thermal Imaging (TI) system for data collection. The centerpiece of the TI system is a ThermoVision SC6000 Mid-Wave Infrared (MWIR) camera [9] (NEDT=0.025°C). We recorded 10 thermal clips from the faces of 10 subjects while resting in an armchair. Concomitantly, we recorded ground-truth GSR, palm thermistor, and piezo-respiratory signals with the PowerLab 8/30, ML870 data acquisition system [10]. The data set features subjects of both genders, different races, and with varying physical characteristics. The subjects were focused on a mental task while they were measured through the thermal imaging and contact sensors. The experiment lasted 4 min. After the first minute the first auditory startle was delivered and after that two more were delivered spaced at least 1 min apart. The experiment ended about 1 min after the delivery of the third startle.

3.1 GSR Results

We applied the modeling methodology detailed in Section 2.1 to each segment of every GSR waveform. Therefore, we had 3 segments ($S1, S2, S3$) \times 3 sub-segments ($LS, RS, LSOS$) \times 10 subjects = 90 cases for which we needed to estimate the scale parameter β (Laplace fitting for LS and RS) or the slope (linear fitting for $LSOS$). The results are shown in Table 1 and elicit the following conclusions:

- For all stimuli ($S1, S2$ and $S3$) LS has a much smaller scale parameter than RS indicating that the phenomenon causes a steep increase and then decays at a much lower rate.
- Comparing the LS parts of $S1, S2$, and $S3$, within the same subject, we observe that usually the 1st stimulus causes the steepest increase and as we move to subsequent stimuli the response is less steep (i.e., the β_L parameter is increasing).
- Comparing the RS parts of $S1, S2$, and $S3$, within the same subject, we observe that usually the subject recovers slowly after the 1st stimulus (i.e., it has a high β_R parameter). Recovery from subsequent stimuli is becoming faster (smaller β_R parameter).
- Comparing the $LSOS$ parts of $S1, S2$, and $S3$, within the same subject, we observe that the estimated (positive) slope of the linear regression is decreasing as we move from $S1$ to $S2$ to $S3$ (habituation).

These conclusions are in accordance with the expected behavior of normal subjects, and therefore our experiment is valid.

3.2 Comparative Wavelets Analysis Results

We applied the wavelets analysis methodology detailed in Section 2.2 for all 6 sympathetic signals from all 10 subjects. Fig. 3 shows the wavelet energy curves

	Sub1	Sub2	Sub3	Sub4	Sub5	Sub6	Sub7	Sub8	Sub9	Sub10
<i>S1-LS</i>	2.18	2.68	1.70	1.94	7.50	14.54	4.79	25.39	4.50	3.87
<i>S1-LSOS</i>	0.20	0.10	0.15	0.14	0.13	0.05	0.06	0.05	0.08	0.12
<i>S1-RS</i>	30.40	33.90	27.11	40.89	36.88	82.50	151.47	38.76	68.08	25.24
<i>S2-LS</i>	3.75	3.04	1.77	8.49	5.69	26.16	7.89	13.13	8.29	9.37
<i>S2-LSOS</i>	0.11	0.07	0.13	0.08	0.12	0.03	0.10	0.04	0.10	0.11
<i>S2-RS</i>	13.39	29.46	34.11	62.67	22.46	53.43	44.81	20.71	61.65	30.69
<i>S3-LS</i>	1.91	2.51	2.30	10.13	2.51	62.46	15.09	2.28	10.63	5.49
<i>S3-LSOS</i>	0.09	0.09	0.12	0.06	0.12	0.01	0.03	0.08	0.07	0.08
<i>S3-RS</i>	17.93	11.36	34.25	66.67	19.12	31.53	81.62	108.44	25.21	13.44

Table 1. The estimated β parameters for the *LS* and *RS* Laplace distributions along with the linear regression slope estimates of *LSOS*.

in lower and higher scales of subject Sub1. In lower scales (i.e., 50-250) the piezo-respiratory signal (Brt) appears to have a dominant component, as it is manifested by the high bell-shaped bulge. This is in accordance with its expected function. The second most prominent component is featured by the maxillary signal (M). This verifies our hypothesis of breathing modulation for this signal, as it is sampled in proximity to the nostrils.

In higher scales, (i.e., 1000-3000) the GSR signal (GSR) appears to have a dominant component, as it is manifested by the high bell-shaped bulge. This is the phasic component as the scale is about 1/3 of the total scale and matches the period of the repeated stimuli in our experiment. The strong presence of a phasic component in the GSR signal is consistent with its nature. The fascinating result here is the almost equally strong phasic component in the maxillary signal (M). This is consistent with our hypothesis of strong sweat gland activation in the maxillary area concomitant to the palm area. Other facial signals (i.e., periorbital-P and supraorbital-S) also have significant but relatively weaker phasic components, which verifies their sympathetic relevance.

At the highest scales that span almost the entire timeline resides the tonic component of the signals. It is worth noting that the GSR signal has the smallest tonic component of all sympathetic channels. This is consistent with the almost unimodal nature of the GSR channel. The maxillary signal (M), which is its facial equivalent, has a much stronger tonic component. In contrast to the GSR signal, the maxillary signal contains not only local sweat gland activation information, but also thermal information related to changes in local blood perfusion. In this sense, the maxillary signal (M) is probably closer to the palm thermistor (Thr) signal.

In general, adrenergic and cholinergic signal components reside in nonoverlapping scales, which makes the adopted multi-resolution approach an ideal analysis tool. The picture emerging from the analysis of the wavelet energy curves for subject Sub1 remains relevant for all the other 9 subjects in our dataset. Fig. 4 shows the mean energy of tonic, phasic, and breathing components of the various sympathetic channels for the entire data set. All the conclusions ex-

8

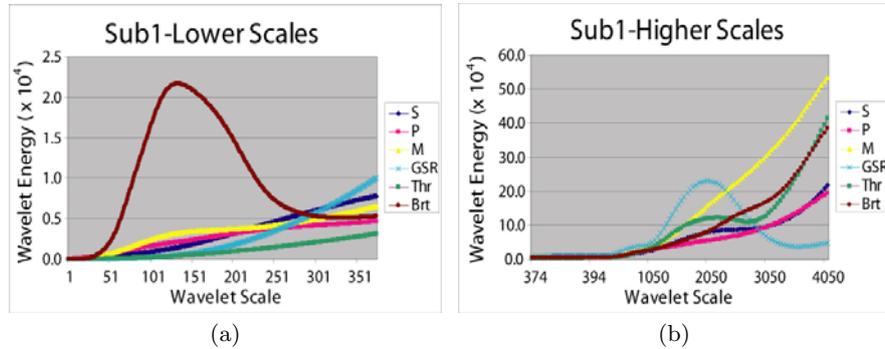


Fig. 3. Wavelet energy curves of subject Sub1 for all 6 sympathetic channels in (a) lower and (b) higher scales.

tracted through the example of subject Sub1 still apply for the thus statistically constructed mean subject.

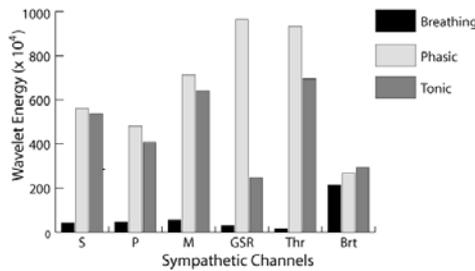


Fig. 4. Mean tonic, phasic, and breathing energy components for the various sympathetic channels.

References

- [1] Levine, J., Pavlidis, I., Cooper, M.: The face of fear. *The Lancet* **357**(9270) (2001) 1757
- [2] Pavlidis, I., Eberhardt, N., Levine, J.: Human behavior: Seeing through the face of deception. *Nature* **415**(6867) (2002) 35
- [3] Puri, C., Olson, L., Pavlidis, I., Starren, J.: Stresscam: Non-contact measurement of users' emotional states through thermal imaging. In: *Proceedings of the 2005 ACM Conference on Human Factors in Computing Systems (CHI)*, Portland, Oregon (2005) 1725–8
- [4] Dowdall, J., Pavlidis, I., Tsiamyrtzis, P.: Coalitional tracking. *Computer Vision and Image Understanding* (2007)

- [5] Baba, M., Watahiki, Y., Matsunaga, M., Takebe, K.: Sympathetic skin response in healthy man. *Electromyography Clinincal Neurophysiology* **28** (1988) 277–283
- [6] Uncini, A., Pullman, S., Lovelace, R., Gambi, D.: The sympathetic skin response: Normal values, elucidation of afferent components and application limits. *The Journal of Neuroscience* **87** (1988) 299–306
- [7] Elie, B., Guiheneuc, P.: Sympathetic skin response: Normal results in different experimental conditions. *Electroencephalogy and Clinical Neurophysiology* **76**(258-267) (1990)
- [8] Lim, C., Rennie, C., Barry, R., Bahramali, H., Lazzaro, I., Manor, B., Gordon, E.: Decomposing skin conductance into tonic and phasic components. *International Journal of Psychophysiology* **25** (1997) 97–109
- [9] FLIR Systems 70 Castilian Dr., Goleta, California 93117: (<http://www.flir.com>)
- [10] ADInstruments 2205 Executive Circle, Colorado Springs, Colorado 80906: (<http://www.adinstruments.com>)

A novel partnership between surgeons and machines, made possible by advances in computing and engineering technology, could overcome many of the limitations of traditional surgery. By extending the surgeons' ability to plan and carry out surgical interventions more accurately and with less trauma, Computer-Integrated Surgery (CIS) systems could help to improve clinical outcomes and the efficiency of health care delivery. CIS systems could have a similar impact on surgery to that long since realized in Computer-Integrated Manufacturing (CIM). Mathematical modeling and computer simulation have proved tremendously successful in engineering. Computational mechanics has enabled technological developments in virtually every area of our lives. One of the greatest challenges for mechanists is to extend the success of computational mechanics to fields outside traditional engineering, in particular to biology, biomedical sciences, and medicine. The workshop provided an opportunity for computational biomechanics specialists to present and exchange opinions on the opportunities of applying their techniques to computer-integrated medicine.