Journal of Mobile Multimedia, Vol. 2, No.1 (2006) 052-080

NON-INVASIVE METHOD FOR PATIENT-SPECIFIC VIRTUAL HEART BASED ON FIBER-FLUID MODEL

S. Y. TAN, SELVANATHAN NARAINASAMY, SOMASUNDARAM NAGAPPAN

University of Malaya, Kuala Lumpur, Malaysia
selvanathan@um.edu.my

Received 29 December 2005
Revised March 1 2006

In this paper, we present a non-invasive methodology in constructing a patient-specific virtual human heart based on fiber-fluid model. We applied digital image processing techniques on patient-specific MR images for obtaining the geometry information of human heart. The techniques include: acquisition, image visualization, image enhancement and segmentation. We incorporate cubic Hermite basis functions in our epicardium surface interpolation algorithm. We formularized a three-dimensional rule based fiber reconstruction mechanism to reconstruct the cardiac fiber architecture. A fiber model has been constructed which consists of 1,038 fibers with 371,239 fiber points. This model describes the ventricular three-dimensional geometry. Immersed Boundary Methods is used in constructing fiber-fluid cardiac simulation. The simulation of early ejection (from 0ms to 0.5ms) for the Left Ventricle (LV) has been implemented in SGI workstation. Simulation results for cardiac fiber and blood flow are presented in three-dimensional (3-D). Open GL-based animated visualization programs are developed to serve three purposes: (1) to demonstrate the interpolation and rule-based fiber reconstruction process. (2) To visualize the simulation results of cardiac fiber and blood flow; (3) To analyse the dynamics of the epicardium fibers as well as the blood flow in the cavity of the LV.

Key words: Virtual Heart, Heart Modelling, Fiber Fluid, Immersed Boundary, Rule based fiber reconstruction, Myocardial fiber orientation
Communicated by: D Taniar

1 Introduction

1.1 Overview

Heart disease is one of the leading causes of fatality in the medical world. By being able to realistically model a human heart, it is hoped that the medical treatment and management of heart diseases can be improved further, thus causing a significant reduction in the loss of human lives. By creating a model, analysis [34] and data mining [36] can be carried out to diagnose and profile heart related diseases. Also, these models can be used to create ontologies [35] for future medical references.

Most of the existing human heart models simulated in a computer do not consider the effect caused by the blood flow in the heart chamber towards the cardiac fiber during systole and diastole. This is because most of the engineering problems predefine the motion of the boundary when applying the Computational Fluid Dynamic (CFD) method. Whereas the heart is a biological fluid dynamic problem where its boundaries are not rigid and its motion is a result of forces imposed on it by the motion of the surrounding fluid [17]. Furthermore, the boundaries of the heart model, which consists of fibers, contribute their own forces when they contract, which exerts into the blood and causes an overall effect on subsequent fiber motions. The motions of the fluid (oxygen enriched blood entering

the LV), and the motion of the LV boundaries, form a coupled system; both must be computed simultaneously and repeatedly, which makes fiber-fluid heart model difficult to build.

The heart models built by Peskin and McQueen are mainly concentrated on observing the blood flow in the heart [5,16,17,24,23]. The fiber architecture of their heart model is constructed based on the characteristics of the fiber trajectories stated in [29]. In our work, we constructed the heart model based on a human heart scanned using MRI. The two dimensional (2-D) MR images are suitable to quantify the shape of the heart accurately in term of geometrical information. Furthermore, by applying suitable 3-D image processing techniques (e.g. interpolation) on multiple 2D images, a realistic 3D cardiac geometry information can be obtained.

Modelling and visualization of human organs and other physical expressions are common. For example, in [32] a simulation of human facial expressions and talking animation has been modelled successfully and in [33] the authors have managed to visualize pain data. However, to the authors' knowledge, a combination of rule based fiber reconstruction algorithm with Immersed Boundary Method in constructing a fiber-fluid human heart model has not been thoroughly investigated. In this paper, we formularized a three-dimensional rule based fiber reconstruction mechanism to reconstruct the cardiac fiber architecture. Our fiber model consists of the fiber architecture information that is obtained from MR images. Based on this data, we incorporate the Immersed Boundary Method to construct a fiber-fluid cardiac simulation. A fiber model has been constructed which consists of 1,038 fibers with 371,239 fiber points. This model describes the ventricular three-dimensional geometry. This fiber-fluid model allows us to study the dynamic of the cardiac muscle as well as the blood flow in the heart chamber while, realistically simulating cardiac contraction. There has been active research being done to discover and study the laminar arrangement of groups of myocytes. According to [10] and [27], the three-dimensional arrangement of ventricular myocytes influences the electro-physiological and elasto-mechanical properties of the heart. Hence, this fact cannot be neglected and must be taken into account in order to obtain realistic physiological heart models. It must be taken into account during the construction of computer simulated heart model.

Visualization is an important concept to allow simulated results to be displayed on a computer screen. There is no significant meaning if the results are represented in a numerical form. Computer graphic technology plays an important role in this area. It is required for translating numerical results into complex three-dimensional objects in real time. Interactive computer graphic is also required during the design of the model and analysis of the results. Through it, objects can be animated, turned on and off, displayed in any point of view and magnify into any region of interest as stated in [17]. However, the concept of visualization is not just limited to visualization of the simulation results but it is also can be used to visualize the simulation processes themselves. In other words, computational processes can be displayed on a computer screen during their execution. Thus, this will help us to verify and understand the calculation when the simulation is running.

1.2 Rule based Cardiac Fiber Orientation (Myocardial Fiber)

The concept of applying rules-based cardiac fiber assignment on myocardial sheet is not new and has been proposed in [27] and [26]. Based on the general geometry information extracted from MR images, we introduce a new rule-based 3D fiber orientation reconstruction technique. A single fiber consists of a set of fiber points. These points are the epicardium geometry points extracted from the MR images. The fiber sheet is constructed from these points by registering them into a fiber set. The selection mechanism of a registered material point will influence the constructed fiber sheet architecture. The mechanism is restricted tightly by a set of pre-defined rules that has been constructed

carefully based on the formation of the fiber orientation. The rules take into consideration the fiber orientation angle, direction of the fiber path, overlap between two or more fibers and duplication of registered fiber points. The details are elaborated in section 2.4.

1.3 Immersed Boundary Method

This method had been proposed and developed by Charles Peskin and David McQueen [15,21]. It is a general numerical method which is able to efficiently simulate a system that is represented by elastic fibers immersed in an incompressible fluid.

In this method, the fibers are considered as elastic material. It is treated as a part of the fluid [13,14]. A fiber is listed as fiber points, where all the fibers will form the boundaries of the heart wall and the fluid surrounding the boundaries are represented by a rectangular cubic lattice. The boundaries are free to move continuously in the computational fluid lattice. The blood can be treated as a Newtonian fluid where its dynamics are able to obtain by numerical solution of the Navier-Stokes (NS) equations.

At every time step the fiber point obtains a force value (resulting from heart contraction). These forces are called Immersed Boundary elastic forces. The fiber points exert force onto the fluid lattice as a sum of Dirac Delta functions of fiber forces evaluated at the lattice points. The velocity of each cell of the fluid lattice is then calculated from these local forces using the NS equation. Once again, via Dirac delta function, the velocities of the fiber points are updated. Each of the fiber point is then moved to new location based on its velocity. In the next time step, based on the new position of the fiber, the forces are recalculated and the entire operation is repeated. Figure 1 summarizes the 4 major steps in Immersed Boundary Method:

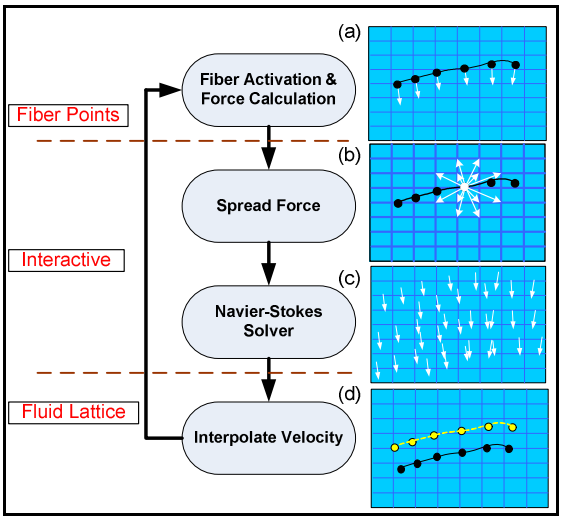


Figure 1: The 4 major tasks in the Immersed Boundary Method in each time step. (a) The forces (white) are generated from each fiber points (black). (b) The fiber points carry forces (white arrow) to be exerted onto the fluid lattice. (c) The fluid lattice in whole simulation space is solved via NS equation in order to obtain the next time step velocities on each fluid grid. (d) The fiber points are moved to new location from their old position (yellow) after updating their velocities.

The virtual heart model being built leverages on two important research directions. First, the heart wall mechanics analysis and secondly, fluid mechanics. These mechanics are both equally important to be considered in order to model a realistic human heart. Thus, the fiber-fluid model is implemented using the Immersed Boundary Method. The concept and details of this method is explained in section 2.7. Meanwhile, some studies of computational method for blood flow in heart have been done using this method and have been covered in [8,9,11,12,16,18,22,23,24].

2 Methodologies

2.1 Materials Acquisition

A set of high-resolution 2D MR images were obtained from a volunteer (27 years old healthy male) on breath-hold at the beginning of the systolic stage of LV. The MR scanner used is the Siemens Magnetom Vision MRI loaded with Numaris 3 V8 31C software. Scan parameter are:1.5 Tesla, TR of 510 ms, TE of 14 ms, α is 15 and FOV of 23cm. The volumetric images are scanned based on T1-weighted Magnetization Prepared Rapid Acquisition Gradient Echo (MPRAGE) images with the dimension of 256 x 256 x 16. The pixel spacing is 1.0mm x 1.0mm x 1.0mm resulting in isotropic voxels with average in-plane resolution of 0.1mm and average cross plane resolution of 2mm. In this work, the scanning process concentrates on the LV. Raw image data are then converted to the TIFF format using a medical imaging software. 67 images were produced for this study.

2.2 Three-Dimensional LV Contour Model

The LV contour model is the geometry information extracted from the MR images, which is described using a 3-D Cartesian coordinate system. A set of contours are obtained from the outermost layer of the LV from each MR image. The contour points are referred as epicardium material points. These points are fairly important for the rule-based fiber reconstruction process. The steps taken to obtain this model are described in the following sections.

2.3 Segmentation

As a first step, an anisotropic filter is used on the captured MR images in order to reduce image noise. Then, we tested a series of threshold values on the heart images and applied an edge detection algorithm to identify and extract the Region-of-Interest (ROI), i.e. the LV chamber, from the images. A set of LV transmural contours from these images is extracted using Active Shape Model (ASM) [3]. We labeled our images from training data set with landmark points; making it easier to extract the contour models of the LV. Accurate labeling of the training set is important for the performance of the model. Each labeled point indicates a feature on the object boundary, which can be located in each image in the training set. Let the i^{th} shape model be defined as

$$\mathbf{x}_i = (x_{i0} , y_{i0} , x_{i1} , y_{i1} , \ldots, x_{ik} , y_{ik} , \ldots, x_{in-1} , y_{in-1})^T \qquad (1)$$

where (x_{ij} , y_{ij}) is the x-y coordinate in image plane for the j^{th} landmark point of the i^{th} shape model. The i^{th} eigenvector affects point k in the model by moving it along a vector parallel to **d**, where

$$\mathbf{d} = (dx_{i0} , dy_{i0} , \ldots , dx_{ik} , dy_{ik} , \ldots , dx_{in-1} , dy_{in-1}) \qquad (2)$$

is the deviation from the mean shape, denoted as $\bar{x}$. Given that N is the total number of shape models the deviation value of every model, $\mathbf{dx}_i$ **,** i=1,2,…,N is calculated using:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad (3) \qquad \text{and} \qquad \mathbf{dx}_i = \mathbf{x}_i - \bar{x} \qquad (4)$$

Since we have obtained the deviations, $\mathbf{dx}_i$ is a ($2n$ x $2n$) covariance matrix, which is obtained via:

$$S = \frac{1}{N}\sum_{i=1}^{N} \mathbf{dx}_i \bullet \mathbf{dx}_i^T \qquad (5)$$

The models of variation for the points from the shapes are described by $\mathbf{p}_k$ where $k = 1\ldots2n$ represents the unit eigenvectors of S, so that

$$S\mathbf{p}_k = \lambda_k \mathbf{p}_k \qquad (6)$$

where λ_k is the k^{th} eigenvalue of S. Hence, any shape in the training set can be approximated using the mean shape as described in equation (7).

$$\mathrm{x} = \bar{x} + \mathbf{P} \bullet \mathbf{b} \qquad (7)$$

Note that $\mathbf{P} = (\mathbf{p}_1 \ \mathbf{p}_2 \ \ldots \ \mathbf{p}_t)$ is the matrix of the first t eigenvectors, and $\mathbf{b} = (\mathrm{b}_1 \ \mathrm{b}_2 \ \ldots \ \mathrm{b}_t)^{\mathrm{T}}$ is a vector of weights for each eigenvector.

ROI from the first image was extracted and used as training data. With a help from a medical image processing toolkit, a sequence of images are loaded and the training data is used for segmentation process. At the end, 67 sets of contours points are segmented from the MR images data set. We represent the extracted contours model as:

$$c_i = \left\{ p_{i,1}, p_{i,2}, \ldots, p_{i,k+1}, \ldots, p_{i,N_i} \right\} \qquad (8)$$

where i=1,2,…Φ refers to the contour's index and Φ is 67 (total number of contours). N_i in equation (8) is the total number of points found in i^{th} contour. By considering a specified point from i^{th} contour, $p_{i,j}=(x_{i,j},y_{i,j}) \forall j \in [1,N_i]$ where $p_{i,j}$ is a 2D Cartesian coordinate. Figure 2 shows the segmentation results of some of the selected MR images.

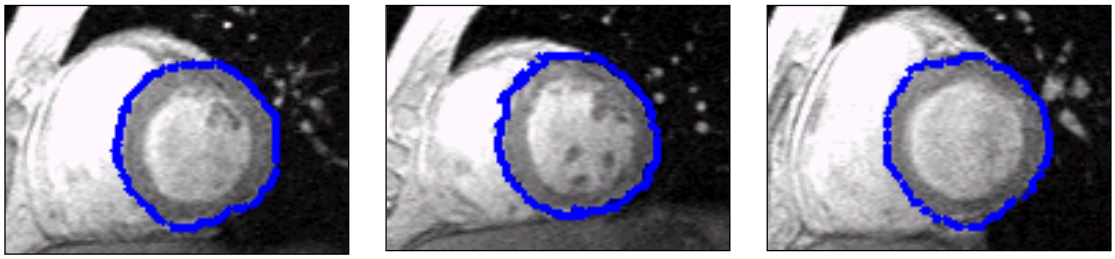


Figure 2: Selected results of epicardium segmentation results on MR images. The contour (blue) of epicardium was extracted from the ASM algorithm.

The contours from model c_i are indexed in sequence from 1 to 67. c_1 refers to the first contour at the LV base while c_{67} refers to the last contour at the LV apex. We generate another additional 3 contours to extend the LV apex because the MR scan was unable to capture enough information in this region. The new contours are generated based on the 67th contour. Equation (9) and (10) show the extrapolation mechanism used to perform this task:

$$(x_{r,j}, y_{r,j}) = (m \times (x_{67,j}, y_{67,j}) + n \times (c_x, c_y)) / (m + n) \qquad (9)$$

for $1 \leq j \leq N_{67}$ where m=1,2,3 and n=4-m. Note that the notation for the new points generated in the new contour are $np_{r,j}$, where $np_{r,j}=(x_{r,j}, y_{r,j})$. r holds the values of Φ+1, Φ+2 and Φ+3. Φ is assigned a new value of 70 after this process. (c_x, c_y) in equation (9) is the centre point of the 67th contour where

$$c_x = \left(\max_{1 \leq j \leq N_{67}} (x_{67,j}) + \min_{1 \leq j \leq N_{67}} (x_{67,j}) \right) / 2 \quad \text{and} \quad c_y = \left(\max_{1 \leq j \leq N_{67}} (y_{67,j}) + \min_{1 \leq j \leq N_{67}} (y_{67,j}) \right) / 2 \qquad (10)$$

2.2.2 3-D Contour Geometry

After stacking all the contours (see Figure 3), we scale and transform all points from ci to fit the 3D contour model into the centre of the 64 x 64 x 64 experimental space. Hence, there is a new notation to represent the previous 2D contour model as 3D model, which is given by:

$$C_i = \{P_{i,1}, P_{i,2}, ..., P_{i,k+1}, ..., P_{i,N_i}\} \qquad (11)$$

C_i contains a set of 3D coordinates belonging to the i^{th} contour, which is denoted as $P_{i,Ni}$. $P_{i,Ni} \equiv P_{i,Ni,d}$ where d=1,2,3 is used to index the dimension value of the 3D coordinate P. Figure 4 shows the constructed 3D contour model.

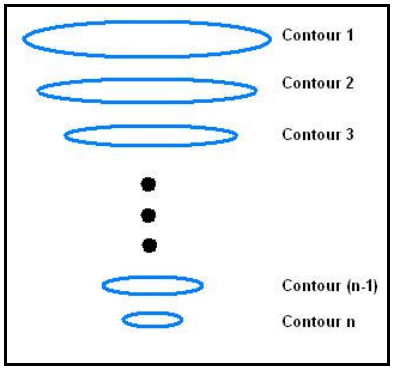


Figure 3: Schematic diagram of how the segmented contours are stacked.

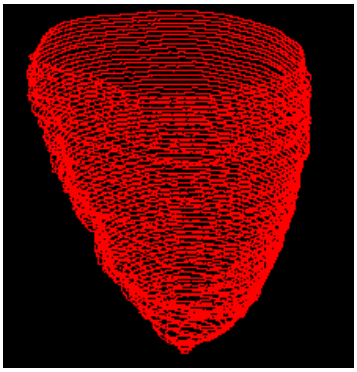


Figure 4: The 3D contour model, formed by 2D contours that will be used for the interpolation process.

2.3 Interpolation

It was observed that there are gaps between adjacent contours after stacking together in 3D space. This can be attributed to the scaling effect in the z dimension. An interpolation technique was implemented in order to create a smooth cardiac wall surface . Cubic Hermite basis function was chosen to smooth the interpolation based on key-points lying on the contours.

The Cubic Hermite function has been used before with finite element deformation to construct the ventricle geometry in [19]. This function ensures derivative continuity across element boundaries [19]. Hermite basis function (shown in Figure 5) is given as:

$$\Psi_1(s) = 1\text{-}3s^2\text{+}2s^3, \quad \Psi_2(s) = s^2(3\text{-}2s), \quad \Psi_3(s) = s(s\text{-}1)^2, \quad \Psi_4(s) = s^2(s\text{-}1). \qquad (12)$$

Lets consider a 1D field variable u. With 2 key values, u_1 and u_2, the derivatives are denoted as $(du/ds)_1$ and $(du/ds)_2$. The cubic Hermite interpolation is given by:

$$u(s) = \Psi_1(s)\, u_1 + \Psi_3(s)\, (du/ds)_1 + \Psi_2(s)\, u_2 + \Psi_4(s)\, (du/ds)_2 \qquad (13)$$

where s is $0 \leq s \leq 1$. Construction of the Hermite Cubic function is a straightforward extension of the 1D basis and for 3D coordinates, interpolation is carried out by replacing the value u with the values of x, y and z dimension for the key points. In our case, we will proceed with two interpolation mechanisms: vertical interpolation and horizontal interpolation.

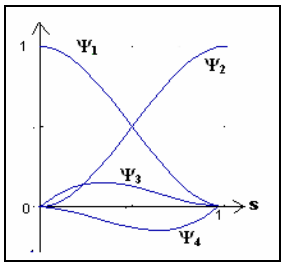


Figure 5: Graph representation of the Cubic Hermite basis Function

2.3.1 Vertical Interpolation

The key points used in vertical interpolation (VI) mechanism are identified vertically throughout all the contours (see Figure 6). The number of point in each contour decreases when the contour's index number increases. Thus, we have $N_\Phi \leq N_{\Phi-1} \leq \ldots \leq N_2 \leq N_1$. Note that $K_f = \{ U_{f,i} \}$represents the key point's set where f indicates the index of a specified set. Let F be the total number of set for vertical key point while $U_{f,i}$ represents the 3D coordinate of points in the key point set f and i is the point index in that set. The range of i and f is given as: $1 \leq i \leq \Phi$ and $1 \leq f \leq F$ respectively. The following description shows the algorithm we implemented to identify the key points set for vertical interpolation.

We let

$$U_{f,1} = P_{1,s} \tag{15}$$

for $1 \leq f \leq F$ and $1 \leq s \leq S$ where, $S=N_1$. Hence, the equation (15) is only valid when $F=S$. The P is the notation for the points from contour model, C (Equation (11)). Equation (15) only determines the first point for each of the VI key point set. The rest of the key points for each VI set will be determined by the following algorithm. We define

$$U_{f,i} = P_{i,r} \tag{16}$$

where

$$\Delta d_{\,i,\alpha} = \sqrt{\sum_{d=1}^{3} \left(P_{i,\alpha,d} - U_{f,i-1,d} \right)^2} \tag{17}$$

and

$$\Delta d_{\,i,r} = \min(\Delta d_{\,i,\alpha}) \tag{18}$$

for $1 \leq f \leq F$, $2 \leq i \leq \Phi$ and $1 \leq \alpha \leq N_i$. r holds the point index on i^{th} contour when the value of Δd is minimum. Equation (17) evaluates the distance between the last identified point (from i^{th}-1 contour), from a specified key point set with each point that lies on the i^{th} contour. The purpose of equation (18) is to obtain the minimum distant between each key point in a set.

The VI mechanism is to generate a number of points to be inserted in between each pair of consecutive points from a set of VI key point. Let a pair of consecutive control points be denoted as $(U_{f,t}, U_{f,t+1})$ where $1 \le t \le \Phi\text{-}1$ and $1 \le f \le F$ (see Figure 6). t is the point index from a specified set f. The number of interpolated points is determined by the value of λ where λ is the maximum distance allowed for each pair of consecutive control points. We define

$$G_{f,t} = \frac{\Delta d_{t+1,r}}{\lambda} \tag{19}$$

where $G \in Z^+$, $1 \le f \le F$, $1 \le t \le \Phi\text{-}1$. G holds the number of vertical interpolated points that are inserted in between the gap t, where the gap is formed between points $U_{f,t}$ and $U_{f,t+1}$.

A new notation, $V_{f,t}$ is given to represent all the new vertically interpolated points (Equation (20)). f is the index for key point sets and t is the gap's index correspond to a specified f.

$$V_{f,t} = \{T_{f,t,1}, T_{f,t,2}, \ldots, T_{f,t,\gamma} \ldots, T_{f,t,\beta_{f,t}}\} \tag{20}$$

The 3D coordinates T in V for $1 \le f \le F$ are arranged based on ascending values of the z-dimension:

$$T_{f,t,1,z} \le T_{f,t,2,z} \le \ldots \le T_{f,t,\gamma,z} \le \ldots \le T_{f,t,\beta_f,z} \tag{21}$$

for z=3. The points are sorted in ascending order according to their z dimension value for the usage of horizontal interpolation described in Section 2.3.2.

$$\beta_{f,t} = G_{f,t} + 2 \tag{22}$$

Note that $\beta_{f,t}$ (used in equation (20) and (21)) is the number of total points in a specified vertical interpolated set. The value held by β is added with 2 (in Equation (22)), as the VI key points are also considered as interpolated points and stored in model V.

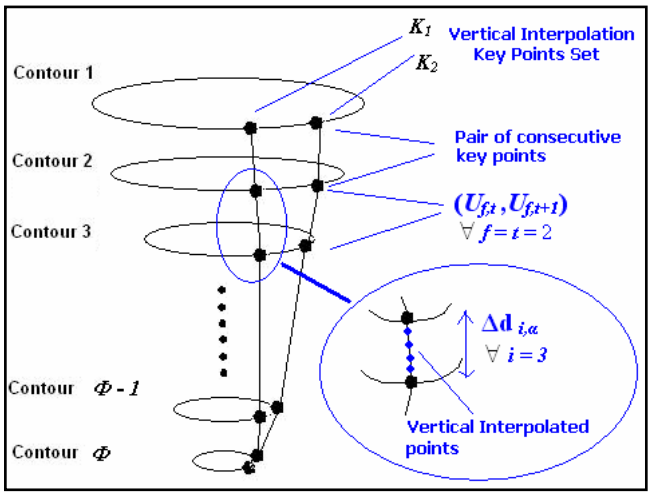


Figure 6: Shows the relationship between VI key points set, the key points itself and the contours. Diagram also depicts the interpolated points' location and distance between 2 consecutive points in a key point set. There are 4 vertically interpolated points inserted into the second gap for f=1.

2.3.2 Horizontal Interpolation

The horizontal interpolation control points are identified from the vertical interpolated points (see Figure 7). Given that

$$N_t = \max(\beta_{f,t}) \tag{23}$$

for $1 \leq f \leq F$ where $1 \leq t \leq \Phi$-1. Here, N_t holds the number of maximum interpolated points on a specified t gap. Key points set for horizontal interpolation is given as $H_{t,h}$, where h is the index for the set in t^{th} gap. We can then define

$$H_{t,h} = \{ W_{t,h,m} \} \tag{24}$$

for $1 \leq t \leq \Phi$-1, $1 \leq h \leq N_t$ and $1 \leq m \leq M_{t,h}$. m represents the index for the key points from a specific set, $H_{t,h}$. $M_{t,h}$ contains the total number of points in set $H_{t,h}$. Both $T_{f,t,\gamma}$ (in Equation(21)) and $W_{t,h,m}$ (in Equation (24)) represent 3D coordinates where they are similar to $T_{f,t,\gamma,d}$ and $W_{t,h,m,d}$ with d=1,2,3.

We choose the vertical interpolated points from model V, traverse f and t, and register the points in H as the key points for the horizontal interpolation mechanism. A maximum set of horizontal key points for each gap is fixed based on the maximum vertical interpolated points available in that gap (Equation (23)). Since every f is traversed in order to register the vertical interpolated point into the set, $H_{t,h}$, the number of total points in $H_{t,h}$ is same as S (Equation (15), $M_{t,h}$=S). Equations (25) and (26) describe the methods to identify key points in set $H_{t,h}$:

$$W_{t,h,m} = T_{f,t,\gamma} \tag{25}$$

for $1 \leq h \leq N_t$ and $1 \leq f, m \leq S$ where $\gamma \in Z^+$ and is defined as

$$\gamma = \frac{h}{N_t} \times G_{f,t} \tag{26}$$

for $1 \leq t \leq \Phi$-1.

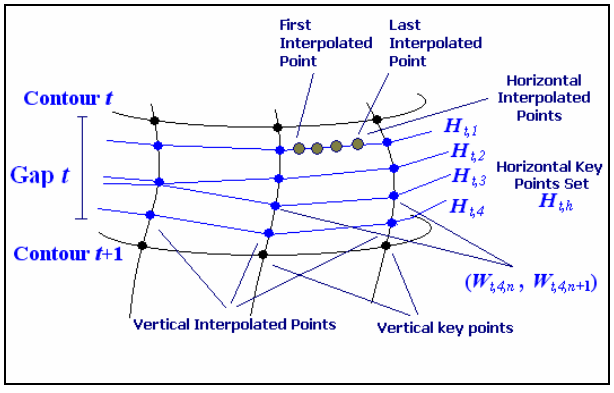


Figure 7: The relationship between vertical interpolated points (blue) and horizontal control points set. Black color dots are the VI key points.

The point's index, m, in set H is arranged into a 'ring' manner (see Figure 8b). Then the new points are generated by considering the pair of consecutive key points in that set. An elaboration on sorting W into a 'ring' manner is presented in APPENDIX A. The two adjacent points in each set H is

denoted as ($W_{t,h,n}$, $W_{t,h,n+1}$) for $1 \leq n \leq M_{t,h}$-1. Additional points are interpolated in between these two points. Let η be the maximum distance allowed among the interpolated points and also between the control points with the first and last interpolated points (see Figure 7). To achieve this condition, the number of interpolated points for each pair of fulcra points is given by:

$$g = \frac{\sqrt{\sum_{d=1}^{3}\left(W_{t,h,n,d} - W_{t,h,n+1,d}\right)^2}}{\eta} \tag{27}$$

where $g \in Z^+$.

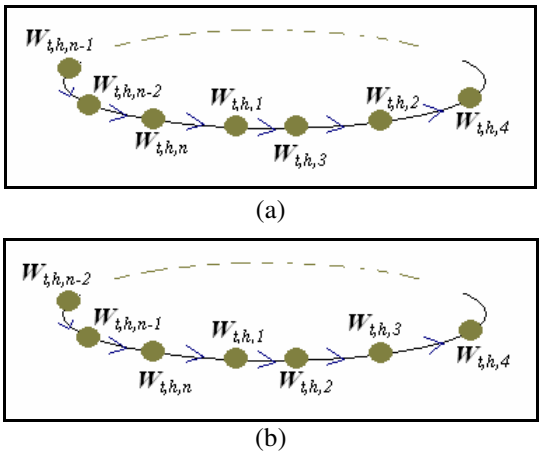


Figure 8: (a) The unsorted points in set H. (b) Shows the key points in *H* after being sorted into a 'ring' (Appendix A). $n = M_{t,h}$.

2.4 Rule-based Heart Fiber Reconstruction

From the interpolation process, the number of epicardium points in our model increases. Let $M=\{P_i\}$ where $1 \leq i \leq N$ and $P_i \equiv P_{i,d}$ is a 3D coordinates point and where d=1,2,3. M set consists of epicardium points while N is the number of points in the model. P_i in M is arranged in such a way that $P_{1,3} \leq P_{2,3} \leq \ldots \leq P_{N\text{-}1,3} \leq P_{N,3}$. The sorted points help to reduce significantly the computational time required during the fiber model formation.

The fiber model is defined as

$$\hat{H} = \{F_1, F_2, \ldots, F_s\} \tag{28}$$

where F_f represents a list of connected fiber points and $1 \leq f \leq S$. S is the total number of constructed fibers. μ_f is the total number of fiber points registered in a specific F. Let

$$F_f = \{\ \rho_m^f\ \} \tag{29}$$

where $1 \leq m \leq \mu_f$ and $\rho_m^f \equiv \rho_{m,d}^f$. The fiber reconstruction mechanism registers a point P from model M into set F based on the following algorithms.

2.4.1 Fiber Point Registration Algorithm

This algorithm aims to find a target point (from *M*), denoted as P_α^*, and register it into a specified fiber list, F_f. P_β^o is the base point used during the determination of next target fiber point. The registered target point is assigned to ρ_r^f, where r is defined as the value of the fiber registration cycle. ρ_r^f represents the registered fiber point for current cycle. In the next cycle, point P_β^o will be set to ρ_r^f in order to act as the base point. In other words, the selected target point will become a base point for the next cycle of fiber registration process. The value of β is used to hold the point's index, i, carried by ρ_r^f.

Steps (i) to (ix) show the fiber reconstruction algorithm:

i. Initialize the starting point, ρ_0^f, for each fiber set with a P_i chosen from *M*, where $\rho_0^f = P_i$ for $1 \le f \le S$ and $i \in N$. Assigned r=0.

ii. For a specified fiber f, let $P_\beta^o = \rho_r^f$.

iii. Let $\mathbf{P}^* = \{ P_\alpha^* \}$, $\alpha \in$ n($\mathbf{P}^*$) be a set that holds a list of points which have a minimum distance to the base point, P_β^o.

iv. Search and store all points P_k from *M* where k is within the range (β+1, $\beta+\Delta n$) into $\mathbf{P}^*$. Note that $\beta+\Delta n \le N$ and all points in $\mathbf{P}^*$ are required to conform to rule (iii).

v. Filter all points in $\mathbf{P}^*$ so that the value of θ (Equation (31)) corresponding to each point is within the range $\theta_c \pm \sigma$.

vi. Determine a target point, P_α^* from set $\mathbf{P}^*$ and assign it to ρ_r^f using the following steps:

```
FOR EACH P*α in set P*
 IF Rr Rr-1 >0
   IF P*α is not registered to other fiber set
      Register P*α in Pa*
   ELSE
      Register P*α in Pb*
   END IF
 END IF
END FOR

IF Pa* is NOT empty
     Choose a P*α from set Pa* which has the smallest Δθ AND the smallest |Vr|
ELSE
     Choose a P*α from set Pb* which has minimum of Δθ AND the smallest |Vr|
END IF
```

vii. If no new P_α^* is registered in a specified fiber f, fiber construction mechanism is considered complete for that fiber.

viii. Repeat steps (ii) to (vii) for all fiber, $1 \le f \le S$.

ix. Increase r by 1and repeat steps (ii) to (viii).

Step (iv) may require huge computational power when N is large. Significant processing power is required to search through the entire set of points in order to get a number of points closest to P_β^o. This search function is performed in every fiber point registration cycle for each fiber set. Thus, in order to reduce search complexity while maintaining a suitable search domain size (see Step(iv)), Δn is introduced, where $\Delta n=B+Cr$. B is a constant value that holds the minimum number of search points that needs to be explored. C is an incremental factor and r is a dynamic value that starts from 0 and increases by 1 after each registration cycle (step (ix)).

2.4.2 Fiber Angle

The rule applied in step (vi) takes into consideration the fiber angle's range ($\Delta\theta$). The vectors $\mathbf{V_r}$ and $\mathbf{V_b}$ are formed as shown in figure 9. Vector $\mathbf{V_r}$ is formed by the base point (P_β^o) and target point (P_α^*). Vector $\mathbf{V_b}$ is formed by the base point with a point $P_\alpha^\perp$, where it is defined as:

$$P_\alpha^\perp = (P_{\alpha,1}^*, P_{\alpha,2}^*, P_{\beta,3}^o) \qquad (30)$$

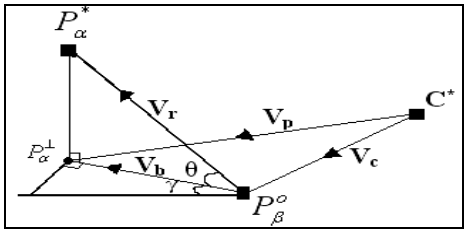


Figure 9: Schematic diagram to show the relation between base point, target point and centre point in 3D space. The diagram includes the vectors as well as their directions.

Value θ represents the epicardium's fiber angle, where its range is defined as $0 \le \theta \le \pi/2$. The value is calculated via:

$$\theta = \cos^{-1}\left(\frac{V_r \bullet V_b}{|V_r||V_b|} \right) \qquad (31)$$

The methodologies to obtain values of the fiber angle around epicardium have been discussed in detail in [10] and [20]. All P_α^* points in set $\mathbf{P}^*$ have a θ value within a specified range, $\theta_c \pm \sigma$ (see Step (v)): θ_c=60º and σ=15º [7][10]. $\Delta\theta$ is introduced as:

$$\Delta\theta = |\theta - \theta_c| \qquad (32)$$

where $\Delta\theta$ is needed to perform step (vi).

2.4.2 Fiber Path

The rule in step (vi) also take into consideration the fiber's direction. The vectors $\mathbf{V_p}$ and $\mathbf{V_r}$ are formed as shown in figure 9. C^* is a centre point where the value of its z dimension is the same as P_β^o. Both the x and y dimensions for C^* have the value of 32. This is because model M is at the centre of 64 x 64 x 64 experimental space (see Section 2.2.2). Since the epicardium points from model M are formed

from the contours (see Section 2.2.2), we assume that C^* is the centre points for a ring formed by those points from model *M* which have the same *z* dimension value as point P_β^o. So, the vector $\mathbf{V_c}$ is the vector formed by points C^* and P_β^o while Vector $\mathbf{V_p}$ is formed by the centre point with points $P_\alpha^\perp$ (Equation (30)).

When $|\mathbf{V_r}|$ and γ are small, curve $P_\beta^o\ P_\alpha^*$ ≈ line $P_\beta^o\ P_\alpha^*$ (see Figure 9). Hence, the curvature of epicardium will still be maintained during the fiber formation.

Note that

$$R\ \hat{\mathbf{n}} = \mathbf{V_c} \times \mathbf{V_p} \qquad (33)$$

where $\hat{\mathbf{n}}$ is the vector unit for the normal vector from the plane where vectors $\mathbf{V_p}$ and $\mathbf{V_c}$ lie on (see Figure 10). The direction of vector $\mathbf{V_r}$ needs to be always clockwise so that all the constructed fiber paths are in the same direction. To satisfy this criterion, we make an assumption that fibers tend to lie in planes parallel to the epicardium and approach a longitudinal orientation on the ventricular surfaces [30].

However, there is a possibility that a point from $\mathbf{P}^*$,while conforming to the above fiber angle rule, will have a direction of vector $\mathbf{V_r}$ which is on the other side (see vector $\mathbf{V'_r}$ in Figure 10).

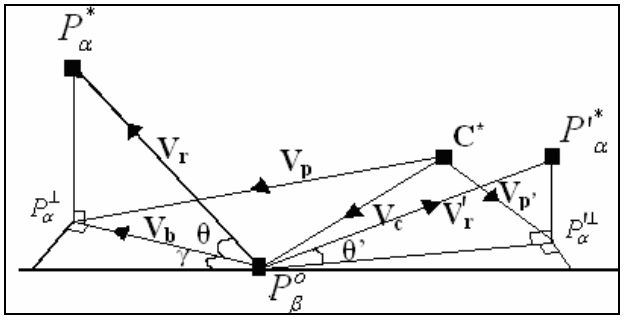


Figure 10: Schematic diagram to show the existence of P'^*_α with its related vector and direction.

Let

$$R'\ \hat{\mathbf{n}} = \mathbf{V_c} \times \mathbf{V_{p'}} \qquad (34)$$

Here, we introduce *R* (in Equation (33)) and R' (in Equation (34)) as constant variables. $R\ R' < 0$ is always true.

Let R_r and R_{r-1} be the values of *R* calculated during the current cycle (*r*) and the previous cycle (*r*-1) respectively. So, our rule is to choose a point, P_α^* which maintains the condition:

$$R_r R_{r+1} > \mathbf{0} \qquad (35)$$

for $0 \le r < r^*$ where r^* is the total number of fiber registration cycles.

2.5 Refinement of Fiber Representation Model

Every pair of consecutive fiber points in each fiber has varying gaps between each other. Some of the large gaps may cause inaccuracy during cardiac simulation. Hence, we need to ensure that the gap for

each pair of consecutive points is limited to a reasonable distance. For the pair of consecutive fiber point that exceed the allowed distance, Fiber-Point Interpolation will be applied to generate additional points in order to refine the fiber model (see Figure 11a).

In addition, when the gap between two adjacent fibers is huge, a new fiber set may need to be constructed (see Figure 11b). On the other hand, a fiber set will be removed if the fibers are too close to each other (see Figure 11b). We apply Fiber-Based Least Mean Square (LMS) analysis (section 2.5.2) on all pairs of adjacent constructed fiber sets. The LMS value of two adjacent fibers set is used to identify whether the two fibers are too far or too close from each other. Section 2.5.2 outlines the implemented algorithm.

2.5.1 Fiber-Point Interpolation

Let the maximum allowed distance be Δd_{max}. For the pair of two adjacent fiber point (ρ_m^f , ρ_{m+1}^f) where its gap exceeds Δd_{max}, a new point, denoted as ρ_j^f , is inserted between them. Note that ρ_m^f and ρ_{m+1}^f for $1 \le m < \mu_f$ are continuous points from a fiber point list referred to a specified fiber set, *f*. ρ_j^f is interpolated linearly as

$$\rho_{j,d}^f = \frac{1}{2}(\rho_{m,d}^f + \rho_{m+1,d}^f) \tag{36}$$

for d=1,2,3.

The interpolation stated above is repeated again if the gap between the fiber points and the newly interpolated point are still larger than $\Delta d_{max.}$ A similar mechanism will be applied to all set of fiber set, where $1 \le f \le S$.

2.5.2 Fiber-Based Least Mean Square analysis

Denote that Fiber-Based Least Mean Square (FB-LMS) value for 2 adjacent fibers i, j , where $1 \le i < j \le S$ is ω^{ij} .

Let $t = \max(\mu_i, \mu_j)$

Let $h = i \quad , \mu_i = \max(\mu_i, \mu_j)$
$\quad = j \quad , \mu_j = \max(\mu_i, \mu_j)$

Let $k = j \quad , \mu_i = \max(\mu_i, \mu_j)$
$\quad = i \quad , \mu_j = \max(\mu_i, \mu_j)$

Thus,

$$\Delta\xi_{mn} = \sqrt{\left(\rho_{m,d}^k - \rho_{n,d}^h\right)^2} \tag{37}$$

Let $e_m = a$ (38)

where $\Delta\xi_{ma} = \min(\Delta\xi_{mn})$

$$\omega^{ij} = \frac{\sqrt{\sum_{m=1}^{\mu_k}\sum_{d=1}^{3}\left(\rho_{m,d}^{k} - \rho_{e_m,d}^{h}\right)^2}}{\sum_{m=1}^{\mu_k}\sqrt{\sum_{d=1}^{3}\left(\rho_{m,d}^{k} - \rho_{e_m,d}^{h}\right)^2}} \tag{39}$$

where $1 \le m \le \mu_k$, $1 \le n \le \mu_h$.

We interpolate a new set of fiber, denoted as F_v using the Cubic Hermite function mentioned in section 2.3 (Equation (13)). The fulcra points are obtained from the two adjacent fiber set, F_i and F_j (see Figure 11b).

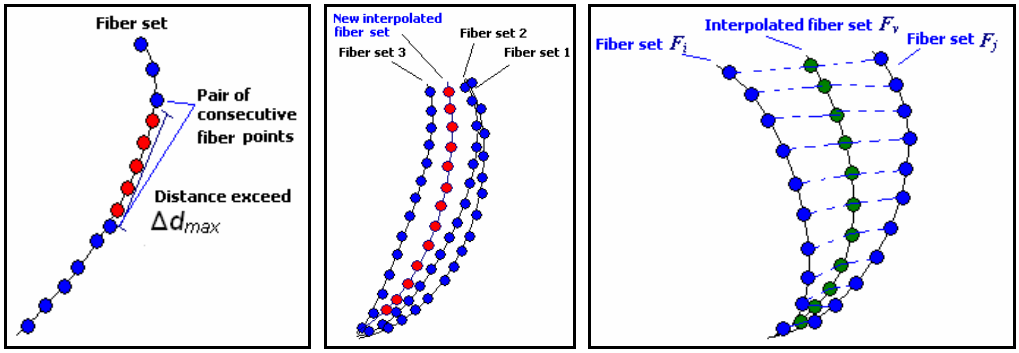


Figure 11: (a) The distance between a pair of consecutive fiber points exceeds Δd_{max} and the Fiber-Point Interpolation mechanism is executed to generate additional points (red) between the adjacent points. (b) A new interpolated fiber set (red) is inserted between fiber set 2 and 3. Fiber set 1 will be removed as it is considered too close to fiber set 2. (c) The fulcra points to interpolate fiber set F_v (green) are based on fiber sets F_i and F_j.

2.6 Fluid Marker Generation

Fluid marker is a set of fibers which do not contribute any force into the fluid experimental space during cardiac simulation. However, their movement will be influenced by the surrounding fluid. Hence, the dynamic for the fluid marker can be used to analyze the blood movement inside the LV chamber. For this reason, the geometry representation of the fluid marker needs to conform in terms of physical location within our fiber model. Specifically, the fluid marker geometry needs to be generated in such a way that they are inside the left ventricle chamber. Figure 12 shows the layout of the fluid marker as well as its relationship within the LV. The fluid markers are generated at multiple horizontal layers where each layer consists of multiple rings. The reason the fluid markers are generated with multiples layers and rings in the LV is to analyze the fluid dynamics at different locations within the LV chamber (Figure 12). The outermost ring in each layer is used to observe the fluid mechanics near the endocardium. The lowest layer horizontal marker is acquired to analyze the fluid dynamic near the heart's apex.

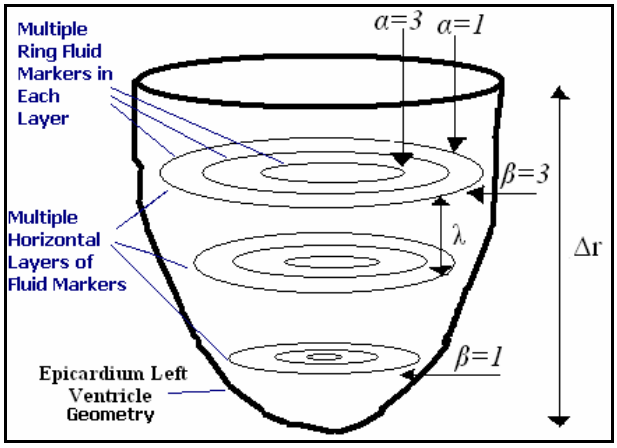


Figure 12: The layout of the fluid marker and its relationship within the LV. The difference between the horizontal layers of fluid markers and rings of fluid markers is clearly shown.

2.6.1 The Fluid Marker Generation Algorithm

Here we denote the fluid marker model as $\hat{W}$, where $\hat{W} = \{ W_{1,1}, W_{1,2}, \ldots, W_{\alpha,\beta}, \ldots, W_{\alpha^*-1,\beta^*-1}, W_{\alpha^*,\beta^*} \}$. α is the index for ring markers on a horizontal layer markers while β is the index for the horizontal layers itself. α^* and β^* hold the total number of ring markers and horizontal layers respectively.

The smallest value of α represents the markers near the epicardium wall while the larger values of α are the markers found near the centre of the LV cavity. For β, a larger value of β represents the marker near the LV's base while the smaller value represents markers near the apex. Figure 12 visualizes the methodology to index fluid markers. So, the 3D marker points are denoted as $b_m^{\alpha\beta}$ where

$$W_{\alpha,\beta} = \{ b_m^{\alpha\beta} \} \tag{39}$$

for $1 \le m \le \gamma_{\alpha,\beta}$. $\gamma_{\alpha,\beta}$ refers to the total number of marker points in a specific set $W_{\alpha,\beta}$. The α and β used in notation $b_m^{\alpha\beta}$ show the point b_m is belonged to the set $W_{\alpha,\beta}$.

The algorithm for fluid marker model construction is given below: Let

$$\Delta r = \max_{1 \le m \le \mu_f} \left(\rho_{m,3}^f \right) - \min_{1 \le m \le \mu_f} \left(\rho_{m,3}^f \right) \tag{40}$$

for $1 \le f \le S$. Then, let

$$\beta^* \in Z^+, \beta^* = \Delta r / \lambda \tag{41}$$

where λ is the gap distance between two adjacent horizontal marker layers. Hence, we define

$$L_n = \min_{1 \le m \le \mu_f} \left(\rho_{m,3}^f \right) + n\lambda \tag{42}$$

for $1 \le n < \beta^*$ and $1 \le f \le S$.

Let

$$T_{\psi,d}^{\beta} = \rho_{m,d}^f \tag{43}$$

for

$$L_n - \frac{\lambda}{2} \le \rho_{m,3}^{f} < L_n + \frac{\lambda}{2} \qquad (44)$$

where d=1,2,3 , $1 \le m \le \mu_f$,$1 \le f \le S$ and ψ=1,2,…,ψ_β^*. ψ_β^* is the number of total points obtained from $\rho_{m,d}^{f}$ for a specified range defined by equation (44).

Define the centre points for each $T_{\eta,d}^{\beta}$, $1 \le \eta \le \psi_\beta^*$ as C^β where

$$C_d^\beta = \{\min(T_{\eta,d}^{\beta}) + \max(T_{\eta,d}^{\beta})\} / 2 \qquad (45)$$

for d=1,2,3. After all, the marker points are generated using the following equation:

$$b_{\eta,d}^{\alpha\beta} = \frac{(\alpha^* + 1 - \alpha)T_{\eta,d}^{\beta} + \alpha C_d^\beta}{\alpha^* + 1} \qquad (46)$$

for d=1,2,3 ,$1 \le \eta \le \psi_\beta^*$,$1 \le \alpha \le \alpha^*$ and $1 \le \beta \le \beta^*$.

Some pairs of adjacent marker points from set $W_{\alpha,\beta}$ are removed as their distance are too close to each other. Hence, the original number of total point obtained by $W_{\alpha,\beta}$ is reduced from ψ_β^* to $\gamma_{\alpha,\beta}$.

2.7 Cardiac Dynamic Simulation

We implement the cardiac simulation mechanism using Immersed Boundary (IB) method. At each time step, the fibers update its force value to reflect the activity of the boundary (e.g. generated fibers' forces from the contraction of heart muscles). The forces then exert into the fluid (e.g. blood). Overall velocities in the fluid space will then be calculated. Based on [28], we assume that in a fiber-fluid heart model, the fluid is Newtonian and incompressible, and the fibers are massless and neutrally buoyant. Finally, the velocities from the fluid will then update the fiber points' velocity and the fiber points are moved to a new location. The following section explains in details for these steps in detail.

2.7.1 Geometry Representation

In the IB Method, the boundaries are represented by lists of fibre points in Lagrangian description, denoted as $X_{i,d}(t) \equiv X_i(t)$ where i represents the fiber entity index and d represent the dimension fields. t represents the fiber points location at a specified time step.

Our fiber model (see Equation (28) and (29)) is also represented in Lagrangian description. Thus, we imply that the location of fiber point from set f at time t is $X_{i,m}(t)$ where $X_{i,m,d}(t) \equiv X_{i,m}(t) \equiv \rho_m^i$ and $1 \le m \le \mu_i$. i represents a fiber set entity.

Fluid surrounding the boundaries is represented by a 3D rectangular cubic lattice with the width of the mesh set as h, and is described in Eulerian description. Let $\mathbf{x} = \{\mathbf{x}_{jlh}\}$ where j, l, h represent the fluid grid index.

2.7.2 Fiber Force Generation

We specify the elastic properties of the immersed boundaries with an energy function. This function is created for each entity of fiber (F, see Equation (29)). The elastic forces that act on the IB points are calculated from the derivatives of the energy function [23]. The elastic energy of each fiber entity is the sum of three energies produced from each pair of consecutive fiber points: the stretching energy act

as Hooke's Law Springs denoted as E_S, the bending energy, denoted as E_B and the tether energy. But, in our case, we desert the usage of tether energy.

The E for the system at time t for all fiber entities is obtained from the formula below:

$$E(X,t) = \sum_{i=1}^{S} E_i(X_{i,m}) \tag{47}$$

for $1 \leq m \leq \mu_i$, where

$$E_i(X_{i,\mu_i}) = \sum_{q=1}^{N-\delta} E_S(X_{i,q}, X_{i,q+1}) + \sum_{q=1+\delta}^{N-\delta} E_B(X_{i,q-1}, X_{i,q}, X_{i,q+1}) + \sum_{q=1}^{N} E_T(X_{i,q}) \tag{48}$$

for δ=0 if the fiber connecting $X_{1,i}$ with $X_{i,N}$ and δ=1 otherwise; and

$$E_S(\mathbf{X}_{p1(l)}, \mathbf{X}_{p2(l)}) = \frac{1}{2} S_l(\left|\mathbf{X}_{p1(l)} - \mathbf{X}_{p2(l)}\right| - R_l)^2 \tag{49}$$

where elastic link l connects immersed boundary points with indices *p1(l)* and *p2(l)* and $\mathbf{X}_{p1(l)}$ and $\mathbf{X}_{p2(l)}$ is their coordinates. R_l is the default rest length and S_l denotes the stiffness coefficient; and

$$E_B(\mathbf{X}_p, \mathbf{X}_q, \mathbf{X}_r) = \frac{1}{2} S_B[\hat{z} \bullet (\mathbf{X}_r - \mathbf{X}_q) \times (\mathbf{X}_q - \mathbf{X}_p) - C]^2 \tag{50}$$

where $\boldsymbol{X}_p$ and $\boldsymbol{X}_q$ are joined by an elastic link, as are points $\boldsymbol{X}_q$ and $\boldsymbol{X}_r$. S_B is the bending stiffness, C is a specified 'curvature'.

Thus, note that the forces generated by a specified fiber point as $\mathbf{F}_{i,m}$ and it is calculated from the derivative of the elastic energy *E(X,t)* with respect to the coordinates in $\mathbf{X}_{i,m}$ as shown below:

$$\mathbf{F}_{i,m}(\mathbf{X},t) = -\frac{\partial E(\mathbf{X},t)}{\partial \mathbf{X}_{i,m}} \tag{51}$$

The minus sign is chosen to drive the system towards a minimum of the energy scenario. Observe that the constants such as S_l , R_l , S_B and C need to be specified and the details on how to is also provided in [23].

2.7.3 Spread Fiber Forces onto Fluid Lattice

The IB points do not coincide with the grid points from the rectangular fluid lattices. The generated fiber forces need to be exerted onto the rectangular cubic fluid lattice. Hence, a mean of communicating information between these two sets of points is required. This is performed via a Dirac delta function, where it's defined as:

$$\delta_h(\mathbf{X}) = h^{-3} \prod_{d=1}^{3} \phi\left(\frac{x_d}{h}\right) \tag{52}$$

where x_d is the dimension component of $\mathbf{X}$. The δ-function will produce 0 for the grid points on fluid lattice which are located more than $2h$ surrounding from $\mathbf{X}$. The details on the construction of $\phi(r)$ are given in [13,14].

2.7.4 Fluid Velocity Calculation

The fluid force density, denoted as $\boldsymbol{f}^n$ is given by:

$$f^n = \sum_{i=1}^{S} \sum_{m=1}^{\mu_i} \mathbf{F}_{i,m}^n(\mathbf{X}) \delta_h(\mathbf{x} - \mathbf{X}_{i,m}^n) \tag{53}$$

The fluid force density is evaluated at each of the IB point's location in each time step. The '*n*' notation states the time step, instead of using *t*. The velocity of each fluid cell is then calculated from these local forces by solving the Navier-Stokes equations:

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \bullet \nabla \mathbf{u}\right) = -\nabla p + \eta \nabla^2 \mathbf{u} + f \text{ and } \nabla \bullet \mathbf{u} = 0 \tag{54}$$

where ρ , η and p represent the fluid density, viscosity and pressure respectively. The equations are solved using numerical method introduced by Chorin [1, 2]. The equations are discretized on a cubic lattice of mesh width h and solved using finite-difference method.

Note that the fluid's velocity and the fiber configuration at time step n are given as ($\mathbf{X}^n$, $\mathbf{u}^n$). The equations can be solved simultaneously for the unknowns ($\mathbf{u}^{n+1}$, p^{n+1}), as illustrated below:

$$\rho\left(\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \sum_{\alpha=1}^{3} u_\alpha^n D_\alpha^{\pm} \mathbf{u}^n\right) = -D^0 p^{n+1} + \eta \sum_{\alpha=1}^{3} D_\alpha^+ D_\alpha^- \mathbf{u}^{n+1} + f^n \text{ and } \mathbf{D}^0 \bullet \mathbf{u}^{n+1} = 0 \tag{55}$$

The formation of $\mathbf{D}^{\pm}$ and $u_\alpha^n D_\alpha^{\pm}$ which analogous to ∇ , are stated clearly in [13,14].

2.7.5 Interpolate Velocity to Fiber Points

Finally, the fiber's velocity is updated based on its surrounding fluid's velocity using the smooth Dirac delta function again. The fibers are moved to a new position, based on their velocities. The formulas below illustrate this step: Let

$$U_{i,m}^n = \sum_{i=1}^{S} \sum_{m=1}^{\mu_i} \mathbf{u}^n(\mathbf{x}) \delta_h(\mathbf{x} - X_{i,m}^n) \tag{56}$$

Then,

$$X_{i,m}^{n+1} = X_{i,m}^n + \Delta t U_{i,m}^n \tag{57}$$

Once this has been performed, a time step is considered complete.

3. Visualization and Results

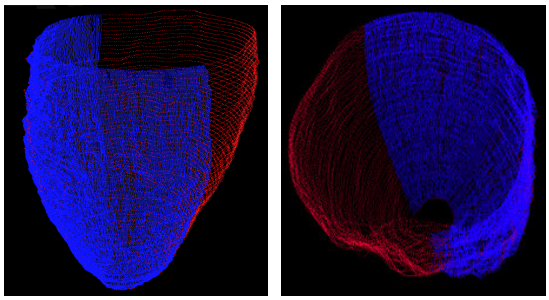
3.1 Visualization Environment and Programs

We have developed a series of visualization programs, which translate sets of numerical simulated results into 3D objects. They are equipped with interactive computer graphics features, whereby a user is able to toggle appearance of fiber in our heart model in real time. We can also interactively change our point of view and magnify any region of interest Furthermore these programs allow the users to rearrange the simulated data sets and display them in a frame by frame manner in order to produce an animated effect. The programs allow the users stop at a certain frame that they interested**.** Moreover, the programs are implemented to be generic enough to enable the display of our fluid markers model.

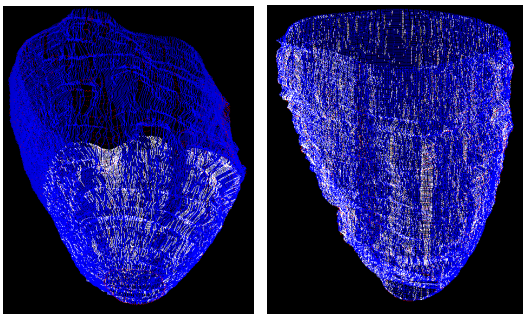
Besides, the programs are used to view and verify the cardiac fiber model in preparing stage before proceeding with the actual cardiac simulation.

However, the functionality of the visualization programs is not just limited to displaying the simulated results. It is also used to visualize the simulation processes of our algorithm. For example: we have implemented a program to visualize the fiber reconstruction process. The program is developed in such a way that it is able to display the constructed fibers during the fiber construction process. In other words, we are able to verify the constructed fiber result in real time before the program completed. This significant advantage allows observation and modification of rules and simulation parameters setting via visualization. Early detection of faulty design has apparently saved us a lot of valuable computational time and cost.

We leverage the implementation of our programs on OpenGL technologies which include OpenGL Utility Toolkit (GLUT) 3.6. C programming language has been used to develop these programs. They are compiled into several executable programs which able to run in different OS platforms such as Window XP, Linux Red Hat 9.0 and Irix 6.5.13. Most of our programs are implemented on a SGI workstation (2.4 MHZ Processor speed, 2GB of memory space) because of its well-known performance in real time graphics visualization tasks.



(a) (b)



(d) (e)

Figure 13: (a) and (b) show the vertical interpolation results which captured from different points of view, (c) shows the horizontal interpolation is being executed, (d) shows the complete output after both horizontal and vertical interpolation procedures have been completed.

3.2 Simulation Results

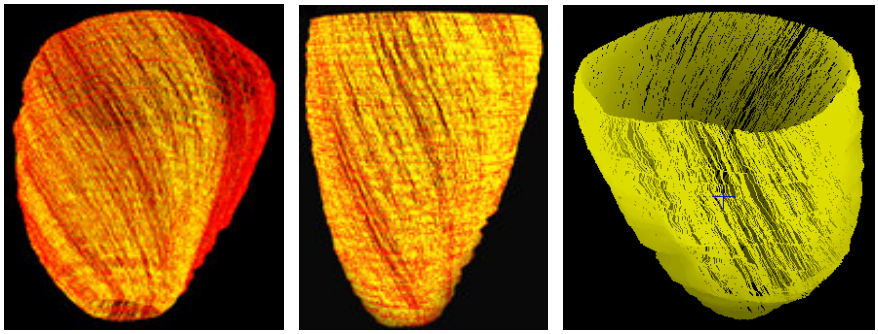
3.2.1 Vertical and Horizontal Interpolation

Figure 13 shows the heart model after Cubic Hermite Interpolation (see Section 2.3.1 and Section 2.3.2). We set λ=0.15 and η=0.10. Red colour points show the control points for the vertical

interpolation. The vertical interpolated points are in blue colour, which are used as the key points for horizontal interpolation. The horizontal interpolated points are in white colour.

3.2.2 Rule-Based Fiber Reconstruction

From the interpolated point obtained (Figure 13d), we proceed to construct the left ventricle fiber model based on the rules described in section 2.4.1. Figure 14 shows the results of constructing 1,200 fibers (yellow path) which contain 584,589 registered fiber points. The model points are represented in red colour while the yellow colour refers to the registered fiber points. The system consumed around 21 CPU hours in order to produce a complete fiber model (Figure 14b and 14c),



(a) (b) (c)

Figure 14: (a) The result from the simulation of the rule-based fiber reconstruction process. (b) The result of the fiber model with model control points. (c) Fiber model without the model points.

3.2.3 Refinement of the Fiber Model

Figure 15(a), 15(b) and 15(c) show the results of fiber model constructed after the refinement processes (see Section 2.5). The model is captured from different point of view. Figure 15(d) and 15(e) show the fiber model with some of the fibers hided in order to view into the inside of the heart chamber.

3.2.4 Fluid Markers

We apply $\alpha^* = 5$ and $\beta^* = 15$ in order to generate all the fluid marker points (see Equation (46)). Figure 16 shows the generated marker point model. Figure 16(a) and 16(b) show the full set of fluid markers in different perspective views where all the ring markers on each horizontal layer are displayed. Figure 16(c) depicts the marker points for rings number 2 and 3 from all horizontal layers. Figure 16(d) illustrates the ring markers number 4 and 5. Figure 16(e) is the outermost layer of ring markers (near to epicardium wall), where α=1 (see Figure 12).

3.2.5 Cardiac Fiber and Blood Simulation Result

Each of the fiber point contains parameters which are needed for IB Method simulation. The parameters are default rest length (R_l, see Equation (49)) for each consecutive fiber points and stiffness coefficient (S_l, S_B , see Equation (49) and (50)). Basically, the stiffness value is determined by the cardiac cell material properties. The contraction factor is also applied. High contraction factor implies larger force produced by the fiber point during its contraction phase and vice versa.

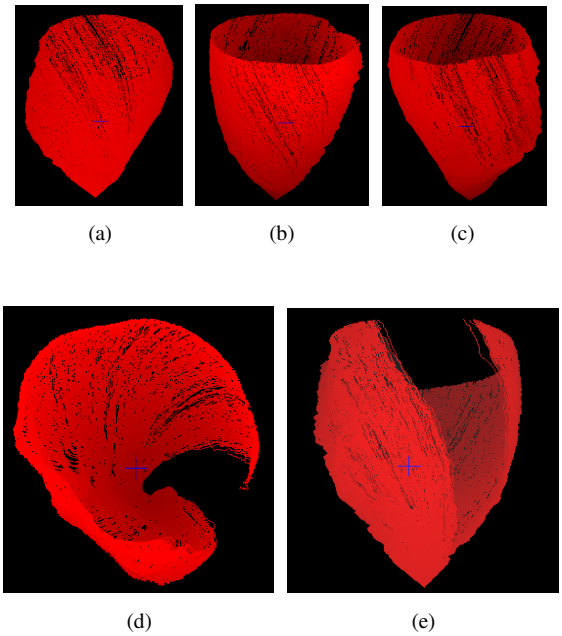


Figure 15: Fiber model is constructed by 1,038 fibers with the total of 371,658 fiber points after the refinement processes.

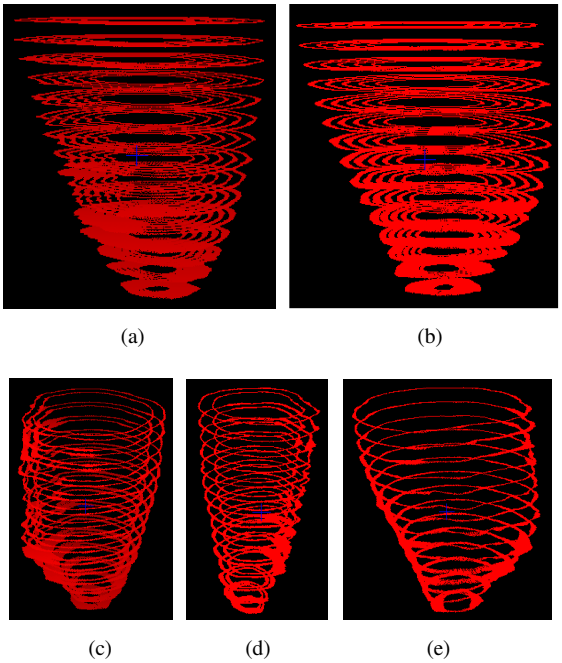


Figure 16: The result of generated markers points from our fluid markers generation mechanism.

We simulate the heart dynamic at early of systolic stage. We assigned our fiber contraction mechanism in such a way that the contraction move sequentially from the apex to the base. The contraction factor at the apex is assigned to be greater than the factor around the base. Furthermore, we

assume that the materials properties of the fibers cell from the LV epicardium layer are all identical. Thus, we apply a unique stiffness values for all the fiber points. A fiber model with 1,038 fibers and 371,658 fiber points incorporated with their correspondent parameters (R_l, S_l, S_B) are provided as input for the cardiac simulation. The simulation consumed about 11 CPU hours to complete a total of 1024 time steps simulation.

After a predefined number of time-steps, simulated numerical results for the fiber model and fluid marker points are dumped to an output file. The results are fed into our visualization programs for cardiac and blood dynamic analysis purpose (see Figure 17 to 20).

We generate the output data for every 8 time step as a frame in order to obtain 128 frames in total. Figure 12(a) to 12(j) show the dynamics of the LV cardiac fiber from the early ejection stage.

A series of outermost (α=5) ring markers from each horizontal layers pictures are shown in Figures 18(a) to (j). The fluid markers simulated results are mapped (in term of physical time) with the cardiac fiber movement shown in figure 17.

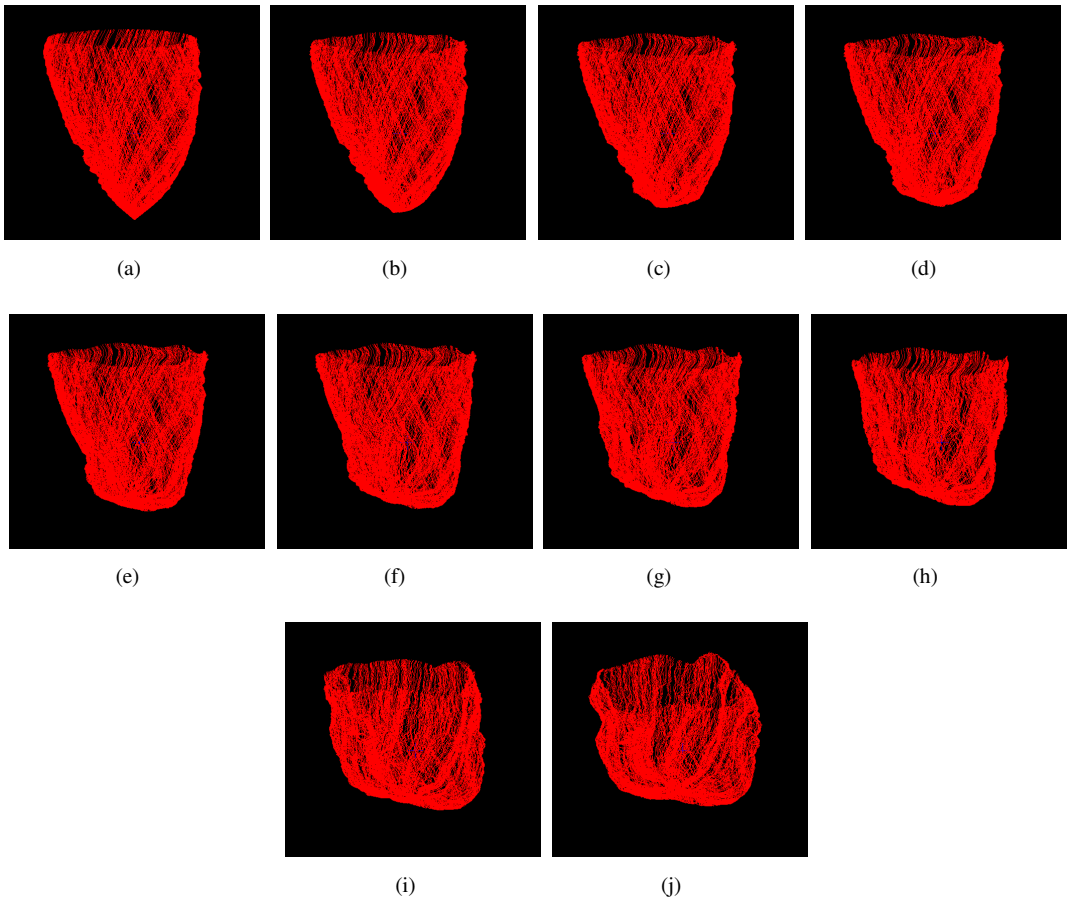


Figure 17: The result of simulated LV cardiac muscle dynamic start from early ejection. From (a) to (j), each frame represents the mechanics of simulated cardiac muscle at different physical time. Starting from 0.05ms for (a) and a snapshot after every 0.05ms, (b) to (j). (j) shows the fiber dynamic at 0.5ms.

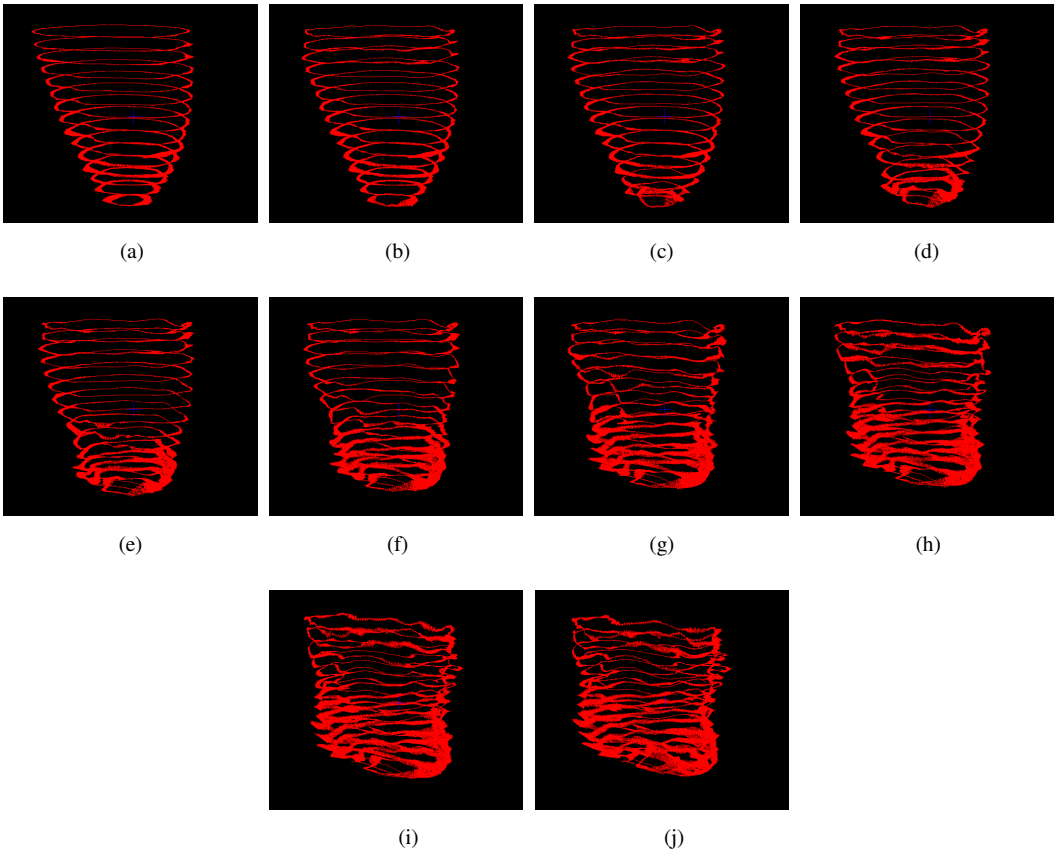


Figure 18: The results of simulated LV blood flow start from early ejection at the same physical time as mapped in figure 17. Only the ring markers nearest to the epicardium were shown in order to display the dynamic of the blood flow in LV chamber.

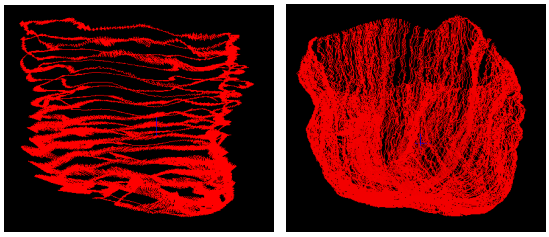
By turning on only the innermost ring of fluid markers (α=1) on all horizontal layers, the blood flow mechanics at the centre of LV cavity (see Figure 20(a)) are captured. Besides, by turning on only certain horizontal layers of fluid markers ($1 \leq \beta \leq 5$), blood flow around the apex can be captured (see Figure 20(d)) as well. In addition, by manipulating the appearance of ring markers and horizontal layer of fluid markers, the blood flow mechanics at different location in LV chamber can be visualized (see Figure 20(b) to (d)). Apparently, we are able to study a patient's specific blood flow using these results.

3.2.6 Cardiac Simulation Analysis

The dynamics of cardiac fibers at different stages and physical time are studied. We observed that the cardiac fibers demonstrate a twisted effect during the contraction stage (see Figure 12(f)) where these effects have been observed via clinical research [5].

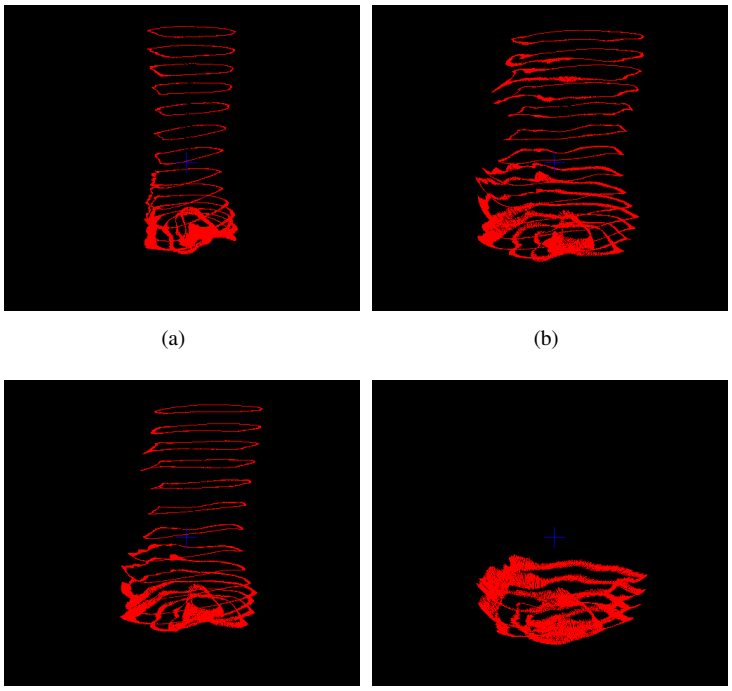
Figure 19(a) and 19(b) show the mechanics of the blood flow together with its fiber mechanics in LV chamber at 0.5ms simulation time. These results are used as the control set. We compare the mechanics of the blood flow between the positions nearest to the LV wall with the different area within the LV chamber. In this case, the second and forth ring markers from all horizontal layers were shown (see Figure 20(b) and 20(d)). We observed that there is not much movement of blood at the

middle of the LV chamber while the blood flow near the LV apex and the LV wall has a high expulsion.



(a) (b)

Figure 19: The mechanics of blood flow nearest to the LV wall together with the cardiac fiber dynamic at the time at 0.5ms.



(a) (b)

(c) (d)

Figure 20: (a) shows the innermost ring fluid markers from each of the horizontal layers at 0.5ms, (b) and (c) show 2nd and 4th layer of ring fluid markers from each of the horizontal at 0.5ms respectively, (d) shows several selected horizontal layer (outermost ring) of fluid markers which are located near the apex of LV chamber.

4. Discussion and Future Work

Our work described procedures and methods which are fairly important in constructing a virtual human heart by using a non-invasive approach. Prior to MR images for 3D model reconstruction, it

was impossible to acquire a very detailed cardiac function maps of the human heart. According to Hunter *et al* [6], the material properties of the heart fiber are another major factor to be considered in order to construct an accurate heart model. However, the method using rule based mechanism in constructing the fiber orientation is only as good as the data and rules it is based on. We consider the method as a feasible way because we will able to obtain a high quality of model as long as the quality of the derived rules is maintained.

Techniques such as using MR diffusion image in obtaining fiber orientation have been investigated in [25][31]. These techniques helps in obtaining the fiber orientation that is more precise, and as such maps better to the actual situation of a patient-specific heart.

Nevertheless, due to the huge computational power needed and large memory required as well as cost and time complexity, we concentrate only on the developpment of the LV model. A more accurate model can be obtained by enhancing the fiber structures from epicardium to endocardium. Our long term goal is able to construct a virtual patient-specified heart in real time, when there are significant advances in computer power, algorithm design and interactive graphics.

The heart valves will also be incorporated into the model as it contributes significantly in obtaining a realistic virtual heart model. The fiber orientation of further tissue types and the electrical cardiac excitation propagation will be identified and assigned in fiber parameters to obtain more accurate cardiac simulation results.

5. Conclusion

We had presented a methodology to develop a patient-specific virtual human heart by using a non-invasive approach. We believe that this is the first attempt in human heart modelling that incorporates rule based fiber reconstruction mechanism with Immersed Boundary Method. The initial simulation results are promising. This virtual model of the heart will be useful for the study of cardiovascular physiology such as hypertension and myocardial ischemia. It will also be a tool for determining the cause and prevention of heart attacks.

APPENDIX A

Here, we further explain on the algorithm for sorting the horizontal fulcra points into a 'ring' manner (as shown in Figure 8b).

Let ε ($\varepsilon \equiv \varepsilon_d$, d=1,2,3) as a centre point by considering all the key points that belonged to a specified set, $H_{t,h}$. Note that W_m represents $W_{t,h,m}$ when for a specified t and h.

Thus, the value of ε is obtained via:

$$\varepsilon_x = \left(\max_{1 \le m \le M_{t,h}} (W_{m,x}) + \min_{1 \le m \le M_{t,h}} (W_{m,x}) \right) / 2$$

and

$$\varepsilon_y = \left(\max_{1 \le m \le M_{t,h}} (W_{m,y}) + \min_{1 \le m \le M_{t,h}} (W_{m,y}) \right) / 2 \qquad \text{(A1)}$$

where x=1 and y=2.

Let Δw_n as the difference of z dimension between 2 adjacent key points in set $H_{t,h}$. Thus, Δw_n is given as

$$\Delta w_n = W_{n+1,z} - W_{n,z} \tag{A2}$$

for $1 \leq n \leq M_{t,h} - 1$.

Since $\Delta w_1 \approx \Delta w_2 \approx \ldots \approx \Delta w_{M_{t,h}-1}$, let

$$\varepsilon_z = \frac{\sum_{m=1}^{M_{t,h}} W_{m,z}}{M_{t,h}} \tag{A3}$$

Define $W^*_m \equiv W^*_{m,d}$, as an affine set of W_m where only the z dimension value of all the points in set W^* are assigned with ε_z.

Let **b** as the vector formed by the points ε and W^*_1 while $\mathbf{t}_m$ is the vector formed by the point ε and W^*_m for $1 \leq m \leq M_{t,h}$. Denote Ω_m, where $0 \leq \Omega_m \leq 2\pi$, represents the angle formed between vector **b** and $\mathbf{t}_m$. Thus, a list of angle values will be obtained by applying:

$$\Omega_m = \Omega\left(\cos^{-1}\left(\frac{\mathbf{b} \bullet \mathbf{t}_m}{|\mathbf{b}||\mathbf{t}_m|}\right)\right) \tag{A4}$$

for $1 \leq m \leq M_{t,h}$ where

$$\Omega(x) = 2\pi - x \forall (\mathbf{b} \times \mathbf{t}_m) < 0 \tag{A5}$$

So, the index of the points in W_m is arranged in such a way that their correspondent Ω_m values are sorted in acceding order. We will finally obtain an 'ring' arranged of W points, which the points sequence in the specified $H_{t,h}$ set is mapped between $\{W_1, W_2, \ldots, W_{M_{t,h}-1}, W_{M_{t,h}}\}$ and $\Omega_1 \leq \Omega_2 \leq \ldots \leq \Omega_{M_{t,h}-1} \leq \Omega_{M_{t,h}}$ respectively.

References

1. Chorin AJ. Numerical solution of the Navier-Stokes equations. Math. Comp 22: 745-762, 1968.
2. Chorin AJ. On the convergence of discrete approximations to the Navier-Stokes equations. Math. Comp 23: 341-353, 1969.
3. Cootes TF, Taylor CJ and Cooper DH. Active Shape Models - Their Training and Application. Computer Vision and Image Understanding 61(1): 38-59, January 1995.
4. David Eyre. Software for Immersed Boundary & Interface Simulation: Background on immersed boundary method [online]. Dept. Of. Math, Univ. of. Utah at Utah, US. http://www.math.utah.edu/ IBIS/ man/node6.html [19 Jun 1998].
5. Hansen DE, Daughters GT, Alderman EL, Ingels NB and Miller DC. Torsional deformation of the left ventricular midwall in human hearts with intramyocardial markers: regional heterogeneity and sensitivity to the inotropic effects of abrupt rate changes. Circ Res 5 (62): 941-52, 1988.
6. Hunter PJ, Pullan AJ and Smaill BH. Modeling Total Heart Function. Annual Review of Biomedical Engineering, Vol. 5: Pages 147-177, Aug 2003.
7. Hunter PJ, Smaill BH, Poul MF. Nielsen and LeGrice IJ. Computational Biology of the Heart, chapter: A Mathematical Model of Cardiac Anatomy. John Wiley & Sons Ltd, p.171-215, 1997.
8. Kovacs SJ, McQueen DM and Peskin CS. Modeling cardiac fluid dynamics and diastolic function. Phil. Trans. R. Soc. Lond. A. 359, 1299-1314.

9. Lai MC and Peskin CS. An immersed boundary method with formal second order accuracy and reduce numerical viscosity. J. Comput. Phys 160: 705-719, 2000.
10. LeGrice IJ, Smaill BH, Chai LZ, Edgar SG, Gavin JB and Hunter PJ. Laminar structure of the heart: Ventricular myocyte arrangement and connective tissue architecture in the dog. Am J. Phys 269: H571-H582, 1995, Heart Circ Physiol 38.
11. Mayo AA and Peskin CS. An implicit numerical method for fluid dynamics problems with immersed elastic boundaries. In Fluid Dynamics in Biology: Proc. AMS-IMS-SIAM Joint Summer Research Conf. Biofluiddynamics, Vol. 141 of Contemporary Mathematics, AMS, pp. 261-277.
12. McCracken M. and Peskin CS. A vortex method for blood flow through heart valves. J. Comput. Phys 35: 183-205, 1980.
13. McQueen DM and Peskin CS. A general method for the computer simulation of biological systems interacting with fluids. In Biological Fluid Dynamics: Proceedings of a meeting held at the University of Leeds, UK, edited by Ellington CP and Pedley TJ, 4-8 July, 1994.
14. McQueen DM and Peskin CS. A general method for the computer simulation of biological systems interacting with fluids. Vol. 49 of Symposia of the Society for Experimental Biology, Cambridge, UK, pp. 265-276, 1995.
15. McQueen DM, Peskin CS and Zhu L. The immersed boundary method for incompressible fluid-structure interaction. In Proceedings of the first M.I.T. Conference on Computational Fluid and Solid Mechanics, edited by Bathe KJ, Oxford, UK: Elsevier Ltd Vol. 1: pp.26-30. June 12-15, 2001.
16. McQueen DM. and Peskin CS. A three-dimensional computational method for blood flow in the heart: (II) contractile fibers. J. Comput. Phys 82: 289-297, 1989.
17. McQueen DM and Peskin CS. A Three-Dimensional Computer Model of the Human Heart for Studying Cardiac Fluid Dynamics. ACM Siggraph Computer Graphics, 2000, Vol. 34 No 1.
18. McQueen DM and Peskin CS. Shared-memory parallel vector implementation of the immersed boundary method for the computation of blood flow in the beating mammalian heart. Journal of Supercomputing 11(3): 213-236, 1997.
19. Nash MP and Hunter PJ. Computational mechanics of the heart. Journal of Elasticity 61(1-3): 113-141, 2000.
20. Nielsen PM, LeGrice IJ, Smaill BH and Hunter PJ. Mathematical model of geometry and fibrous structure of the heart. Am J. Physiol Heart Circ Physiol 260: H1365-H1378, 1991.
21. Peskin CS. The immersed boundary method. Acta Numerica, pp. 1-39, 2002.
22. Peskin CS. Two examples of mathematics and computing in the biological sciences: Blood flow in the heart and molecular dynamics. In: Mathematics into the Twenty-first Century, edited by Browder FE, AMS, pp. 390-115, 1992.
23. Peskin CS. Numerical analysis of blood flow in the heart. J. Comput. Phys 25: 220-252, 1977.
24. Peskin CS and McQueen DM. A three-dimensional computational method for blood flow in the heart: (I) immersed elastic fibers in a viscous incompressible fluid. J. Comput. Phys 81: 372-405, 1989.
25. Sachse FB, Henriquez C, Seemann G, Riedel C, Werner CD, Penland RC, Davis B and Hsu E. Modeling of fiber orientation in the ventricular myocardium with MR diffusion imaging. In Proc. Computers in Cardiology, Vol. 28, 617–620, 2001.
26. Sachse FB, Schulte RF, C. D. Werner CD and Dössel O. A model based approach to assignment of myocardial fibre orientation. In Proc. Computers in Cardiology, pp.145-148, 1999.
27. Schulte RF, Sachse FB, Werner CD and Dössel O. Rule based assignment of myocardial sheet orientation. Biomedizinische Technik 45(2): 97-102, 2000.
28. Stockie JM and Sheldon IG. Simulating the Motion of Flexible Pulp Fibres Using the Immersed Boundary Method. Journal of Computational Physics, Volume 147, Issue 1, Pages 147-165, 20 November 1998.
29. Thomas CE. The muscular architecture of the ventricles of hog and dog hearts. Am. J. Anat 101: 17-57, 1957.

30. Winslow RK, Scollan DF, Greenstein JL, Yung CK, Baumgartner W, Bhanot JG, Gresh DL and Rogowitz BE. Mapping, modeling, and visualization exploration of structure-function relationships in the heart. Deep computing for the life sciences Volume 40, Number 2, 2001.
31. Zhukov L and Barr AH. Heart-Muscle Fiber Reconstruction from Diffusion Tensor MRI. In 14th IEEE Visualization Conference, Oct 2003.
32. J. S.-S. Tang, A. W.-C. Liew and H. Yan, Human Face Animation Based on Video Analysis, with Applications to Mobile Entertainment, Journal of Mobile Multimedia vol 1, no 2: 133-148, 2005
33. T. Serif, G. Ghinea and A.O. Frank, Visualizing Pain Data for Wheelchair Users: A Ubiquitous Approach, Journal of Mobile Multimedia vol 1, no 2: 161-177, 2005.
34. Xiaohua Hu, Il-Yeol Song, Hyoil Han, Illhoi Yoo, Ann A. Prestrud, Murray F. Brennan, Ari D. Brooks, Temporal rule induction for clinical outcome analysis, Intl J of Business Intelligence and Data Mining, vol 1, no 1: 122-136, 2005
35. Maja Hadzic, Elizabeth Chang, Medical ontologies to support human disease research and control, Intl J of Web and Grid Services, vol 1, no 2: 139-150, 2005
36. Xiaohua Hu, Xuheng Xu, Mining novel connections from online biomedical text databases using semantic query expansion and semantic-relationship pruning, Intl J of Web and Grid Services, vol 1, no 2: 222-239, 2005