Advances in Computational Vision
and Medical Image Processing

Computational Methods in Applied Sciences

Volume 13

Series Editor

E. Oñate
International Center for Numerical Methods in Engineering (CIMNE)
Technical University of Catalunya (UPC)
Edificio C-1, Campus Norte UPC
Gran Capitán, s/n
08034 Barcelona, Spain
onate@cimne.upc.edu
www.cimne.com

João Manuel R.S. Tavares · R.M. Natal Jorge
Editors

Advances in Computational Vision and Medical Image Processing

Methods and Applications

🦉 Springer

Editors

João Manuel R.S. Tavares
University of Porto (UP)
Faculty of Engineering (FEUP)
Institute of Mechanical Engineering
and Industrial Management (INEGI)
Rua Dr. Roberto Frias, s/n
4200-465 Porto
Portugal
tavares@fe.up.pt

R.M. Natal Jorge
University of Porto (UP)
Faculty of Engineering (FEUP)
Institute of Mechanical
Engineering (IDMEC)
Rua Dr. Roberto Frias, s/n
4200-465 Porto
Portugal
rnatal@fe.up.pt

springer.com

Preface

Computational methodologies of signal processing and imaging analysis, namely considering 2D and 3D images, are commonly used in different applications of the human society. For example, Computational Vision systems are progressively used for surveillance tasks, traffic analysis, recognition process, inspection purposes, human-machine interfaces, 3D vision and deformation analysis.

One of the main characteristics of the Computational Vision domain is its inter-multidisciplinary. In fact, in this domain, methodologies of several more fundamental sciences, such as Informatics, Mathematics, Statistics, Psychology, Mechanics and Physics are usually used. Besides this inter-multidisciplinary characteristic, one of the main reasons that contributes for the continually effort done in this domain of the human knowledge is the number of applications in the medical area. For instance, it is possible to consider the use of statistical or physical procedures on medical images in order to model the represented structures. This modeling can have different goals, for example: shape reconstruction, segmentation, registration, behavior interpretation and simulation, motion and deformation analysis, virtual reality, computer-assisted therapy or tissue characterization.

The main objective of the ECCOMAS Thematic Conferences on Computational Vision and Medical Image Processing (VIPimage) is to promote a comprehensive forum for discussion on the recent advances in the related fields trying to identify widespread areas of potential collaboration between researchers of different sciences.

The present book contains the extended versions of sixteen papers selected from the works presented in the first ECCOMAS thematic conference on Computational Vision and Medical Image processing (VIPimage 2007), held in Faculty of Engineering of University of Porto, Portugal. It collects the state-of-the-art on the subject of Computational Vision and Medical Image processing contributing for the development of these areas of knowledge.

The Editors would like to take this opportunity to thank to the European Community on Computational Methods in Applied Sciences, the Portuguese Association

of Theoretical, Applied and Computational Mechanics, the University of Porto, all sponsors, all members of the International Scientific Committee, and to all Invited Lecturers and Authors.

Faculty of Engineering,   *João Manuel R.S. Tavares*
University of Porto, Portugal   *R. M. Natal Jorge*

Contents

Chapter 1
Modeling Cardiovascular Anatomy
from Patient-Specific Imaging

Chandrajit Bajaj and Samrat Goswami

1.1 Introduction

The importance of modern imaging techniques for capturing detailed structural information of a biological system cannot be understated. Unfortunately images do not reveal the "full functional story" and a spatially realistic computer model is often necessary for a comprehensive understanding of the complicated structural and physiological properties of the biological system's entities under investigation [1]. Deeper insights into structure-to-function relationships of different entities is achieved via finite element simulations of the modeled biomedical process. A 3D (three dimensional) finite element meshed computer model of the biological system is therefore a first step to perform such simulations.

The behavioral attributes of a biological entity or the physiological interaction between different participating components of a biological system are often modeled mathematically via a coupled set of differential and integral equations, and quite often numerically evaluated using finite element (or boundary element) simulations. To further emphasize the premise of cardiac modeling from imaging data, we state a few computational biomedical modeling and simulation examples: 3D computational modeling of the human heart for a quantitative analysis of cyclical electrical conductance on the heart membrane [2–6]; the biomechanical properties (stress-strain, elasticity) of the heart ventricular walls [7–12]; 3D modeling and simulation of pulsatile blood flow through human arteries/veins for vascular by-pass surgery pre-planning on a patient specific basis [13–18]. A finite element decomposition of the geometric domain, capturing the detailed spatial features that can be gleaned from the imaging, is therefore the essential first step toward performing the necessary numerical simulations [19–22].

C. Bajaj and S. Goswami
Computational Visualization Center, Institute of Computational Engineering and Sciences,
University of Texas, Austin, Texas 78712
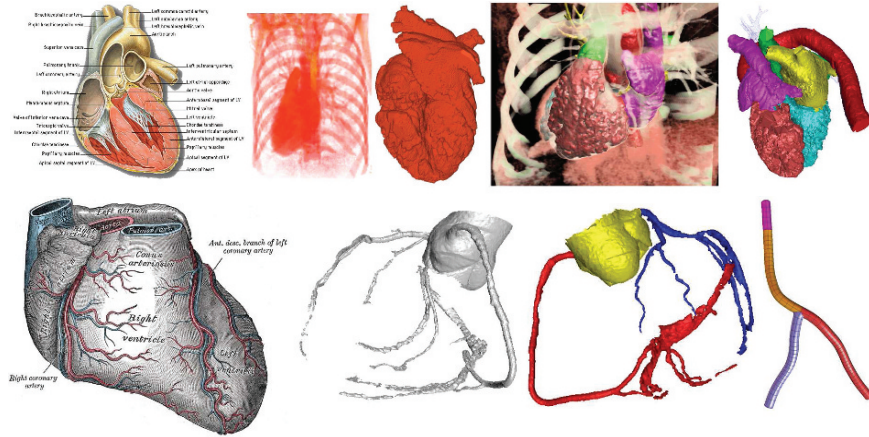e-mail: bajaj@cs.utexas.edu; samrat@ices.utexas.edu

Fig. 1.1 Cardiovascular Modeling from Imaging Data. Top row illustrates the modeling of patient heart from imaging data. From left to right: an illustration of internal structure of human heart (Courtesy [23]), volume visualization of a typical Computed Tomography (CT) reconstructed imaging volume, the extracted geometric model of a patient heart from the rest of the thoracic bone and tissue and the annotated (multi-colored) segmentation of the same into biologically meaningful components, namely Aorta, Pulmonary Artery, Left and Right Ventricle and Atrium (colored differently). Bottom row illustrates the modeling of coronary artery. From left to right: an illustration of annotated coronary arterial tree (Courtesy http://en.wikipedia.org/wiki/Coronary_circulation), a geometric model extracted from CT volumetric imaging, segmentation into left and right subtree (colored red and blue), and the NURBS model of a small portion that was used for Isogeometric analysis of blood flow in [24]

Modeling of human vasculature from three-dimensional (3D) Computed Tomography (CT) images of the thorax is a critical step for computer-aided diagnosis (CAD) in disease domains such as lung nodules [25], coronary artery disease [26], and pulmonary embolism (PE) [27]. Even though there are many published approaches, the problem is still unsolved. Survey of various techniques on this topic can be found in [28,29] (Fig. 1.1). Thoracic CT angiography (CTA) imaging is often performed for patients suspected of having a PE that is defined as a thrombus (or a clot of blood) [30]. Therefore, in order to detect pulmonary embolism, a suitable model of vasculature distnguishing between arterial and venous blood vessel trees is a crucial step.

In subsequent subsections we highlight the computational pipeline, the main algorithmic components and a few descriptive results of our Cardio Vascular Modeling from Imaging software.

1.2 Data Processing

The Imaging-to-Modeling software system for cardiovascular data employs both Image Processing and Geometry processing functionalities to produce a suitable linear or higher order meshed model of the anatomy. Figure 1.2 describes the data-flow layout. We describe the major algorithmic components of each of the processing
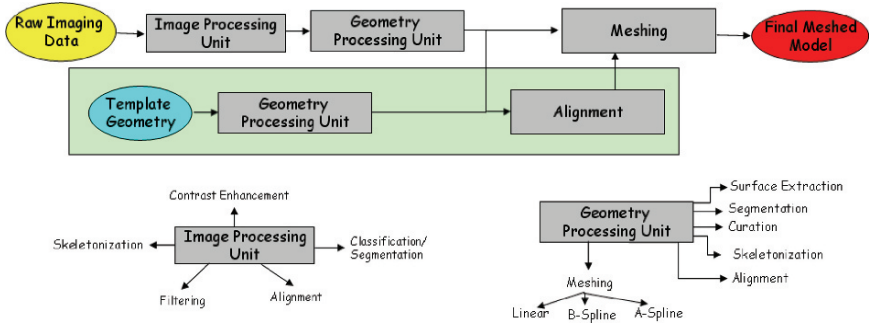
Fig. 1.2 Data Flow of Cardio-Vascular Modeling from 3D Imaging Data: There are two major data processing algorithmic modules – Image Processing and Geometry Processing. The Image Processing module consists of sub-modules for *Contrast Enhancement*, *Classification/Segmentation*, *Filtering*, *Skeletonization* and *Alignment*. The Geometry Processing module consists of *Surface Extraction*, *Curation*, *Segmentation*, *Skeletonization*, *Alignment* and *Meshing*, which itself is further subdivided into linear and higher order boundary and finite element mesh generation components. The 3D CT Imaging data is first passed through the Image Processing module for improvement of image quality which is then processed by the Geometry Processing module for extraction of a clean geometry annotated with the present features. Finally the clean geometry is converted to a linear or higher-order mesh. Occasionally, to deal with incomplete or low quality imaging data, a twin data processing pipeline is employed where a template geometry is processed to extract vital geometric information which is then passed to robustly model patient specific anatomical model from low-quality imaging data

modules in Sections 1.2.1 and 1.2.2. The reader must note that, the modules are selectively used depending on the nature and quality of the input imaging data (Section 1.3).

1.2.1 Image Processing

(A) Contrast Enhancement:

The three dimensional intensity data often possesses a low contrast between structural features and the background, thereby making further processing all the more difficult. Image contrast enhancement is a process used to improve the image quality for better visual appearance for subsequent operations. The most commonly used methods utilize global contrast manipulation based on global [31, 32] or local histogram equalization [31–34], *retinex model* [35, 36] and wavelet decomposition [37, 38].

We have developed a fast method for image contrast enhancement [39] based on a localized version of the classical contrast manipulation [31, 32]. In this method, we design an adaptive transfer function for each individual voxel, based on the intensities in a suitable local neighborhood. First, we compute the local statistics (local average, local minimum, and local maximum) for each voxel using a fast propagation scheme [40, 41]. Then a transfer function is
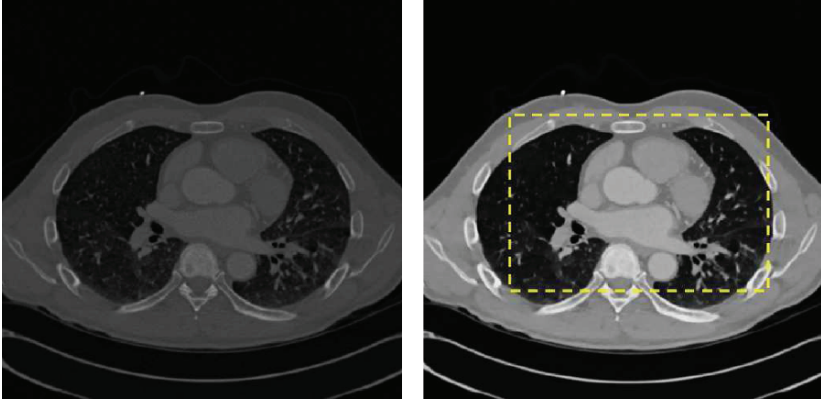
Fig. 1.3 The performance of the contrast enhancement algorithm is shown for one slice of the CTA data

designed based upon the calculated local statistics. Various linear or nonlinear functions can be used here to stretch the contrast profile. We build a transfer function which consists of two pieces: a convex curve in the dark-intensity range and a concave curve in the bright-intensity range. The overall function is C^1 continuous. Finally, we map the intensity of each voxel to a new one using the calculated transfer function. The performance of this algorithm is shown in Fig. 1.3.

(B) Filtering:

The input images are often contaminated with noise and are therefore need to be filtered. Traditional image filters include Gaussian filtering, median filtering, and frequency domain filtering [31]. Compared to these, anisotropic filters are preferred as they tend to preserve the features better. Bilateral filtering [42–45] is a straightforward extension of Gaussian filtering by simply multiplying an additional term in the weighting function. Partial differential equation (PDE) based techniques, known as anisotropic geometric diffusion, have also been studied [46, 47]. Another popular technique for anisotropic filtering is by wavelet transformation [48]. By carefully designing the filter, one can smooth image noise while maintaining the sharpness of the edges in an image [49]. Finally, the development of nonlinear median-based filters in recent years has also produced promising results [50, 51]. Among the aforementioned techniques, two methods for noise reduction have been suggested for tomographic data sets, namely wavelet filtering [52] and non-linear anisotropic diffusion [53] (Fig. 1.4).

Our approach, utilizing a bilateral pre-filtering coupled with an evolution driven anisotropic geometric diffusion PDE (partial differential equation), has shown significant results in enhancing the features of intensity maps. The PDE model is :

$$\partial_t \phi - \|\nabla \phi\| div \left(D^\sigma \frac{\nabla \phi}{\|\nabla \phi\|} \right) = 0$$
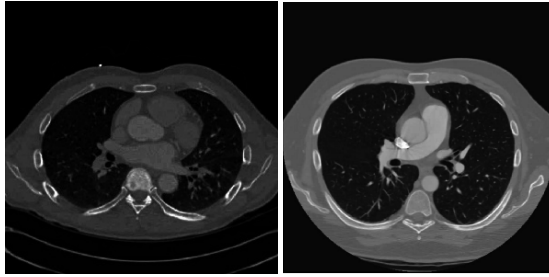
Fig. 1.4 The performance of the bilaterally pre-filtered (left) and anistropic diffusion filtered (right) algorithm is shown for one slice of the CTA data

The efficacy of our method is based on a careful selection of the anisotropic diffusion tensor D^σ based on estimates of the normal and two principal curvatures and curvature directions of a feature isosurface (level-set) in three dimensions [54–56]. The diffusivities along the three independent directions of the feature isosurface are determined by the local second order variation of the intensity function at each voxel. In order to estimate continuous first and second order partial derivatives, a tricubic B-spline basis is used to locally approximate the original intensity.

(C) Classification/Segmentation:

Voxel Classification:

The Fuzzy C-Means (FCM) algorithm [57] and the Expectation Maximization (EM) algorithm [58] have been used for soft clustering in data-mining and image classification. Pham et al. proposed an Adaptive Fuzzy C-Means (AFCM) algorithm to classify inhomogeneous medical images and volume datasets [59, 60]. Ahmed et al. proposed a bias corrected FCM algorithm to compensate inhomogeneities of images of volume datasets [61, 62]. Each of these algorithms minimizes an objective function through iterative methods. Gopal et al. proposed a maximum likelihood estimate algorithm with a Spatially Variant Finite Mixture model (SVFMM) for image classification [63]. Laidlaw suggested a partial-volume Bayesian classification algorithm based on Bayes theorem [64].

Our goal of 3D map segmentation is to partition the map into a number of connected regions of interest. We have compared and implemented several classification algorithms [64–67]. We first locate the seed points by gradient vector diffusion [68]. We then compute the min-max range of every seed point's neighbors and cluster the seed points to belong to the same region if the min-max ranges overlap. We then apply GVF snake [69, 70] to cluster the voxels falling into separate regions. In addition, the contour spectrum is used to identify the number of materials in the image and is also used to select critical isovalues based on volume fraction of the material [71]. We have also developed a multi-dimensional signature based voxel classification
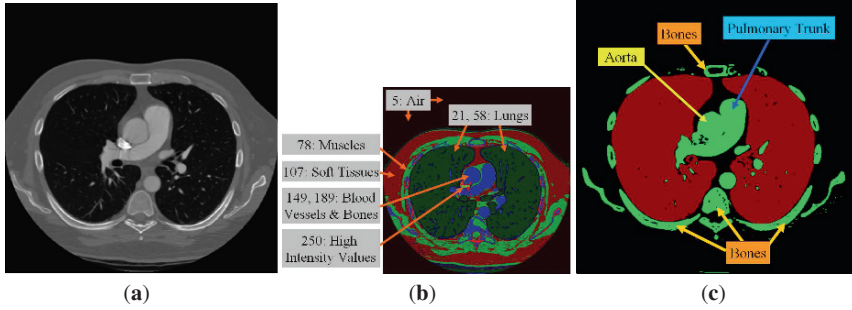
Fig. 1.5 Voxel Classification: a single slice of CTA image of a patient (a) is classified to tag voxels belonging to different anatomical regions (b). The intensity values are also marked (b). The final classified voxelized image is shown with most important regions (c)

scheme [72] appropriate for medical imaging data. Figure 1.5 shows the results of the classification on a single slice of the patient scan where the voxels have been classified into the background, lungs and vasculature.

Segmentation via Fast Marching Method:

Segmentation is a way to electronically dissect the significant biological components, and thereby obtain a clear view into the machinery's architectural organization [73]. Segmentation is usually carried out either manually [74–78] or semi-automatically [79, 80]. Current efforts on the decomposition still largely rely on manual work with an assistance of a graphical user interface [81,82]. Manual segmentation can be tedious and often subjective [76,83]. Automated segmentation is still recognized as one of the hardest tasks in the field of image processing although various techniques have been proposed for automated or semi-automated segmentation. Commonly used methods include segmentation based on edge detection, region growing and/or region merging, active curve/surface motion and model based segmentation. In particular, two techniques were discussed in details in the electron tomography community. One is called *water-shed immersion method* [79] and the other is based on normalized graph cut and eigenvector analysis [80].

In [84–86] we adopted a variant of the fast marching method [87–89]. In this method a contour is initialized from a pre-chosen seed point, and the contour is allowed to grow until a certain stopping condition is reached. Traditionally this method is designed for a single object segmentation. We present an approach based on an idea of "re-initialization" by simply regarding and classifying the critical points as seeds. Every seed initiates one contour and all contours start to grow simultaneously and independently. We further classify the critical points into clusters and merge the growing contours which are initiated by the critical points in the same cluster. This multi-label idea has been used elsewhere (e.g. [90]), but the detection and classification of seeds are different in our approach. Figure 1.6 shows the process of segmentation, namely the seed point
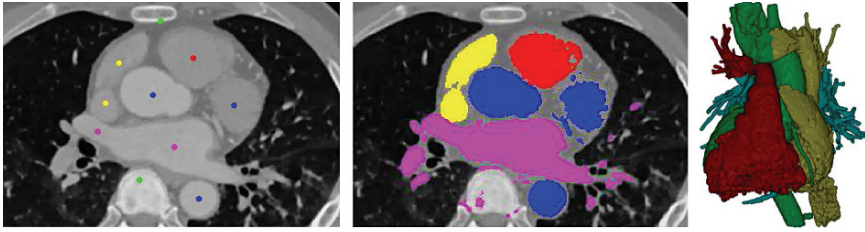
Fig. 1.6 Segmentation via Fast Marching Method: leftmost subfigure shows a single slice of the CTA image of a patient along with manually placed seeds for each of the main compartments of the human heart to be identified. The middle subfigure shows the segmentation on a single slice colored by segmented regions. The rightmost subfigure shows the three dimensional segmented model of heart via isosurface rendering. The different colors represent left and right ventricle and aorta

classification and region growing on a single slice of the image followed by the final segmented three dimensional model of human heart from patient imaging data.

(D) Skeleton Extraction:

Extraction of skeletal description often leads to a complexity-reducing better understanding of the image as it amplifies lower dimensional key features present in the data. Image based skeletonization algorithms are abound in the field of image processing. The previous efforts on this topic by other authors can be categorized into three approaches – one based on isotropic diffusion, governed by a set of linear PDEs [91,92], one using scale-space theory [93–95] and one based on pseudo-distance map [96].

We have developed two distinct approaches to extract skeletons from imaging data which offer robustness and efficiency. First approach is based on the critical point structure of the imaging data. We first compute the gradient vector field at every voxel of the imaging data. Because of the noise, the vector field is somewhat arbitrary and do not carry much useful information. To bring out the underlying structure, we then apply gradient vector diffusion (GVF). The resulting vector field is then analyzed to detect the critical points and these critical points are joined to form a skeletal structure of the foreground of the image. The details of the process is given in [56].

The second approach for skeletonization starts with an isosurface and the distance function induced by it [28]. It then places a sequence of inner medial balls in a greedy fashion, capturing as much inner volume as possible. A neighborhood graph, based on the intersection pattern of these inner medial balls, is then constructed which provides the one dimensional skeletal structure. Figure 1.7 shows an example result of this algorithm applied on the CTA data of human heart.

(E) Flexible Alignment:

Image registration is a commonly employed to flexibly match two different instances of a biological structure. In the context of cardiovascular modeling,
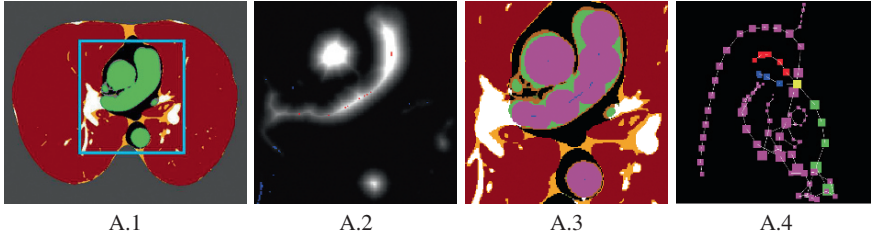
A.1	A.2	A.3	A.4

Fig. 1.7 Skeletonization: a slice of the imaging data, its distance map and the extracted skeletal graph using the algorithm in [28] are shown

the problem is to fit one image of the anatomy in question with its counterpart observed at a different time, or imaged from a different patient. Based on the transformation model that is to be applied on one instance (source) to fit the other (target), image resgistration is primarily of two types – Affine and Elastic/Deformable.

In case of affine registration, the relationship between the source and target image is established via a set of transformation parameters, and then those parameters are estimated by minimizing a quadratic error function. Algorithms under such approaches can be found in [97–99]. We have developed an algorithm in [100] which exploits the non-equispaced Fourier transformation techniques [101, 102] to speed up the affine image registration process.

The task is far more challenging when deformation of one image needs to be taken into account to properly fit it into the other image. For a nice survey on affine and deformable image registration algorithms, see [103]. However this survey is old, and since its publication many other algorithms have been published. Bajcsy and Kovacic modeled the elastic image registration problem by the deformation of elastic plates [104]. Christensen et al. considered the deforming image to be embedded in a viscous fluid whose motion is governed by Navier-Stokes equation [105]. Based on a similar viscous fluid registration scheme, Yanovsky et al. designed a new energy function introducing Jacobian maps [106] and this method was shown to perform better than [105] in terms of converegence and stability. There are other level-set based methods, e.g. by Clarenz et al. [107], and via edge matching technique by Mumford and Shah [108].

1.2.2 Geometry Processing

(A) Surface Extraction:
Geometry extraction from three dimensional volumetric data is a primary step toward further analysis. There are several approaches for accomplishing this task.

Contouring:

Isosurfacing is a popular method to extract surface geometry from scalar volume containing intensity values of the scanned anatomy. There are typically two types of contouring method frequently used in the literature – Primal Contouring and Dual Contouring. The most widely used primal contouring technique is the Marching Cubes Method [109] which extracts the geometry in a piecewise fashion by visiting every voxel of the volume data. There are several improvements of this technique that has been reported since the first appearance of the algorithm [110]. Dual contouring technique is similar to primal contouring in a sense that it also extracts the isosurface in a piecewise fashion. However it samples every voxel, instead of every edge of the voxel, to better approximate possible sharp features in the extracted geometry. We have experimented with both techniques for extracting a geometry from Cardiac CT data and we have seen that they perform similarly without rendering any significant advantage of any technique over the other. Figure 1.8 shows sample isocontours extracted from CTA imaging data of human heart.

Other than isocontouring, one can also apply level set based methods where a seed is grown to capture the boundary of a region in the image based on the intensity values. In literature, such technique is commonly known as *snake* [69, 70]. Note, this is similar to the image segmentation technique described earlier.

Point Cloud Reconstruction:

Both of these approaches are very sensitive to the noise present in the data and especially isosurfacing techniques suffer when the imaging data is inhomogeneous. To circumvent these problems we adopt a third approach which is based on scattered data interpolation. After performing image segmentation on
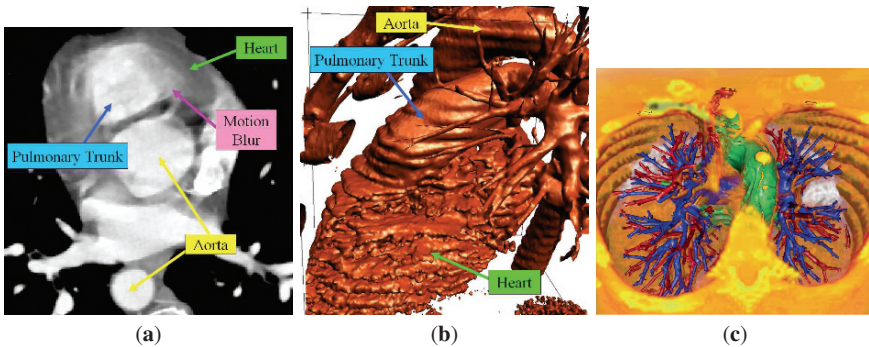

(a) (b) (c)

Fig. 1.8 Contouring: a single annotated slice of human cardio-vasculature and the geometry extracted from the imaging data via contouring (isosurfacing) are shown from left to right. In the rightmost subfigure, the heart of the patient is shown in green while the blood vessel tree is color coded by red and blue depending on if a branch is an artery or a vein

the CT image, we obtain a set of voxels belonging to every region of interest. We then apply the point based reconstruction technique to extract the geometry from the cloud of boundary voxels.

The point based reconstruction technique has been researched extensively in the last decade. We refer the readers to some recent surveys for prior work in this area, e.g. [111]. We have adopted two recent techniques for our purpose of reconstruction – *TightCocone* and *RobustCocone*. *TightCocone* algorithm by Dey and Goswami [112] reconstructs a watertight triangulated surface from possibly undersampled input point cloud. A variant of this algorithm, called *RobustCocone*, was also developed in 2004. This algorithm is particularly suitable for noisy data. In our case, we often encounter noise in the segmented image even after applying image segmentation techniques, and to tackle such cases, we use *RobustCocone* for a reconstruction of the geometry. Figure 1.9 shows the results of surface reconstruction on the pointsets sampling compartments of human heart.

Surface reconstruction from scattered data has also been approached using variational approach. These techniques typically formulate an energy function based on the input data points and try extract a surface that minimizes that energy. Such approach was first advocated by Zhao et al. in [113] who formulated the energy function as the integral of the distance function weighted by the area element of the input set of primitives for which a surface needs to be fit. Then they evolved an initial guess using a convection based approach.

We adopt an approach based on higher order level set spline (HLS) method. Given a non-negative energy function $g(\mathbf{x})$, the surface Γ is defined to be the one that minimizes the energy function $E(\Gamma) = \int_{\Gamma} g(\mathbf{x})d\mathbf{x} + \varepsilon \int_{\Gamma} h(\mathbf{x},\mathbf{n})d\mathbf{x}$. Given a input set of points, one then formulates this energy using the distance function and evolves an initial approximation to guide the evolution so that the resulting deformation minimizes the energy. Details of this method can be found in [114].

(B) Curation/Filtering:

The cardiovascular geometry extracted from imaging data typically has topological anomalies, namely small components, spurious noisy features etc. Therefore a careful investigation and subsequent removal of the spurious features present in the data is essential. Following are different scenarios.
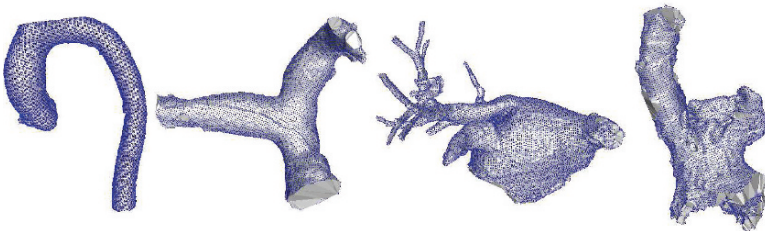

Fig. 1.9 Surface Reconstruction from Point Cloud Data: major components of the human heart are reconstructed using the voxels surrounding the boundaries of the individual regions

Regularization:

Geometry from volume data is often reconstructed via image segmentation. However the set of voxels segregated from the imaging data does not always conform with the true surface topology. As a result, we encounter subsets of voxels which do not sample a two dimensional manifold. Therefore it is important to recognize the dimension of the underlying space of the voxels marked by the segmentation process and remove the spurious ones. To perform this task, we use the technique described in [115]. A similar Voronoi-based approach was also reported in [116]. Following [115] we first construct a k-neighborhood graph. Then for every point we collect the neighboring points and perform Principal Component Analysis (PCA) on that subset. The eigenvalues of the covariance matrix determines the underlying dimension of the manifold. More precisely if all the eigenvalues are almost equal, the voxel is inside the segmented region and the point samples a three dimensional manifold. If two eigenvalues are almost equal and one is much smaller than the other two, the voxel lies on the boundary surface of the segmented region and is a true candidate for subsequent geometry reconstruction. Finally if two of the eigenvalues are much smaller than the third one, then the voxel samples a dangling one dimensional strand and such voxels must be removed.

Volumetric Feature Quantification:

Given a set of points P sampling the entire shape, possibly contaminated with topological artifacts like small connected components and thin tunnels, we synthesize a distance function h_P which assigns every point in $\mathbb{R}^3$ the distance to the nearest sample point in P. There are four types of critical points of h_P, namely maxima, index 2 saddles, index 1 saddles and minima. It was shown that these critical points can be detected efficiently using the duality of Voronoi and Delaunay diagram of the original pointset P [117]. It was further shown that the stable manifolds of the maxima interior and exterior to the shape contains useful information connecting to the primal and complementary feature space of the shape [118, 119]. Figure 1.10 shows how the bone, ribs and thin blood vessels are removed via volumetric feature quantification process since they are not essential for creating a suitable model of human heart.

(C) Segmentation:

Based on the critical points of the distance function induced by the input geometry, we perform the geometry segmentation as follows. The detail of the algorithm is given in [118].

The geometric shape is given by a set of points P sampling the shape. The *feature* of the shape is then defined in terms of the stable manifold of the maxima of the distance function h_P. The maxima are first computed by identifying the Voronoi vertices which lie inside their dual tetrahedra. Applying a Delaunay-based reconstruction technique on the pointset, one can further classify the tetrahedra holding the maxima into *inside* or *outside*. For our purpose we use only the *interior* maxima. We compute the stable manifold of these
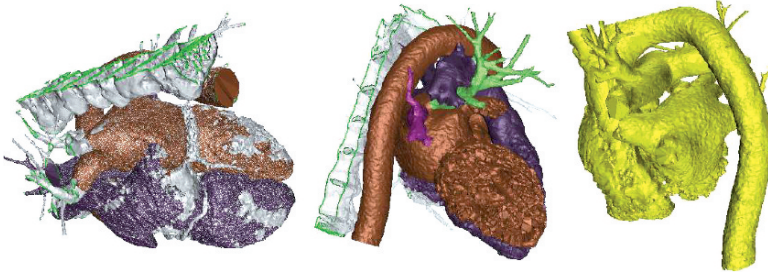
Fig. 1.10 Geometry Curation: geometry of human heart extracted from the imaging data, which is cluttered with bone and other unnecessary parts, is cleaned up using the curation process
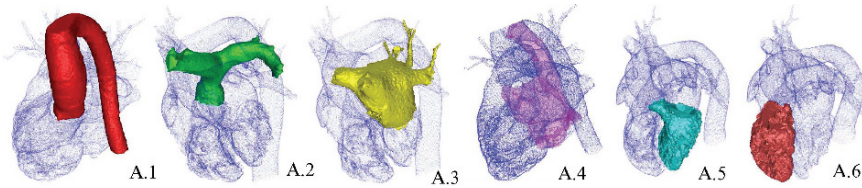

Fig. 1.11 Geometry Segmentation: different components – Aorta (A.1), Pulmonary Artery (A.2), Left Atrium (A.3), Right Atrium (A.4), Left Ventricle (A.5) and Right Ventricle (A.6) are extracted from the point sample of the whole heart

maxima using the algorithm in [118]. The adjacent stable manifolds are then merged if the generating maxima have almost same value of h_P as measured by a parameter δ. Figure 1.11 shows the performance of this algorithm on the cardiovascular geometries. In this figure, the six main components of a human heart, namely aorta, pulmonary artery, left and right atrium and ventricle are segmented out from a set of points sampling the boundary of heart.

(D) Skeletonization:

Computing skeletons of a geometric shape is a research issue that has been around for a long time. Medial axis cite (Blum) is considered a standard skeletal description of a shape and there are algorithms [120, 121] and publicly available software to compute the Medial Axis Transform (MAT) of a shape from its pointsample [122]. However, medial axis is composed of planar (two dimensional) and linear (one dimensional) parts. In order to compute such dimension-dependent decomposition of the medial axis, as well as pruning away hairy branches from the medial axis one requires some extra gadgets which we describe in the next paragraph. There are some previous works which focussed on computing a linear skeleton of an arbitrary shape. Some of these are topological thinning [123], distance field based methods [124–127], potential field based methods [128], thinning via medial geodesic function [129] and others [130–132]. Cornea et al. [133] give a comprehensive survey of these techniques.
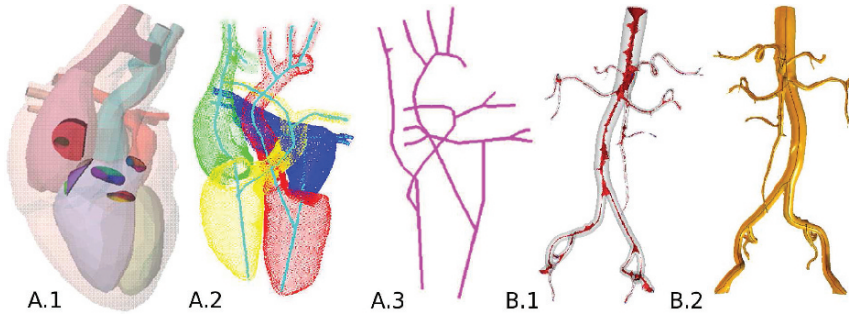
A.1   A.2   A.3   B.1   B.2

Fig. 1.12 Skeletonization: (A). Skeleton of the template geometry is extracted. (B). The skeletal structure of the abdominal aorta is extracted. Note, compared to the medial axis (B.1), the unstable manifold of index 2 saddles is a much cleaner one dimensional skeletal structure

Once a suitable description of the geometry is obtained either by reconstruction or by contouring, the distance function based approach is used to compute a skeletal structure of the shape. As for segmentation, the critical points of the function h_P is computed for a set P of input points sampling the shape. The index 1 and index 2 saddle points are then detected using the Voronoi-Delaunay duality [117]. To generate the skeletal structure, we then compute the unstable manifold of these critical points. The unstable manifold of an index 2 saddle point (U_2) is one dimensional and the unstable manifold of an index 1 saddle point (U_1) is two dimensional. Moreover, every U_1 is bounded by some U_2's. The details of the computation of U_1 and U_2 are given in [134]. Figure 1.12 shows two instances where the skeletal structures have been constructed using this method.

(E) Alignment:

Alignment of two similar but not identical geometric objects is a difficult problem. There are relatively few papers in the geometric modeling community that address this problem. Recently, an interesting technique was reported by Eckstein et al. [135] where generalized surface flow was used for non-rigid alignment of a template geometry into the patient data. The authors design an energy function based on pseudo-Hausdorff distance between the two geometries and evolve the template geometry to fit the patient geometry following the gradient of the energy function.

We experimented with a different approach for non-rigid fitting of the segments of the template heart into the patient data in order to inherit the correct topological structure from the template geometry as well as retrieve the missing information in the patient data. We construct a skeletal description of different parts of the template geometry as described earlier. Every segment is then described as a union of balls centered at the one dimensional skeleton following a popular approach due to [136]. A mass-spring network is then built where each ball's mass is proportional to its radius and the spring constant is taken to be constant. The normal mode analysis (NMA), which is a popular way to

depict the vibrational nature of a molecule [137], is then applied to the network which produces a spectrum of possible deformed shapes. We choose one deformed shape from the spectrum that best fits the patient geometry.

(F) Quality Meshing:

The final goal of this Imaging to Modeling software system is to produce suitably discretized meshed model of the imaged biological entity. The task of meshing is primarily divided into two parts – Boundary Element and Finite Element. Each of Boundary Element and Finite Element meshing again has three sub-parts.

Boundary Elements:

Boundary Element meshing refers to the meshing of the surface geometry. Depending on the smoothness and shape of every patch forming the surface mesh.

(a) Triangle/Quadrilateral Elements: Given a CT image, the task is to compute a triangulated or quadrangulated discretization of the boundary of the cardio-vascular anatomy. In a sense the task is similar to contouring the zero-set of the original intensity function or a distance function induced by the reconstructed geometry. There are numerous algorithms to accomplish such tasks as mentioned in the previous subsection. However in this step our goal is to build a boundary element mesh of superior quality than what is typically output by the contouring routines. The mesh quality metrics are described in [138].

We approach the problem in two steps. First we apply the dual contouring method proposed by Ju et al. [139] and apply geometric flow to improve the quality of the surface mesh. Figure 1.13 shows the performance of the boundary element meshing algorithm.

(b) B-Spline Elements: Building B-Spline model for free-form geometric objects has been an active area of research in the past. Given a triangulated or quadrilateral surface mesh, there have been numerous approaches
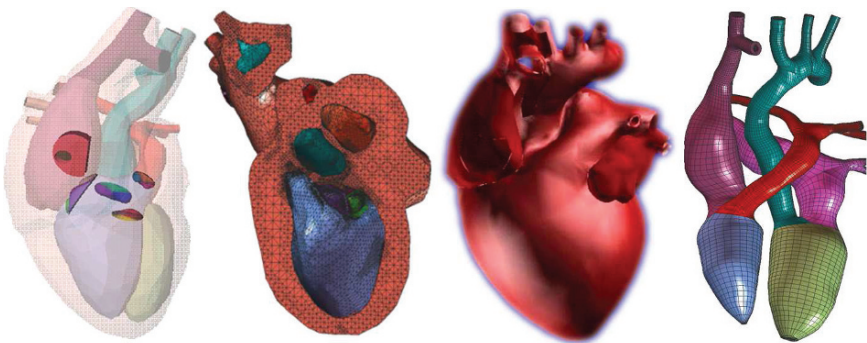

Fig. 1.13 Meshing: triangle-tetrahedral, A-patch, NURBS meshes

to build a smooth network of B-Spline patches to model the underlying surface [140]. Subdivision based methods, namely Catmull-Clark [141], Doo-Sabin [142, 143] and Loop subdivision [144] for producing quadratic and cubic B-Spline models from triangulated and quadrilateral control meshes have also gained popularity due to their simplicity. However in most of the cases, it is not straight-forward how to generate the initial control mesh for free-form shapes especially when the input is a set of scattered set of points or a set of voxels representing the boundary of a segmented region inside a three dimensional volume.

Until recently there were very works that dealt with the problem of building a control quadrilateral mesh from a free-form geometry of arbitrary topology [145, 146]. Recently there have been substantial research works in the computer graphics community that tackled the problem of quadrangulating a surface mesh following the intrinsic anisotropy of the geometry [147–149]. Following these approaches, one can build a quadrilateral base surface mesh on which any of the standard subdivision scheme can be applied to build the desired B-Spline model.

(c) A-Patch Elements: The linear meshes do not always provide the adequate smoothness for them to be effectively used further. This is the reason a higher order meshed description of the geometry is often desired. Bajaj et al. presented a solution for this problem in 1995 where they devised a scheme that takes a triangulated surface mesh and builds a higher order (cubic) patch complex to describe the same surface. Under certain conditions, they also showed that these patches meet in a C^1 continuous manner. They called this patched description of a two dimensional geometry an algebraic surface patch or an *A-patch* [150].

The construction of A-patches proceeds as follows. First a triangle mesh is created that linearly approximates the given geometry. A tetrahedral scaffolding is then built around it so that the triangle mesh lies inside it. By assigning a suitable weight at every node of the scaffold, a polynomial inside each tetrahedron is constructed in such a way that its zero set satisfies certain properties that makes the higher order patch inside the tetrahedron free of singularity. Moreover the pacthes are derivative-continuous across the tetrahedra. The patch complex then provides a higher order smooth description of the geometry. Details of this method can be found in [150]. This method of construction of surface A-patches has also been generalized to hexahedral elements, as well as to the multi-resolution construction of shell finite elements [151, 152]. Figure 1.13 shows a higher order geometric model of human heart.

3D Finite Elements:

Finite Element meshing refers to the techniques of producing the discretization of the volume enclosed by the surface geometry.

(a) Tetrahedral/Hexahedral Elements: There have been prior works done on building volumetric (finite element) tetrahedral and hexahedral meshes from imaging data. For a detailed survey on the prior works see [153].

We have developed two algorithms for such purposes – TetMesh and HexMesh. The approach is as follows. First we extract a correct boundary using the boundary element meshing. For tetrahedral mesh generation we use the triangulated boundary and for hexahedral mesh generation we use the quadrangulated boundary meshing scheme as described above. Next, we design a series of templates to build a solid tetrahedral or hexahedral mesh from that conforms with the boundary mesh. The details about the templates are given in [138, 154]. As with the boundary element mesh, we further improve the quality of the finite element mesh by applying geometric flow [155]. Figure 1.13 shows the tetrahedral mesh cut-away for a template human heart.

(b) B-Spline Elements: As the most highly developed and widely utilized technique, NURBS (Non Uniform Rational B-Splines) [156–158] has evolved into an essential tool for a semi-analytical representation of geometric entities. Sometimes NURBS solid models are taken as input for finite element mesh generation [159]. Anderson et al. proposed a fast generation of NURBS surfaces from polygonal mesh models of human anatomy [160]. An enhanced algorithm was developed for NURBS evaluation and utilization in grid generation [161]. In isogeometric analysis [162], NURBS basis functions are used to construct the exact geometry, as well as the corresponding solution space.

We have developed a skeleton-based approach for building NURBS model of vasculature. Using the skeletonization approach described in the earlier paragraph, a one dimensional polylinear skeletal structure is first extracted from the tubular geometry of vasculature. Then we design a set of templates that builds a hexahedral mesh around the skeleton. The details about the templates can be found in [24]. The hexahedral mesh is used as the control mesh for further NURBS mesh generation. Figure 1.13 shows the NURBS model of the inner blood volume of human heart.

(c) Shell A-Patch Elements: Shell structures appear frequently in biological entities. The muscle wall of heart and blood vessels are perfect examples of such surfaces. It is desired to model such *fat* surfaces with desired smoothness. Bajaj et al. presented algorithms that model smooth shell structures using shell A-patch finite elements [151, 152].

The algorithm takes a pair of triangle mesh as input where the correspondence between the triangle meshes is implicit. It then decimates both the meshes simultaneously to build a coarser representation and occasionally merges the adjacent triangles to form quadrilaterals wherever possible. Using the correspondence between the triangles and quadrilaterals in the two-sheeted surface, the interval volume is then filled with 3-sided (triangular) and 4-sided (quadrilateral) prisms. A C^1 piecewise trivariate function is then constructed over this collection of prisms. The range of the

function varies from -1 to 1. For any scalar $\alpha \in [-1,1]$, the 0-set thus gives the higher order approximation of the intermediate surface within the shell element. When α equals -1 or 1, the resulting patch gives a higher order approximation of the inner and outer boundaries of the shell.

1.3 Implementation Results

We exhibit the implementation results of our image and geometry processing algorithms on modeling the Heart and Vasculature (Coronary Artery, Pulmonary Arterry, Abdominal Aorta and Thoracic Aorta).

1.3.1 Heart

We have experimented with images of patient hearts. For illustration of the performance of the algorithms, we have selected two datasets one of which is a low quality image whereas the other is of relatively better quality.

The first heart dataset is of dimension $512 \times 512 \times 432$ and the spacing in x, y, z directions are respectively 0.390625, 0.390625, 0.3 mm. We first applied the contrast enhancement to the original image and then applied the fast marching based segmentation on the contrast-enhanced image to separate the subvolumes corresponding to the aorta, pulmonary artery, right atrium and left atrium. Because of the poor quality of the image, the ventricles could not be recovered well. The result of the image processing on this dataset is shown in Fig. 1.14B. After the four components of the patient heart are segmented from the volume, we took the boundary voxels of each of these four regions and applied surface reconstruction from scattered points to build the initial triangulated geometric models. As one can see these models have a reasonable amount of spurious parts as well as some missing information that could not be retrieved from the patient data. We analyzed each of the recovered geometries using the critical point structure of the distance function described in Section 1.2.2. Using curation and geometry segmentation, we identified the portions which most prominently correspond to the portions of a template heart model while pruning away the undesired portions. Figure 1.14C shows the relevant portions that have high correlation with the corresponding portions of the template geometry. Parallel to the processing of the patient heart, we also performed geometric analysis of the template heart model. We construct the one dimensional skeletal structure of all the six components of the template, as well as we segmented the template into aorta, pulmonary artery, right and left ventricle and atrium (Fig. 1.14D). From there we could draw a correspondence between the segmented portions of the patient heart with the template heart (Fig. 1.14E).

The second dataset was of better quality in terms of contrast and noise present. We first extracted the geometry via isocontouring. However using a single isovalue could not entirely serve the purpose as it also extracts the surrounding vasculature
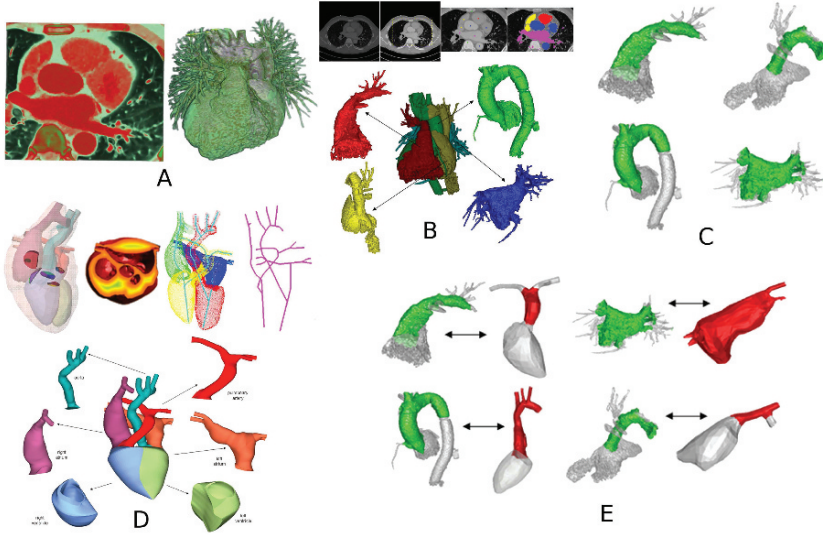
Fig. 1.14 Result: heart I. (A) One slice of the original data and isocontour-enhanced volume rendering of the input. (B) Due to relatively poor quality, the imaging data is first passed through the image processing unit that enhances the contrast and segments the volume into major components using fast marching based image segmentation. (C) The extracted geometry from the segmented image is then curated to keep only the (green) portions which can be matched with the corresponding portions from the template heart model (D, E)

as shown in Fig. 1.15A. We therefore resorted on curation to extract just the geometry of heart by pruning away the surrounding thin blood vessels using curation (Fig. 1.15B). After this step we were left with the geometry of heart which we further segmented using the stable manifold of the maxima of the distance function induced by the geometry. The segmentation step was crucial as it was able to separate the six main components of the patient heart, namely aorta, pulmonary artery, right and left ventricle and atrium as shown in Fig. 1.15C. We were then able to draw the correspondence between each of the segmented parts with the corresponding ones from the template geometry as shown in Fig. 1.15D.

1.3.2 Vasculature

1.3.2.1 Pulmonary Artery

The primary objective behind modeling pulmonary artery was to detect pulmonary embolism (PE) automatically [29]. This required an initial artery-vein separation from the CT scans of the vasculature which was performed directly from the input imaging data by the image skeletonization technique described in Section 1.2.1. The skeletons are traced from the end of the branches and traversed upward toward
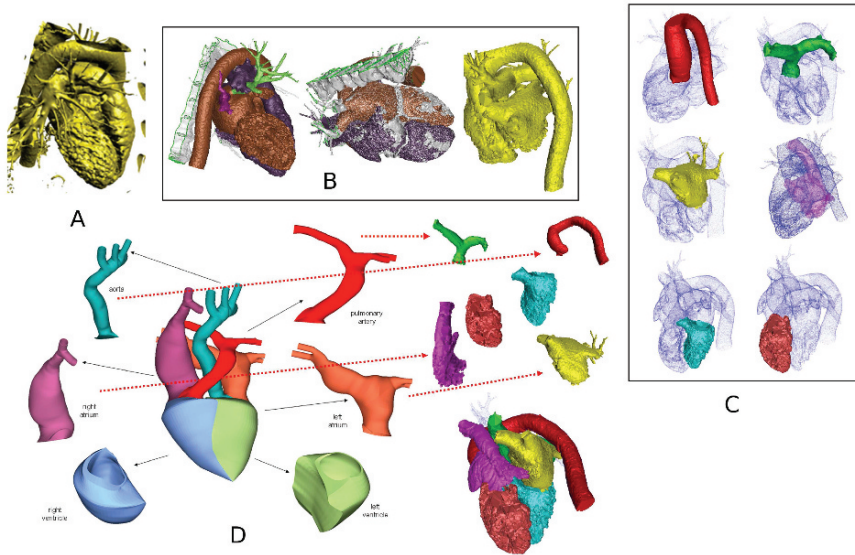
Fig. 1.15 Result: heart II. Starting with an isocontour extracted from the raw imaging data (A), the model is first curated to extract a clean geometry without the nearby bone structure or the thin blood vessels which are otherwise irrelevant for the specific task of modeling heart (B). This initial model is then geometrically segmented into six main components (C) and further a correspondence is drawn with the segmented template. The processing of the template heart model, namely, skeletonization and annotated segmentation is shown in (D)

the heart. Once the trace reached the patient heart through a series of disambiguation, as needed because of the poor image quality, one of the branches was tagged arterial while the other is tagged venous. At the same time, the rest of the volume was classified using the voxel classification technique as described in Section 1.2.1 (Fig. 1.16b, c). This led to a complete characterization of the CT data into the major components along with the artery and vein separated (Fig. 1.16d–f).

1.3.2.2 Abdominal Aorta

Starting with the CT scan of the abdominal section of the patient, we first extracted the geometry using an isovalue that best captures the geometry of the abdominal aorta along with the surrounding bone structure and other anatomical parts which are however irrelevant for the modeling of the aorta itself. To separate the aorta, we performed geometry segmentation, using the stable manifold of the maxima of the distance function. The segmented aorta is shown in green in Fig. 1.17a, b. We then performed geometry skeletonization on this segmented geometry and as a result a one-dimensional skeletal structure was produced (Fig. 1.17c, d). This skeleton was then used for building a swept hexahedral volume that best represented the geometry of the aorta, as well as served the purpose of a control mesh which was then used to build a solid NURBS model of the abdominal aorta (Fig. 1.17e, f).
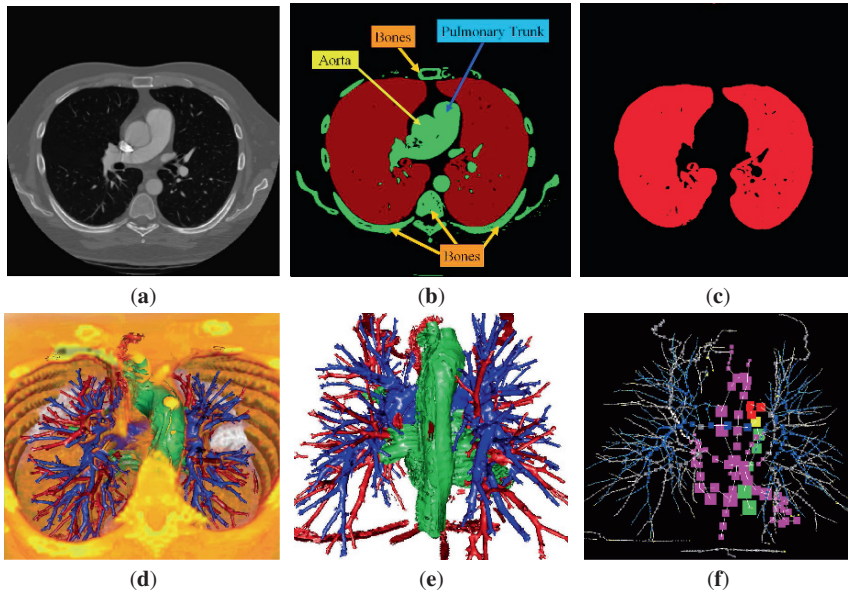
Fig. 1.16 Result: pulmonary artery. The imaging data (one slice is shown in (a)), is first classified to identify the voxels belonging to lungs and the vasculature (b, c). Using the skeletonization technique from Section 1.2.1, the arterial (red) and venous (blue) trees are detected. The vasculature and heart (green) are shown in (d–f) superimposed with the volume rendering of the CTA image
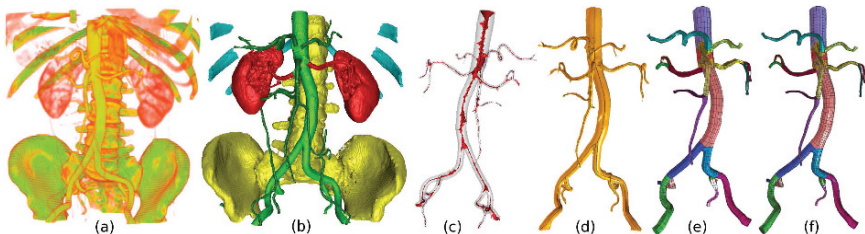

Fig. 1.17 Result: abdominal aorta. Volume rendering of the original imaging data is shown in (a). An isocontour is extracted from the imaging data from which the abdominal aorta (green) is segmented (b). The medial axis and the linear skeleton extracted from it are shown in (c, d) from which an initial control polyhedron (e) and the final NURBS model (f) are created

1.3.2.3 Thoracic Aorta

Starting with the scan of the patient heart, we first segmented out and extracted the geometry of the thoracic aorta via isocontouring. We then performed the skeletonization of this geometry in order to obtain an one-dimensional polylinear skeletal (Fig. 1.18 A.1). An external pathway is added to the skeletal structure in order to simulate the LVAD used in time of open-heart surgery. Using the skeleton as a sweeping path, we then built a hexahedral control mesh that best represented the
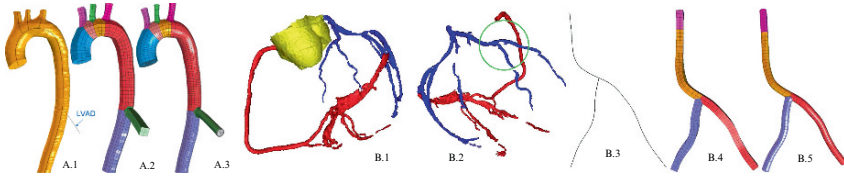
Fig. 1.18 Result: (A) Thoracic aorta. Skeleton is extracted from the geometric model (A.1). An artificial pathway is added to simulate the LVAD. The hexahedral control mesh and the resulting NURBS model are shown in (A.2, 3). (B) Coronary artery. The coronary artery is segmented into left and right subtrees (B.1, 2). NURBS mesh of a small portion of the tree is then built using the extracted skeleton (B.3–5)

aorta along with its inner volume (Fig. 1.18 A.2). This control mesh was then used to construct a solid NURBS mesh of the thoracic aorta as shown in Fig. 1.18 A.3.

1.3.2.4 Coronary Artery

Starting with a CT scan of the patient heart, first the coronary artery was extracted as shown in Fig. 1.18B.1. The coronary artery has two main branches – right and left, which were then geometrically segmented from the whole vascular structure. Figure 1.18B.2 shows both the vasculature trees colored in red and blue. We then computed the skeletal structure for each of these substructures using the skeletonization method described earlier (Fig. 1.18B.3) and subsequently a NURBS mesh model was constructed using the swept volume technique as described in the meshing subsection earlier (Fig. 1.18B.4, B.5).

1.4 Conclusions

We have developed a comprehensive software collection (http://cvcweb.ices.utexas.edu/software/) of data processing tools (http://cvcweb.ices.utexas.edu/cvc/projects/medx/pipeline.php) that can be utilized in producing patient specific, and spatially realistic, linear and curved, boundary and finite element models of human cardio-vasculature. Such a combination of geometry extraction and geometric modeling are the necessary enablers for quantitative and interrogative querying, analysis and visualization. These boundary and finite elements are additionally useful for a variety of physiological, bio-chemical modeling and simulation of normal or diseased conditions. They are also useful for virtual surgical training, treatment planning and drug or dosage delivery. Current imaging modalities we have successfully processed through our software include Computed Tomography (CT) and Magnetic Resonance Imaging (MRI), and their blood perfused variations for cardio-vasculature modeling and micro-CT for osteoporotic bone modeling.

References

1. Winslow, R.L., Scollan, D.F., Greenstein, J.L., Yung, C.K., Jr., W.B., Bhanot, G., Gresh, D.L., Rogowitz, B.E.: Mapping, modeling, and visual exploration of structure-function relationships in the heart. Deep Computing for the Life Sciences **40**(2) (2001)
2. Luo, C., Rudy, Y.: A model of the ventricular cardiac action potential: Depolarization, repolarization and their interaction. Circulation Research **68**(6) (1991) 1501–1526
3. Luo, C., Rudy, Y.: A dynamic model of the cardiac ventricular action potential: I. simulations of ionic currents and concentration changes. Circulation Research **74**(6) (1994) 1071–1096
4. Hodgkin, A.L., Huxley, A.F.: A quantitative description of membrane current and its application to conduction and excitation in nerve. Journal of Physiology **117**(1952) 500–544
5. Hille, B.: Ionic Channels of Excitable Membranes, 2nd edn. Sinauer Associates, Sunderland, MA (1992)
6. Winslow, R.L., Scollan, D.F., Holmes, A., Yung, C.K., Zhang, J., Jafri, M.S.: Electrophysiological modeling of cardiac ventricular function: from cell to organ. Annual Reviews in Biomedical Engineering **2**(2000) 119–155
7. Vetter, F., McCulloch, A., Rogers, J.: A finite element model of passive mechanics and electrical propagation in the rabbit ventricles. Computers in Cardiology (1998) 705–708
8. Rogers, J.M., McCulloch, A.D.: A collocation-galerkin finite element model of cardiac action potential propagation. IEEE Transactions on Biomedical Engineering **41**(1994) 743–757
9. Rudy, Y., Plonsey, R.: A comparison of volume conductor and source geometry effects on body surface and epicardial potentials. Circulation Research **46**(1980) 283–291
10. Costa, K.D., Hunter, P.J., Wayne, J.S., Waldmann, L.K., Guccione, J.M., McCulloch, A.D.: A three-dimensional finite element method for large elastic deformations of ventricular myocardium: Ii - prolate spheroidal coordinates. Journal of Biomedical Engineering **118**(4) (1996) 464–472
11. Hunter, P., McCulloch, A., Nielsen, P., Smaill, B.: A finite element model of passive ventricular mechanics. ASMS BED **9**(1988) 387–397
12. Sachse, F.B.: Computational Cardiology: Modeling of Anatomy, Electrophysiology, and Mechanics. LNCS 2966. Springer, Berlin, Heidelberg, New York (2004)
13. Taylor, C., Hughes, T., Zarins, C.: Finite element modeling of blood flow in arteries. Computer Methods in Applied Mechanics and Engineering **158**(1–2) (1998) 155–196
14. Taylor, C., Hughes, T., Zarins, C.: Finite element modeling of 3-dimensional pulsatile flow in the abdominal aorta: relevance to atherosclerosis. Annals of Biomedical Engineering **26**(6) (1998) 1–13
15. Taylor, C., Hughes, T., Zarins, C.: Effect of exercise on hemodynamic conditions in the abdominal aorta. Journal of Vascular Surgery **29**(1999) 1077–89
16. Sahni, O.: Adaptive procedure for efficient blood-flow simulations. PhD thesis, RPI (2005)
17. Sahni, O., Mueller, J., Jansen, K.E., Shephard, M.S., Taylor, C.A.: Efficient anisotropic adaptive discretization of the cardiovascular system. Technical report, RPI (2005)
18. Yin, L., Luo, X., Shephard, M.S.: Identifying and meshing thin sections of 3-d curved domains. Technical report, RPI (2005)