EG UK Theory and Practice of Computer Graphics (2008)
Ik Soo Lim, Wen Tang (Editors)

Time-varying Image Data Visualization Framework for Application in Cardiac Catheterization Procedures

Y. L. Ma[1], K. S. Rhode[1], A. P. King[1], G. Gao[2], P. Chinchapatnam[2], T. Schaeffter[1], R. Razavi[1] and K. Saetzler[3]

[1]Imaging Sciences, King's College London, SE1 7EH, UK
[2]Centre for Medical Image Computing, University College London, M13 9PL, UK
[3]System Biology Research group, University of Ulster, Coleraine, BT52 1SA, UK

Abstract

Visualization plays an important role in image guided surgery. This paper presents a real-time 3D motion visualization method where pre-computed meshes of the beating heart are synchronized with and overlaid onto live X-ray images. This provides the surgeon with a navigational aid in guiding catheters during cardiac catheterization. In order to generate time-varying meshes of the beating heart, we first acquire a time-series of images of the patient using Magnetic Resonance Imaging (MRI). The MRI heart images used for the cardiac catheterization procedures can either be contrast-enhanced by injecting a contrast agent prior to imaging or they can be unenhanced. The contrast-enhanced images can easily be segmented and binarized using a fixed grey-level threshold. In this case, we can use an adaptive Delaunay-based surface extraction algorithm for mesh generation, for which specifically developed for noisy binary image data sets. For unenhanced images, we have to choose a semi-automated segmentation approach, where a region of interest in the patient's heart is outlined manually in an intermediate slice in the 3-D MRI data set and then propagated to neighbouring slices. In a next step, the extracted snake contours are propagated in time from the first phase of the cardiac cycle to subsequent phases using multiple snake contours. In this scenario, the final mesh is generated using a serial section reconstruction algorithm. However, due to the nature of the underlying MRI images which frequently contain areas of inhomogenous contrast caused by motion and blood flow, it is difficult to generate a smooth mesh directly from the result of the previously described semi-automatic segmentation procedure. Therefore, we also introduce a contour-based mesh smoothing algorithm using a 1D Gaussian filter in order to post-process the snake contours along the series of cross-sections before reconstruction.

Keywords: Time-varying mesh, mesh denoising, image-guided surgery, augmented reality

1. Introduction

Image-guided surgery has revolutionized traditional surgical techniques by providing surgeons with a navigational aid in the form of three-dimensional (3D) images acquired using modalities such as magnetic resonance imaging (MRI), computerised tomography (CT), and ultrasound (US). During cardiac catheterization, one of the challenging tasks is to insert a flexible and hollow tube, the catheter, starting at the patient's groin or arm artery/vein and to guide this to particular target regions within the heart. X-ray fluoroscopy is currently the modality of choice for procedure guidance due to its high spatial and temporal resolution allowing good visualisation of interventional devices such as guide-wires and catheters. However, X-ray imaging offers very little soft tissue contrast resulting in poor visualisation of the cardiovascular anatomy. Therefore, X-ray contrast agents are routinely and repeatedly injected to highlight target structures during the procedure. Another major disadvantage of X-ray imaging is that it uses ionising radiation, which is a potential hazard to both patients and staff. X-ray radiation is of particular importance for very long procedures such as electrophysiology (EP) procedures.

In order to reduce the amount of ionising radiation and to overcome the lack of soft tissue contrast, we have developed a system that allows structures derived from MRI to be overlaid in real-time onto X-ray fluoroscopy images [RHE*03]. An MRI-derived 3D mesh model is aligned to the X-ray image coordinate system with an accuracy of 2mm using a combination of system calibration and real-time tracking. However, motion of the patient's body and heart during the MRI imaging and catheterization procedure make this accuracy difficult to achieve. In this paper, we focus mainly on extracting cardiac motion during a single heart beat from MRI images and displaying the resulting series of meshes in our image guided surgery software. Our method can potentially also be used in medical education software, surgical planning or simulation software.

The rest of the paper is organized as follows. In section 2, we describe the semi-automated segmentation approach. Section 3 presents our Delaunay-based surface extraction method which is used to generate surface meshes from contrast-enhanced and thresholded images. Section 4 presents a contour-based mesh smoothing algorithm, while section 5 describes the surface reconstruction algorithm for serial sections. Section 6 shows how the resulting series of meshes are finally overlaid with live X-ray images. In section 7 we discuss our results.

2. Segmentation propagation

One of the most time-consuming, but important tasks is to segment the time-varying MRI heart image data which captures several phases of the cardiac cycle. Unlike high-contrast Computed Tomography (CT) image data, MRI images are generally of low contrast and often show areas of inhomogeneous contrast. This makes the development of automatic segmentation algorithms for MRI particularly difficult. However, during clinical procedures MR contrast agents are frequently injected into the target region to enhance the contrast. The enhanced images can be segmented by simply thresholding the grey-level image thus converting them into binary images. For unenhanced MRI heart images we choose a semi-automated segmentation approach, where a region of interest in the patient's heart is outlined manually in an intermediate slice in the 3-D MRI data set and then propagated to neighbouring slices using multiple B-spline snakes [JBU04]. In a next step, the extracted snake contours are propagated in time from the first phase of the cardiac cycle to subsequent phases.

3. Delaunay based surface extraction method

For real-time guidance applications in cardiac catheterization procedures, a 3D mesh is overlaid onto live X-ray images. The 3D mesh is rendered in each frame because the orientation and position of the 3D mesh relatively to live X-ray images might constantly change. The complexity of the 3D mesh will therefore directly impact on the rendering speed of the overlay image. One common solution to generate meshes from 3D image data in real-time guidance applications is to use the Marching Cubes (MC) algorithm [LC87] to extract an iso-surface and then to apply a mesh decimation algorithm [LUE01]. Although this solution can reduce the number of triangles used for the final visualisation substantially, the decimation algorithm show artefacts when approaching compression rates of 90% or higher. A reduction in the number of triangles, however, is crucial in order to be able to visualize the full heart chamber motion in real-time. To do so, at least 20 phases of the 3D heart beat image data need to be segmented and reconstructed. This means that at least 20 3D meshes need to be overlaid for each single heart beat more than 70 times per minute. Therefore, the above mentioned approach does not yield optimal real-time visualization results and needs to be further optimised. For this purpose, we have previously developed an adaptive 3D Delaunay-based surface extraction algorithm which is particularly useful for noisy, binary image data sets as generated when thresholding the contrast-enhanced MRI images [MS08]. Our approach generates an adaptive triangulation that catches a similar level of detail when compared to the MC algorithm, but only uses 10% or less triangles.

4. Contour-based mesh smoothing algorithm

Due to contrast inhomogeneities in MRI images, the contours generated by the automatic segmentation methods discussed above still contain noise leading to uneven surface reconstructions. This noise across different slices could even become worse if the contours are manually edited or corrected. In order to still be able to generate reasonably smooth meshes, we designed a 1-D Gaussian contour filter that smoothes geometrically the resulting contours in contrast to the widely used Gaussian filter used in image or signal processing. Our filter takes three groups of contours as input. The first group contains the contours in the current slice. The second one spans the contours in the adjacent, upper slice if the upper slice exists. The third group includes the contours in the adjacent, lower slice if it exists. In a discrete form, 1-D Gaussian kernel can be expressed in a 1-D convolution mask. The total weight of this mask should add to 1.0:

0.25	0.5	0.25	G_Mask

This 1-D Gaussian convolution mask is now used to smooth the positions of control nodes of all contours in three neighbouring slices. The Gaussian filter is typically not energy preserving. While this is less noticeable in the image data, in meshes, this manifests itself as a shrinkage of the mesh. Since the Gaussian contour filter can be applied iteratively, the volume of the target object could therefore significantly shrink. To preserve the volume of the object, we use the approach suggested by Desbrun et al. [DMS*99]. After each iteration step the mesh will have a new volume V_n. As we want to scale it back to its original volume V_0, we multiply all vertices by a simple scale factor $\beta = (V_0 / V_n)^{1/2}$. This forces the volume to go back to its original value. The complete algorithm is shown as follows:

Algorithm 1. Gaussion_contour_ filter
for *each image slice in 3D image data*
  for *each control node* P_c *in a group of contours*
    if *upper slice exists* **then**
      Compute the nearest point P_u *in a group of contours in upper slice.*
    else
      return
    endif
    if *lower slice exists* **then**
      Compute the nearest point P_l *in a group of contours in lower slice.*
    else
      return
    endif
    $P_n = 0.25 * P_u + 0.5 * P_c + 0.25 * P_l$
    $P_n = P_n * (V_0 / V_n)^{1/2}$
    Update contours with new control node P_n
  end*{End of the loop for}*
end*{End of the loop for}*
End *of Algorithm 1.*

Normally, after applying the Gaussian contour filter less than 5 times, we obtain a significantly smoothed surface (figure 1).

(a) (b) (c)

Figure 1: Panel (a) shows the original surface extracted from unsmoothed contour data. Panel (b) shows the reconstructed surface after applying the 1-D Gaussian filter two times. Panel (c) shows the surface after applying our filter four times.

5. Surface reconstruction from serial sections

The smoothed contours can now be used to generate a final surface mesh. In contrast to volumetric and voxel-based reconstruction methods as the MC or our Delaunay-based subdivision approaches, we can find a second class of reconstruction algorithms which has been optimized to generate 3-D meshes from 2-D cross sections (for an overview see [ES99]). These methods are computationally efficient as they usually operate in 2-D, but can also cope with several topological changes (e.g. bifurcations, birth of holes) that occur particularly frequently when voxel sizes are very anisotropic. In particular, we adopted a 3-D version of the reconstruction method originally published by Boissonnat in 1988 [B88], which has already previously been used for serial section reconstructions from electron microscopical serial sections. In brief, our method creates a Delaunay tetrahedrization for each pair of consecutive sections using the contour points on each section and labels the tetrahedrons as being part or outside the set of contours. The final surface is then generated by extracting all triangles that are at the interface between inside and outside tetrahedrons (see also [MS08]). The result mesh can be seen in Figure 1.

6. Overlay with live X-ray images

As outlined before, we generate surface meshes by either using our Delaunay-based adaptive surface extraction algorithm for binary images stemming from contrast-enhanced input data or we use a serial section reconstruction algorithm which uses the semi-automatically traced and smoothed B-spline curves as input. In total, we generate about 20 3D meshes and load them into the visualization pipeline. The phase mesh display is now synchronized with the ECG of the patient. Using the ECG delay information inside the image header of the DICOM file, we can display the correct phase mesh by using the ECG R wave peak as trigger. The peak of the ECG R wave is produced in the period of ventricular contraction when the volume of heart chamber becomes smallest. Figure 2 illustrates this mechanism.

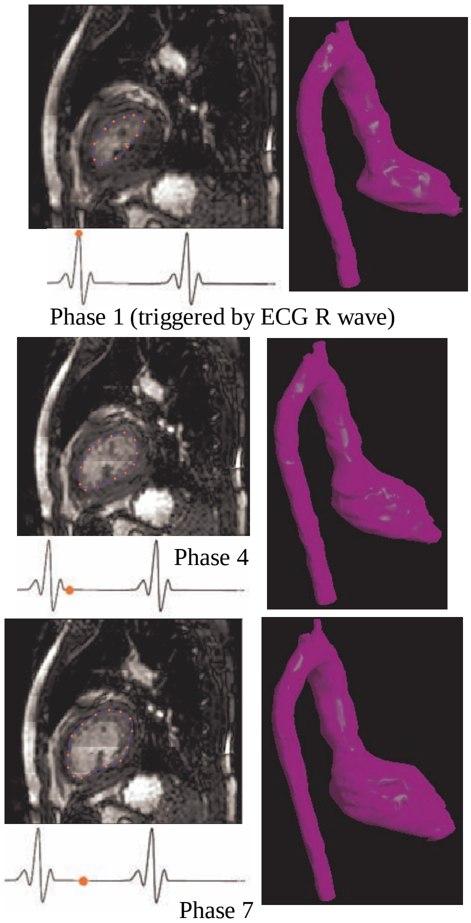
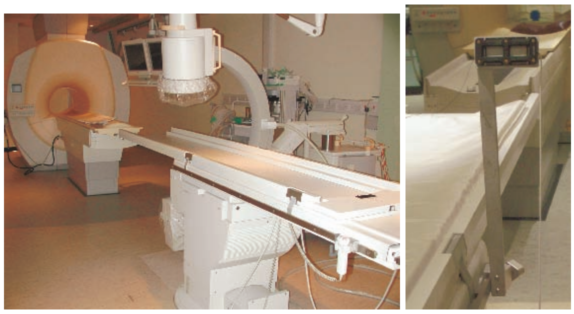


Figure 2: This shows the series of 3-D meshes which are displayed in synchronization with the ECG R wave.



(a) (b)

Figure 3: The hybrid X-ray and MR image systems (XMR). Panel (a) shows the location of the MR scanner and X-Ray system. Optical tracking sensors are attached to X-ray table (see panel (b)).

In our cardiac catheterization procedure room, we have a hybrid X-ray and MR image system (XMR) [RHE*03], which can provide MR images as well as live X-ray images. The room has two distinct zones: the MRI zone, in which the magnetic field is above 0.5mT, and the non-MRI zone, which comprises the rest of the room. As shown in figure 3a, the MR scanner is located in the MRI zone (far side) and the X-ray system is positioned in the non-MRI zone (bright floor colour). The patient can be easily moved between the two systems using a specially modified sliding MR table top that docks with and transfers patients to a specially modified X-ray table. In order to register the MR 3D volume with live X-ray images, it is necessary to define a common coordinate system between MR and X-ray images. The common coordinate system we used is the coordinate system defined by the X-ray table. From the MRI image space to the MR table space, we can use the geometric information provided by the header of the medical image format: DICOM. From the MR table to the X-ray table, there is only a translation, which is provided by the MR scanner itself. From X-ray table space to X-ray image space, the optical tracking of the X-ray table and the C-arm provide sufficient information to accomplish the task.

After registration of the real-time X-ray images with the 3D heart mesh model that is derived from MR images, we can not only guide catheter placement in real-time but also significantly reduce the radiation dose patients receive. Because the X-ray images by themselves provide very little soft tissue contrast and the heart is poorly visible in X-ray images, a visualization framework has been developed to provide the cardiologist with a real-time guidance system for cardiac catheterization procedures. A video frame grabber (Matrox Inc.) is used to capture the X-ray images in real-time. Then, through a shared memory block which allows file mapping, live X-ray images are sent to the visualization software. The live X-ray image is then displayed using OpenGL texture mapping capabilities. A stationary or a time-varying 3D heart mesh can now be overlaid onto the frame-grabbed X-ray image data. The positions and orientations of the 3D mesh are updated in real-time using the tracking capabilities of the X-ray image system [RHE*03]. A screenshot of this visualization framework is shown in Figure 4.

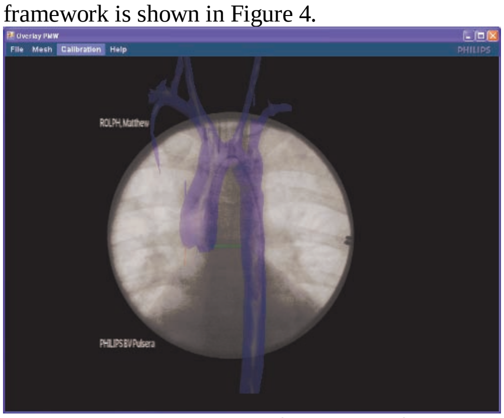


Figure 4: A Screen shot of the GUI of the visualization software showing a catheter inside the aorta.

7. Conclusion

In this paper, we have presented a real-time hybrid visualization framework that allows the simultaneous rendering of pre-computed 3D heart surface meshes onto live X-ray images that are synchronized to the actual ECG recorded for the patient. This is particularly helpful in cardiac catheterization procedures. Because we use a patient-specific 3D heart model and compensate for cardiac motion, we can achieve a high accuracy. The stationary 3D mesh overlay system is already used in clinical procedures at the St Thomas' Hospital in London and the time-varying 3D mesh overlay system is currently under clinical validation. The presented framework could also be used in medical education and surgical planning applications.

8. Acknowledgements

This work was funded by DTI technology programme grant 17352 and Philips Medical Systems, Best, the Netherlands. We are grateful to the members of the Imaging Sciences division at St Thomas' Hospital.

9. References

[Boi88] Boissonnat, J. Shape reconstruction form planar cross--sections Computer Vision, Graphics, and Image Processing, 1988, 44, 1-29.

[DMS*99] Desbrun, M., Meyer, M., Schroder, P. and Barr, A. H.: Implicit fairing of irregular meshes using diffusion and curvature flow. In Proc. of SIGGRAPH 99 (1999), 317–324.

[ES99] Eils, R. & Sätzler, K. Jähne, B.; Haußecker, H. & Geißler, P. (ed.) Shape Reconstruction from Volumetric Data Handbook of Computer Vision and Applications, Academic Press, 1999, 2, 791-815.

[JBU04] Jacob M., Blu T. and Unser M., Efficient Energies and Algorithms for Parametric Snakes, *IEEE Transactions on Image Processing*, vol. 13(9), 1231-1244, 2004.

[LC87] Lorensen W.E. and Cline H.E.: Marching Cubes: A High Resolution 3D Surface Construction Algorithm, In Proc. of SIGGRAPH '87 (1987), pp. 163-169.

[LUE01] Luebke D.P.: A Developer's Survey of Polygonal Simplification Algorithms, IEEE Computer Graphics Application, vol. 21(3), pp. 24-35, 2001.

[MS08] Ma Y.L. and Saetzler K.:, A Parallelized Surface Extraction Algorithm for Large Binary Image Data Sets Based on an Adaptive 3D Delaunay Subdivision Strategy, IEEE transaction on visualization and computer graphics, Vol 14(1), pp. 160-172, 2008.

[RHE*03] Rhode K. S., Hill D. L. G., Edwards P. J., Hipwell J., Rueckert D., Sanchez-Ortiz G., Hegde S., Rahunathan V., and Razavi R.: Registration and Tracking to Integrate X-Ray and MR Images in an XMR Facility. *IEEE Transactions on Medical Imaging*, 22(11):1369–1378, 2003.