Tagged Volume Rendering of the Heart

Daniel Mueller[1], Anthony Maeder[2], and Peter O'Shea[1]

[1] Queensland University of Technology, Brisbane, Australia
d.mueller@qut.edu.au
[2] e-Health Research Centre, CSIRO ICT Centre, Brisbane, Australia

Abstract. We present a novel system for 3-D visualisation of the heart and coronary arteries. Binary tags (generated offline) are combined with value-gradient transfer functions (specified online) allowing for interactive visualisation, while relaxing the offline segmentation criteria. The arteries are roughly segmented using a Hessian-based line filter and the pericardial cavity using a Fast Marching active contour. A comparison of different contour initialisations reveals that simple geometric shapes (such as spheres or extruded polygons) produce suitable results.

1 Introduction

It is often overlooked that segmentation is integral to volume visualisation [1]. In the case of indirect volume rendering ('surface rendering') the segmentation takes the form of isosurface extraction using Marching Cubes [2] prior to the actual rendering. In the case of direct volume rendering (DVR) the segmentation is typically achieved during the rendering process ('online') via a lookup table ('transfer function'). Various online segmentation algorithms for DVR have been discussed in the literature [3,4,5,6]. Kniss et al. [4] advocated the use of two-dimensional transfer functions dependent on pixel value and gradient magnitude, in which the user interactively specifies the function using a number of widgets. Tzeng et al. [5] proposed an approach using a machine learning algorithm (such as a support vector machine) applied to each pixel in the image.

However, such online methods suffer from a number of drawbacks. Firstly, as discussed by Hadwiger et al. [7], 2-D transfer functions can not typically distinguish between objects with similar characteristics. Secondly, online segmentation unnecessarily constrains the choice of algorithm. Take for example 3-D texture rendering: the segmentation method must be implemented on the graphics hardware in real-time, currently precluding whole classes of algorithms (including active contour methods). Tagged volume rendering addresses these shortfalls using a number of *a priori* segmented binary masks ('tags') which are assigned separate transfer functions [7,8,9].

To date much of the work regarding tagged volume rendering has fixated on *generic* matters: there has been little focus on the applicability of the methods. In contrast, this paper presents the material from an applications perspective centred around the visualisation of the heart. For diagnostic and treatment planning purposes, radiologists and surgeons require means to visualise the coronary arteries from multi-slice spiral computed tomography angiography (MS-CTA)

images. However, this task is non-trivial for various reasons: unwanted structures (such as the thoracic cage) clutter the regions of interest; and the contrast agent highlights the coronary arteries (as desired), but also portions of the ventricles, atria, aorta, and pulmonary arteries. As such, the traditional DVR approach using value-gradient transfer functions is insufficient for this application.

In this paper we apply tagged volume rendering to the problem. We utilise Hessian-based line filters for segmenting the coronary arteries [10] and propose the use of an active contour method [11] for segmenting the pericardial cavity. A suitable speed function is derived for controlling the expansion of the active contour. We experiment with various mechanisms for specifying the initial contour, and conclude that simple geometric shapes (such as spheres or extruded polygons) produce suitable results. Finally the three *a priori* segmented volumes (heart, arteries, and surrounds) are used as tags within a rendering scheme facilitating per-tag transfer functions. The proposed system is demonstrated using a number of clinically obtained *in vivo* datasets.

2   Method

The proposed system consists of two main steps: (1) the offline segmentation of binary tags, and (2) the online refinement of structures within these tags using traditional value-gradient transfer functions. This two-stage approach provides for online interactivity while promoting relaxed criteria for the offline segmentation. Furthermore, the offline segmentation can be performed on down-sampled volumes (which decreases the execution time) without affecting quality.

2.1   Tagged Volume Rendering

We implement generic direct volume rendering using 3-D texture mapping [4]. In this approach, 3-D textures are uploaded to the graphics processing unit (GPU). At the application level, the rendering system slices the uploaded 3-D textures using view-aligned polygons. This 'proxy-geometry' tri-linearly interpolates the textures at any given viewing angle. A small program ('shader') is applied to each potential pixel ('fragment'), acting as a post-interpolative transfer function.

For tagged volume rendering, a tag image is also uploaded to the GPU, the value of which switches on or off per-object transfer functions. In the original method [7], each tag represented an exact binary object to be classified (using a 1-D transfer function), resulting in the need to linearly filter tag boundaries to achieve appropriate resolution at the pixel level. Higuera et al. [8] extended this approach by adding a smoothing mechanism to create near-continuous tags.

We introduce an assumption (different from [7,8]) which simplifies the concept of a tag: we assume a tag is a mask guaranteeing to *include* structures of interest, but not *exactly* delineate them. Similar to Svakhine et al. [9, Sec. 3], we then apply value-gradient transfer functions to each object. As a result, the tag texture can be interpolated using a simple nearest-neighbour scheme and the complexity of pixel-resolution boundary filtering is avoided altogether.

2.2   Line Filtering for Vessel Segmentation

We use the line enhancement filter proposed by Sato et al. [10] to segment the coronary arteries. The response of a 1-D line filter is as follows:

$$R(\boldsymbol{x}, \sigma_f) = \left[-\frac{d^2}{dx^2} G(\boldsymbol{x}, \sigma_f) \right] * I(\boldsymbol{x}) \tag{1}$$

where $*$ denotes convolution, $I(\boldsymbol{x})$ is the input image, and $G(\boldsymbol{x}, \sigma_f)$ is the Gaussian function. This filter can be extended to multiple dimensions using the Hessian matrix $\nabla^2 I(\boldsymbol{x})$. The eigenvalues $\lambda_1(\boldsymbol{x})$, $\lambda_2(\boldsymbol{x})$, $\lambda_3(\boldsymbol{x})$ (where $\lambda_1(\boldsymbol{x}) > \lambda_2(\boldsymbol{x}) > \lambda_3(\boldsymbol{x})$) of the Hessian can then be used to derive a line similarity measure (see [10, Sec. 2.1] for full details).

For our application we tune the filter to capture a single line width ($\sigma_f = 0.7$). After applying the filter, the final binary tag is extracted using region growing: for each dataset two seeds were placed in the left and right coronary arteries, and an intensity threshold specified to exclude non-vascular structures.

2.3   Fast Marching for Heart Segmentation

The Fast Marching method is a computationally efficient scheme for computing the propagation of an active contour [11]. In this scheme the front always expands outwards; that is, the speed function $F > 0$. The position of the front can be characterized by the arrival function $T(\boldsymbol{x})$ and reduces to the non-linear partial differential equation:

$$\|\nabla T\| F = 1 \tag{2}$$

Fast Marching methods are advantageous because they implicitly handle topological changes and are computationally efficient (when implemented using heap sort algorithms). The method returns the arrival function $T(\boldsymbol{x})$, from which the contour at any desired arrival time can be obtained.

We propose the use of Fast Marching for segmenting the structures comprising the heart. The pericardial cavity — in which the heart resides — acts as a boundary restricting the propagation of the contour. In this sense the Fast Marching method can be seen as a form of boundary-regulated region growing.

Speed Function
As Sethian [11, pp. 4] states, the challenge is to specify a suitable speed function F. We propose an edge-based speed function:

$$F(I(\boldsymbol{x})) = \begin{cases} SIG(\ RGM(I(\boldsymbol{x}))\ ); & I(\boldsymbol{x}) > t_l \\ 0; & \text{otherwise} \end{cases} \tag{3}$$

$$RGM(I(\boldsymbol{x})) = \left\| \nabla (\ G(\boldsymbol{x}, \sigma_g) * I(\boldsymbol{x})\ ) \right\| \tag{4}$$

$$SIG(I(\boldsymbol{x})) = \left[1 + \exp\left(\frac{-I(\boldsymbol{x}) - \beta}{\alpha} \right) \right]^{-1} \tag{5}$$

where RGM is the gradient magnitude computed using recursive filters which approximate convolution with the first derivative of a Gaussian, and SIG is the

non-linear monotonic sigmoid mapping function. The proposed speed function F varies between $[0, 1]$ and depends on four parameters: σ_g, the aperture of the Gaussian convolution kernel; α, the width of the sigmoid function; β, the centre of the sigmoid function; and t_l, a global threshold which suppresses air in the lungs. It was empirically found these parameters could assume constant values across all the datasets: $\sigma_g = 0.5$, $\alpha = -10$, $\beta = 10$, and $t_l = -250$.

In most cases the proposed speed function is adequate, however the results can be improved by pre-processing the datasets. Firstly, we employed an edge-preserving denoising technique: the curvature flow method discussed in [11, Ch. 16] with time-step $t_{cf} = 0.1$ and iterations $n_{cf} = 10$ is adequate. Next, in order to suppress artificial boundaries caused by structures containing contrast agent, a global intensity threshold ($t_c \approx 125$) was applied. Finally, in cases with thin pericardium and pleura, the sternum must be reinserted to prevent contour leakage. This can be achieved using an intensity-based region growing operation seeded anywhere in the sternum. Fig. 1 depicts this pre-processing pipeline.

Initial Contour

The initial contour plays an important role in the final arrival function. For our application it is important to quickly and easily specify an initial contour likely to propagate evenly to the pericardium. We therefore devised a simple experiment comparing three initial contour specification methods:

Point Landmarks: The user places n_p points centrally and evenly within the pericardial cavity. This is the typical method for specifying the initial contour for Fast Marching methods [12]. It should be noted that it is difficult to ensure an even spread of these points.

Spherical Landmarks: The user places n_s spheres centrally within the pericardial cavity. Each sphere is positioned and sized such that its volume does not enter unwanted structures (except the lungs). See the first row in Fig. 2.

Extruded Polygon: The user positions n_g points ($n_g \geq 3$) describing a polygon on the sagittal plane. The polygon is extruded along the plane between $[z_{start}, z_{end}]$. As with the spheres, the specified volume is not allowed to enter unwanted structures (except the lungs). See the second row in Fig. 2.

To exclude portions of the initial contours in the lungs, the initial images were masked using the same threshold as the speed function (ie. $t_l = -250$). An ROC-like analysis comparing these methods is presented in the next section.

Obtaining the Final Contour

Because the tag will be refined using a gradient-based transfer function, it is important that material boundaries are included in the final contour. To achieve this we eroded the arrival function with a ball structuring element (radius $= 2$), denoted $T_e(\boldsymbol{x})$. The optimal arrival time (t_a) can be found using a number of mechanisms: (a) interactive thresholding of the arrival function, (b) the arrival time at a user specified point, or (c) the maximum or average arrival time of a group of points. Once the desired arrival time is determined, the final binary volume can be extracted by applying a binary threshold ($0 < T_e(\boldsymbol{x}) < t_a$).
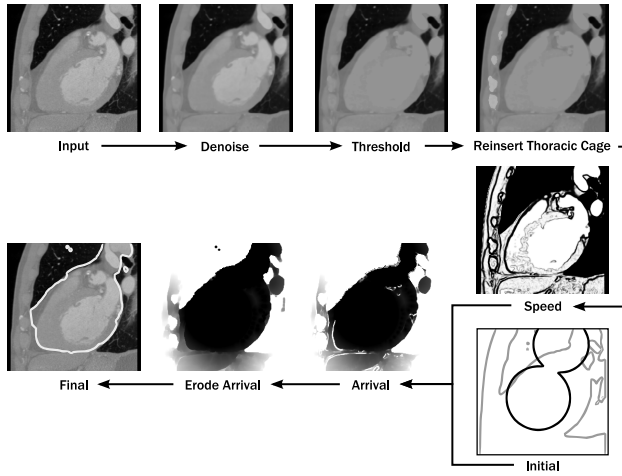
Fig. 1. This flow diagram depicts the data preparation and use of the Fast Marching method for segmenting structures within the pericardial cavity

3   Results

3.1   Data

For this study five datasets were chosen to represent anatomical variability. The first two datasets (A & B) exhibited severe calcific atherosclerosis and aortic calcification, typifying patients with advanced ischemic heart disease. The other datasets (C, D & E) displayed early indicators of heart disease: mild atherosclerosis in the coronary arteries. Dataset A is from an open-data initiative supported by the National Library of Medicine [13]. Datasets B-E were acquired in a hospital setting using contrast-enhanced, ECG-gated, 40-slice spiral computed tomography angiography (Brilliance 40, Philips Medical Systems, MA, USA)[1].

3.2   Initial Contour

An ROC-like analysis was performed to compare the three contour initialisation methods. As previously discussed, the binary tags have an inherent flexibility: they are not required to exactly delineate objects. To reflect this in our analysis — rather than comparing with manually segmented volumes — the results were compared with a number of point landmarks. For each dataset 100 points were placed in desired structures (eg. aorta, left and right coronary arteries, ventricles, and atria) and 100 points in undesired structures (eg. spine, sternum, ribs, diaphragm, and liver). A true positive occurred when a desired point fell inside the resultant segmentation; and similarly, a true negative occurred when an undesired point fell outside the resultant segmentation. This scheme allowed for

---

First row: Initial, $t_a = 0$, $t_a = 10$, $t_a = 20$, $t_a = 30$, $t_a = 40$

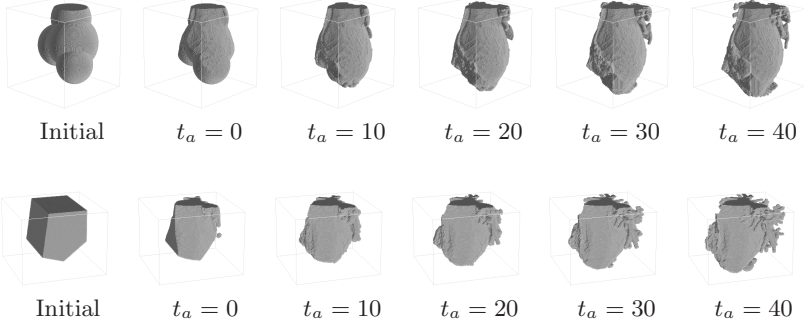Second row: Initial, $t_a = 0$, $t_a = 10$, $t_a = 20$, $t_a = 30$, $t_a = 40$

Fig. 2. Two of the contour initialisation methods at various arrival times (t_a). First row: Dataset C, four spherical landmarks. Second row: Dataset A, five-sided extruded polygon.
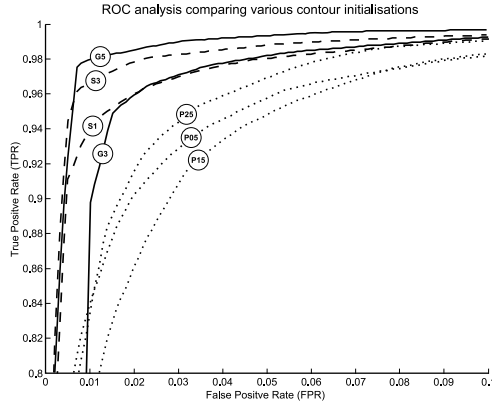

Fig. 3. ROC analysis comparing various contour initialisations. Spheres (S) and extruded polygons (G) performed slightly better than point landmarks (P).

'fuzzy' structures — such as the pericardium — which may either be included or excluded in the resultant segmentation without affecting the measure.

Different settings were used to generate a range of initial contours: for point landmarks $n_p = \{5, 10, 15, 20, 25\}$, for spherical landmarks $n_s = \{1, 2, 3, 4, 5\}$, and for extruded polygons $n_g = \{3, 4, 5, 6, 7\}$. The classifier discrimination threshold was the arrival threshold (t_a). The results from each dataset were linearly interpolated to create a common abscissa and then averaged.

The ROC curves depicted in Fig. 3 indicate that all of the contour initialisation methods produce good results (all curves tend towards the $[0, 1]$ point). The geometric shapes (spheres and extruded polygons) performed slightly better than point landmarks, most likely because they provided a better initial estimate of the desired contour. As such, the tags for the following renderings were generated using $n_g = 7$, with t_a computed using an interactive binary threshold.

Dataset A: Normal Dataset A: Tagged

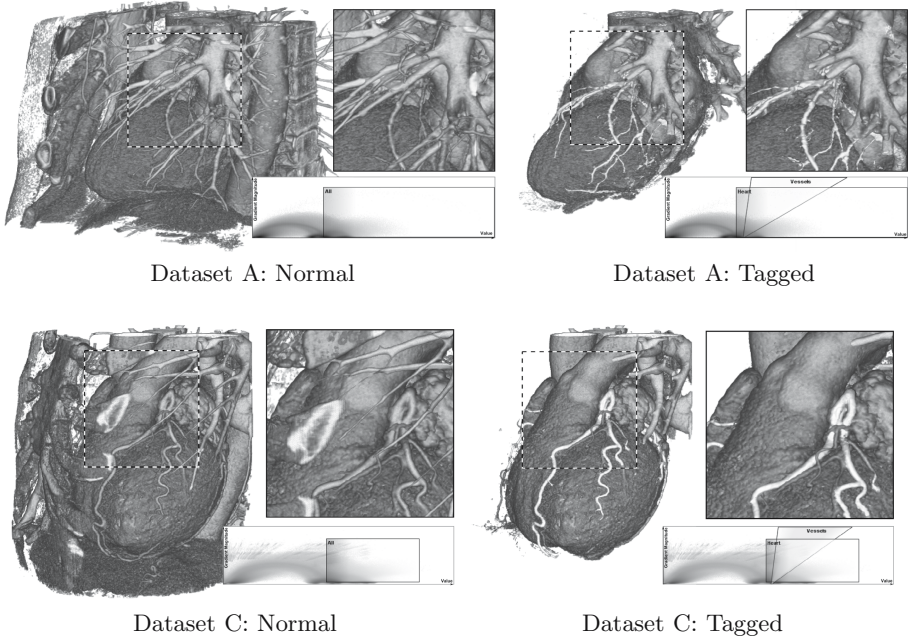Dataset C: Normal Dataset C: Tagged

Fig. 4. Resultant images using the normal and tagged volume rendering approaches. In the tagged method the heart and coronary arteries are assigned different transfer functions (inset, lower right). For a colour version of this figure see: http://eprints.qut.edu.au/archive/00008156/

3.3 Final Renderings

The offline segmentation was implemented using the Insight Toolkit (ITK) [12] and the volume rendering using OpenGL and OpenGL Shading Language (GLSL) on a desktop computer (Intel Pentium D, 2×3.0 GHz processors, 2 GB RAM, NVIDIA GeForce 8800 GTX, 768 MB VRAM). The value image was represented using 16-bits, while the gradient magnitude and tag images were represented using 8-bits. The datasets were down-sampled (256^3) using linear interpolation for the tag segmentation and ROC analysis, and then the tag returned to full size ($\approx 512^3$) using nearest-neighbour interpolation for rendering.

Each tag image was processed in roughly 7 minutes: 3 min for vessel segmentation, 2 mins for pre-processing, 1 min for computing the speed function, and 1 min for Fast Marching. Each scene was rendered with a 512^2 viewport at interactive framerates: ≈ 15 fps with an interactive sampling rate ($r = 0.5$) and ≈ 5 fps with a high quality sampling rate ($r = 2.0$). Fig. 4 depicts normal and tagged results from two of the datasets (A & C). The advantage of tagged rendering over the normal approach is self evident: the coronary arteries are more easily distinguished on account of the removed clutter and improved contrast.

4   Conclusion

We presented a novel system for tagged volume rendering of the heart. The system is comprised of two stages: offline segmentation of the coronary arteries and pericardial cavity; followed by the online application of gradient-value transfer functions to refine the tags. This two-stage approach provides for interactivity while promoting relaxed segmentation criteria. An ROC-like analysis of the active contour method (used for segmenting the pericardial cavity) showed that simple geometric shapes (such as spheres or extruded polygons) offer good trade-offs between simplicity, sensitivity, and specificity. The proposed system is useful for diagnosis and treatment planning of ischemic heart disease.

References

1. Udupa, J.: Three-dimensional rendering in medicine: some common misconceptions. In: Medical Imaging: Visualization, Display, and Image-Guided Procedures, San Deigo, USA, SPIE, vol. 4319, pp. 660–670 (2001)
2. Lorensen, B., Cline, H.: Marching cubes: A high resolution 3D surface construction algorithm. In: Computer Graphics and Interactive Techniques, pp. 163–169. ACM Press, New York (1987)
3. Pfister, H., Lorensen, B., Bajaj, C., Kindlmann, G., Schroeder, W., Avila, L., Raghu, K., Machiraju, R., Lee, J.: The transfer function bake-off. IEEE Computer Graphics and Applications 21(3), 16–22 (2001)
4. Kniss, J., Kindlmann, G., Hansen, C.: Multidimensional transfer functions for interactive volume rendering. IEEE Transactions on Visualization and Computer Graphics 8(3), 270–285 (2002)
5. Tzeng, F., Lum, E., Ma, K.: An intelligent system approach to higher-dimensional classification of volume data. IEEE Transactions on Visualization and Computer Graphics 11(3), 273–284 (2005)
6. Weber, G., Dillard, S., Carr, H.: Topology-controlled volume rendering. IEEE Transactions on Visualization and Computer Graphics 13(2), 330–341 (2007)
7. Hadwiger, M., Berger, C., Hauser, H.: High-quality two-level volume rendering of segmented data sets on consumer graphics hardware. In: Visualization, pp. 301–308. IEEE Computer Society Press, Los Alamitos (2003)
8. Higuera, F.V., Hastreiter, P., Naraghi, R., Fahlbusch, R., Greiner, G.: Smooth volume rendering of labelled medical data on consumer graphics hardware. In: Medical Imaging: Visualization, Image-Guided Procedures, and Display. SPIE, vol. 5744, pp. 13–21 (2005)
9. Svakhine, N., Ebert, D.: Illustration motifs for effective medical volume illustration. IEEE Computer Graphics and Applications 25(3), 31–39 (2005)
10. Sato, Y., Nakajima, S., Shiraga, N., Atsumi, H., Yoshida, S., Koller, T., Gerig, G., Kikinis, R.: Three-dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images. Medical Image Analysis 2(2), 143–168 (1998)
11. Sethian, J.: Level Set and Fast Marching Methods. Cambridge Press, Cambridge (1999)
12. Ibanez, L., Schroeder, L., Ng, W., Cates, L.: J.: The ITK Software Guide: The Insight Segmentation and Registration Toolkit. Kitware, Inc. (2007)
13. Holmes, D., Workman, E., Robb, R.: The NLM-Mayo Image Collection. The Insight Journal, special issue: MICCAI Open Science Workshop pp. 1–8 (August 2005)