AUTOMATIC SEGMENTATION OF THE LEFT-VENTRICULAR CAVITY AND ATRIUM IN 3D ULTRASOUND USING GRAPH CUTS AND THE RADIAL SYMMETRY TRANSFORM

Radford Juang¹  Elliot R. McVeigh²  Beatrice Hoffmann³  David Yuh⁴  Philippe Burlina¹,⁵

Johns Hopkins University
¹Applied Physics Lab., ²Dept. of Biomedical Eng., ³Dept. of Emergency Medicine, ⁴Div. of Cardiothoracic Surgery, ⁵Dept. of Comp. Sci.

ABSTRACT
In this paper, we propose a graph-based method for fully-automatic segmentation of the left ventricle and atrium in 3D ultrasound (3DUS) volumes. Our method requires no user input and can segment volumes with open and closed mitral valves. We utilize the radial symmetry transform to determine a central axis along which the 3D volume is warped into a cylindrical coordinate space. A graph is constructed for the volume in this space and a min-cut algorithm is applied to segment the left ventricle and atrium from the background. Since segmentation in the cylindrical coordinate space is defined as finding a boundary between the left (interior) and right (exterior) sides, we obviate the need for user specified seeds. The segmented results are transformed back to the Cartesian coordinate space. Experiments using intraoperative 3D ultrasound data show promising results.

Index Terms— Graph cuts, Segmentation, 3D Ultrasound, Radial Symmetry Transform

1. INTRODUCTION
Ultrasound (US) is an important imaging modality when it comes to analyzing heart motion. Unlike other imaging modalities (e.g., CT, MRI), US offers fast data acquisition rates. This comes at the expense of poorer imaging resolution. Noise from attenuation, speckle, shadows, and signal dropout makes the task of image segmentation in US imagery challenging. In this paper, we focus on the segmentation of the left ventricle and atrium in cardiac 3D US imagery.

There have been many approaches to segmenting the left ventricle ranging from level set segmentation [1, 2], to active contours or region growing techniques [3, 4], to mathematical morphology and watershed algorithms [5]. Previous research has also applied graph-based techniques to segment 2D/3D cardiac echocardiography [6]. These techniques require user initialized seeds to specify region priors. Only recently, work has been done to fully automate segmentation so that no user input is required. Active shape models [7, 8] do not require user input; however, they require a model of the left ventricle to be created as a prior. Methods to localize and select seeds as part of the segmentation process have also been devised [9, 10]. Our method is another fully automatic method that requires no user input nor any initial modeling. We present a novel approach of using the radial symmetry of the left ventricle and atrium to automate the segmentation process.

Our method is inspired by the work in [11]. In [11], a user selects a “fixation point” that acts as a point of interest for the target object. The 2D image is converted to the polar domain with the center located at this fixation point. In the polar domain, points closest to the fixation point are on the left side, while points farthest away from the fixation point are on the right side. Thus, assuming the fixation point is centered on the target object, the leftmost edge of the polar image represents points inside the target and the right most edge of the image represents points outside of the target. Graph cut segmentation in the polar domain becomes a problem of separating the left (interior) and right (exterior) sides. We extend this idea to 3D volumes, and furthermore, we automate the selection of the fixation point with the radial symmetry transform.

In this paper, we explain how the radial symmetry transform is applied to automatically select a central axis in our 3D volume. We describe how to convert this 3D volume to a cylindrical coordinate system. Subsequently, we detail how to setup a graph cut segmentation in this transformed space. We conclude with results and a discussion at the end.

2. METHOD
Our method takes advantage of the radial symmetry of the left ventricle and atrium structure. We note that each anteroposterior plane has a remarkable amount of radial symmetry (see Fig. 1a). By detecting the points containing high radial symmetry, we can determine the center of the left ventricle and atrium structure. We utilize the Fast Radial Symmetry transform developed in [12]. Using this approach, we can locate the center of the left ventricle and atrium at each anteroposterior plane and form an axis of symmetry from these points.
Polar Axis of Symmetry

We can quantify this symmetry contribution for radius \( n \) as follows:

\[
S_n = F_n * A_n
\]

\( A_n \) is a 2D Gaussian kernel to smooth out noise. \( \alpha \) is a parameter (which, for our purposes, is set to 2). \( O_n(x) \) and \( M_n(x) \) are normalized versions of \( O_n(x) \) and \( M_n(x) \) (i.e., divided by their maximums over all \( x \)). Once we have computed \( S_n \) for varying radius values of \( n \), these results are averaged to yield a map expressing the radial symmetry at each position.

2.2. Cylindrical Coordinate System

We perform the Fast Radial Symmetry Transform described in the last section for each anteroposterior slice (i.e., z-plane) of the volume. The result of the transform applied to the slice in Fig. 1a is shown in Fig. 1c. Once the transform is computed, we locate its local maxima. If the value of the maxima is sufficiently large (meaning there is a high measure of radial symmetry surrounding the point), we include this point’s position in our list of maxima positions. Since it is possible for each slice to contain multiple maxima positions, we have to fit a line running through all slices to the set of maxima positions that we have identified. We remove any outliers (e.g., using RANSAC) and apply linear regression. This fitted line represents the axis of symmetry for the volume (see Fig. 1d).

Fig. 1. An anteroposterior slice of the left ventricle.

We perform the Fast Radial Symmetry Transform described in the last section for each anteroposterior slice (i.e., z-plane) of the volume. The result of the transform applied to the slice in Fig. 1a is shown in Fig. 1c. Once the transform is computed, we locate its local maxima. If the value of the maxima is sufficiently large (meaning there is a high measure of radial symmetry surrounding the point), we include this point’s position in our list of maxima positions. Since it is possible for each slice to contain multiple maxima positions, we have to fit a line running through all slices to the set of maxima positions that we have identified. We remove any outliers (e.g., using RANSAC) and apply linear regression. This fitted line represents the axis of symmetry for the volume (see Fig. 1d).

The axis of symmetry line intersects each z-plane at exactly one point. This intersection can be used as a central point to convert the plane from rectangular coordinates \((x, y)\) to polar coordinates \((r, \theta)\). The conversion is as follows:

\[
r = \sqrt{x^2 + y^2}
\]

\[
\theta = \tan^{-1}\left(\frac{y}{x}\right)
\]

Each z-plane is converted to polar coordinates. The end result is a volume in the cylindrical coordinate system \((r, \theta, z)\). Fig. 1b is the polar coordinate version of Fig. 1a.

2.3. Graph Cuts

The next step in the process is to perform the actual segmentation. We utilize graph cut segmentation as detailed in [13]. Consider the set of voxels in the volume as a network...
of nodes. Each voxel is a node \((V_i)\) that is connected to its neighbor \((V_i \in N(V_j))\) by an edge \((E_{ij})\). Each edge \(E_{ij}\) is associated with a weight \(w_{ij}\). In addition, every voxel is connected to both a Source node \((S)\) and a Terminal node \((T)\). The edges from voxel \(V_i\) to these two nodes are \(E_{S,i}\) and \(E_{T,i}\), and they have weights \(s_i\) and \(t_i\) respectively.

The goal of graph cuts is to partition the network of nodes into two partitions by "cutting" (removing) edges such that there exists no edges linking nodes from one partition to the other. The edges are cut such that the cost of the cut is minimal. The cost of the cut is the sum of the weights \((w_{ij})\) of the edges that were removed. In the end, all nodes in the partition will be connected to either \(S\) or \(T\) (but not both). Nodes that are connected to \(S\) are classified as object nodes. Nodes that are connected to \(T\) are classified as background nodes. Fig. 2 is an illustration of how neighboring voxels in one slice are connected. More details on performing optimal graph cuts can be found in [13, 14]. In this section, we will focus on the assignment of the edge weights of the graph. Once we perform a minimum cut on the graph, we have automatically assigned each voxel to either the background or object class.

In the cylindrical coordinate system, the \(r = 0\) plane lies within the left ventricle and atrium. The only exceptions to this is the endocardium wall at the ends of the axis and also locations where the mitral valve leaflets are closed. These are easy to identify. We cluster the intensities in the \(r = 0\) plane into two clusters using k-means. The cluster with the higher intensity values are intensity values denoting tissue locations. All voxels in this set are labeled as set \(\text{FG}\). The other cluster represents values denoting the cavity. We identify the voxels belonging to this latter cluster as \(\text{BG}\).

We assign weights connecting \(S\) and \(T\) as follows:

\[
s_i = \begin{cases} 
\infty & \text{if } r = 0 \text{ and } V_i \in \text{FG} \\
0 & \text{otherwise}
\end{cases}
\]

\[
t_i = \begin{cases} 
\infty & \text{if } r = r_{\text{max}} \text{ or } (r = 0 \text{ and } V_i \in \text{BG}) \\
0 & \text{otherwise}
\end{cases}
\]

For \(E_{ij}\) connecting voxels \(V_i\) to \(V_j\) (where \(i \neq j\)), we want \(E_{ij}\) to be cut if it straddles a detected intensity edge (i.e., a steep intensity gradient). These are voxels that straddle the foreground and background regions. These cuts are preferred over any other cuts. The next priority given to cuts are if the voxels lie within the same region and are close to the detected intensity edge. The closer the voxels are to the intensity edge, the more likely the edge linking them will be cut. We can express this as follows.

Let \(G_{\text{thresh}}(x)\) represent the detected intensity edge. This can be as simple as the thresholded gradient (where the volume is smoothed prior to computing the gradient):

\[
G_{\text{thresh}}(x) = \begin{cases} 
1 & \text{if } (\|\nabla I(x)\| \geq T) \\
0 & \text{otherwise}
\end{cases}
\]

We use a more robust method to detect intensity edges by using the k-means to cluster the volume intensities into two groups. We call the group with the lower intensity values \(K_1\) and the group with the higher intensity values \(K_0\). The voxels situated on the boundary between these two group intensities are the thresholded edges we desire. The task of finding the boundary voxels can be accomplished by constructing a binary map consisting of 0’s for voxels in \(K_0\) and 1’s for voxels in \(K_1\). The perimeter is just this binary map multiplied by the morphological erosion of the binary map.

Next, we construct a distance map, \(D_{\text{edge}}(x)\). This is the distance transform of \(G_{\text{thresh}}(x)\) where the distance to the intensity edges are computed. \(D_{\text{edge}}(x)\) is the distance of voxel \(x\) to the closest intensity edge. With this constructed, \(w_{ij}\) can be expressed as follows:

\[
w_{ij} = \begin{cases} 
\lambda(e^{-(I_i - I_j)^2/(2\sigma^2)}) & \text{if } (V_i \in K_0 \text{ and } V_j \in K_1) \text{ or } (V_i \in K_1 \text{ and } V_j \in K_0) \\
C + |D_{\text{edge}}(V_i)| + |D_{\text{edge}}(V_j)|/2 & \text{otherwise}
\end{cases}
\]

\(C\) is the maximum of \(\lambda(e^{-(I_i - I_j)^2/(2\sigma^2)})\) for voxels where \((V_i \in K_0 \text{ and } V_j \in K_1)\) or \((V_i \in K_1 \text{ and } V_j \in K_0)\) is true. We set \(\lambda\) to twice the range of the intensity values in the volume data (in our case, this is 500). After graph cuts is performed, the result is transformed to the Euclidean coordinate system to yield our final results.

### Table 1. Individual Results (in pixels)

<table>
<thead>
<tr>
<th>Data</th>
<th>Mean Error ± StdDev</th>
<th>RMS Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>3.3892 ± 5.1392</td>
<td>6.1559</td>
</tr>
<tr>
<td>A2</td>
<td>3.9955 ± 4.8047</td>
<td>6.2487</td>
</tr>
<tr>
<td>B1</td>
<td>3.2623 ± 4.9935</td>
<td>5.9646</td>
</tr>
<tr>
<td>B2</td>
<td>3.0075 ± 3.9015</td>
<td>4.5321</td>
</tr>
<tr>
<td>Average</td>
<td>3.4136 ± 4.5819</td>
<td>5.7253</td>
</tr>
</tbody>
</table>

### Table 2. Individual Results (in mm)

<table>
<thead>
<tr>
<th>Data</th>
<th>Mean Error ± StdDev</th>
<th>RMS Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>2.3724 ± 3.5974</td>
<td>4.3091</td>
</tr>
<tr>
<td>A2</td>
<td>2.7968 ± 3.6363</td>
<td>4.3741</td>
</tr>
<tr>
<td>B1</td>
<td>2.2836 ± 3.4954</td>
<td>4.1752</td>
</tr>
<tr>
<td>B2</td>
<td>2.1052 ± 2.3731</td>
<td>3.1725</td>
</tr>
<tr>
<td>Average</td>
<td>2.3895 ± 3.2073</td>
<td>4.0077</td>
</tr>
</tbody>
</table>

### 3. RESULTS

We applied our method to intraoperative patient data. The volumes are of size 224 x 208 x 208 with a resolution of 0.7 x 0.7 x 0.6 mm per voxel. They were acquired with an iE33 Philips console fitted with a Philips X2-T Live 4D TEE probe. The data was collected intraoperatively using a 7 breath-hold cycle protocol. The results of the segmentation are shown in Figure 3. The four volumes examined came from two different patients. Volumes containing both an open and closed mitral valve were acquired from each patient.

Comparison to ground truth data yields a mean error of 3.41±4.58 pixels (2.39±3.21 mm). These results are detailed...
in Tables 1 and 2. The segmentation results are more conservative in comparison to the ground truth with errors corresponding to regions just outside of the segmentation. These regions are marked in the ground truth and are areas where uncertainty exists in delineating a boundary between the ventricle and myocardium wall.

4. CONCLUSION

We have described a method to automatically segment the left ventricle and atrium without any user input. We propose a novel 3D US segmentation approach using the radial symmetry transform and the application of segmentation techniques in the cylindrical coordinate system. This method can be extended to other types of segmentation techniques, such as level sets and watershed. We tested our method on intraoperative 3D ultrasound data and found promising preliminary results. The method is generic enough to be applied to other imaging modalities such as MRI or CT.

5. REFERENCES


