1989-09-01

A Relaxation Algorithm for Segmentation of the Endocardial Surface from Cine CT

William A. Barrett
william_barrett@byu.edu

Bryan S. Morse
morse@byu.edu

Original Publication Citation
W. A. Barrett and B. S. Morse, "A relaxation algorithm for segmentation of the endocardial surface from cine CT," in IEEE Proceedings of Computers in Cardiology, pp. 95-98, September 1989.

A RELAXATION ALGORITHM FOR SEGMENTATION OF THE ENDOCARDIAL SURFACE FROM CINE CT

William A. Barrett and Bryan S. Morse

Department of Computer Science, Brigham Young University,
Provo, Utah 84602

Summary

A relaxation algorithm has been developed for automated segmentation of the endocardial surface from contrast Cine CT images. The image is contoured at an initial density threshold and a one-dimensional edge operator is applied orthogonally to each point of the contour. Output from the operator is used to generate a histogram, the mode of which identifies a new threshold. The image is contoured again at the new threshold and the process is repeated. Iteration continues with successive threshold estimates converging to a stable value in the region of the endocardial surface. Computer-determined thresholds compare favorably with manual segmentation while reducing processing time and increasing reproducibility as well.

Introduction

Techniques have been previously developed for automated tracking and dynamic interactive display of three- and four-dimensional (3D and 4D) endocardial surface anatomy from Cine CT images.[1] However, identification of the surface to be tracked still requires manual interaction for

1. selecting a segmentation threshold
2. defining the volume surrounding the surface
3. specifying a start point for surface tracking

As a result, extensive user interaction is required to extract multiple (4D) surfaces representing discrete time instances over the cardiac cycle, thereby limiting patient throughput and minimizing clinical utility. In addition, intra/interobserver variability in the specification of these three parameters frequently requires the process to be repeated. However, these parameters can be determined automatically using a new improved[2] relaxation algorithm for segmentation of the blood-contrast mixture which defines the endocardial surface.

The Relaxation Algorithm

The relaxation algorithm consists of four modules shown in the control diagram in Figure 1.

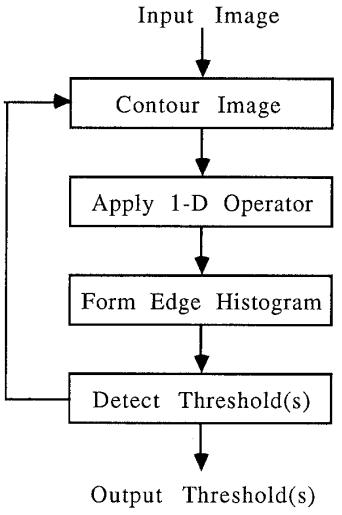

Figure 1. Control structure for algorithm modules

The relaxation algorithm begins by applying a contour-following algorithm to the image at an initial (arbitrary) density threshold, t_0. The contour information is then passed to the next module which samples one-dimensional density profiles orthogonal to the contour in order to produce a new threshold estimate. The new threshold estimate, t_1, is identified in the third module as the mode of a histogram formed from mid-range densities extracted from each of the profiles. The image is contoured again at t_1 and the process is repeated until a stable threshold is obtained, where "stability" is determined by one or more convergence criteria. Stable threshold(s) are then ranked according to their prominence (determined from the edge histogram) and output. Each of the four modules are described below.

Contour-Following Algorithm

Input: Threshold estimate, t; contour value, m
Output: 4-connected contours surrounding connected pixel regions with value $\geq$ t

```
begin
   Scan pixels, p, in image in row major order
   if (p ≥ t) and (p_1 < t ) then if (p_1 ≠ m) then
   a. back up one pixel;
   b. start_pos <- p_pos;
   c. done <- false;

   d. while (not done)do
      1. while pixel in front ≥ t do
            turn left;
         end_while;

      2. p <- m;
      3. step;
      4. done <- (start_pos = p_pos);
      5. if (not done) then
            turn right;
         end_if;
      end_while;
   end_if;
end
```

The contour-following algorithm scans the image in row major order until a region/pixel with value $\geq$ the current threshold estimate is encountered. The pixel just outside of the region is marked as the start position (b). Then, as long as the pixel in front is greater than or equal to t, the stepping direction is updated by turning left 90^0 (d.1). Contour pixels are marked, while stepping and turning right 90^0 (d.2-5), until the start position is again encountered.

The One-dimensional Edge Operator

After the image is contoured, a one-dimensional (1D) edge operator is applied in an orthogonal direction to each point of the contour. The 1D operator is faster than conventional 2D edge kernels and can be increased in size with no additional increase in computation. Application of the operator in a direction orthogonal to the contour insures that the gradient is maximized. The end points, (x_1, y_1) and (x_2,y_2), of a line orthogonal to a given contour coordinate, (x,y), are determined by calculating the x and y displacements, dx and dy, between contour coordinates sampled $\pm$ m from (x,y), as diagrammed in Figure 2 below.

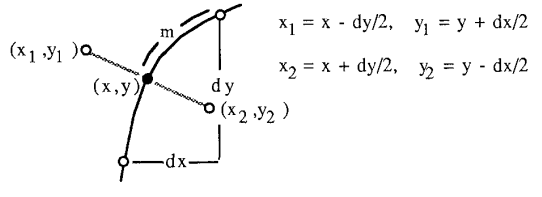

Figure 2. Calculation of end points for line orthogonal to contour.

Using this method, the distance from a contour point (x,y) to one of the end points ((x_1,y_1) or (x_2,y_2) will be on the order of 1 to 1.414m. This allows the length of the orthogonal line to decrease for areas of the contour with high curvature. The 1D edge operator itself makes use of orthogonal line end points to obtain successive threshold approximations as described below.

Consider the characteristic sigmoidal density profile (1D edge) diagrammed in Figure 3 below, with the optimal edge threshold, T, occurring at the mid-range density (ie. maximum gradient). Assume the image has been contoured at an initial threshold approximation, t_0. The function of the operator is to bring the initial approximation closer to the ideal value, T. This is accomplished by sampling k densities D_j at opposite ends of the orthogonal line. In Figure 3, k=3, with the middle sampled value corresponding exactly to one of the two end points (x_1,y_1) or (x_2,y_2). The new threshold approximation, t_i, is simply the average of the 2k (n) sampled values. The number of iterations required to converge on T is a function of the sampling interval and the distance between T and the initial approximation t_0.

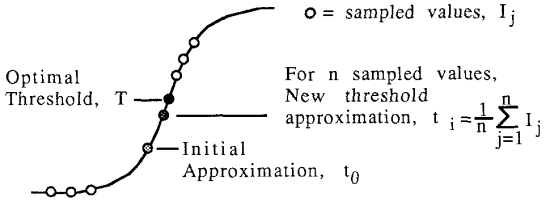

Figure 3. The one-dimensional edge operator.

The purpose of the 1D operator is to determine the most frequent (local) density at which the steepest gradient occurs in order to obtain a global threshold. Hence, points on the orthogonal line are only sampled if the density at (x_1,y_1) is less than the current approximation t_i and the density at (x_2,y_2) is greater than t_i (or vice versa). This is efficiently determined by (frame-time) thresholding of the image at level t_i and then simply checking for a difference in the sampled end points in the thresholded image.

Formation of the Edge Histogram

The edge histogram is simply a histogram of the new approximation, t_i, derived from the orthogonal sampling for each contour point in the image. The characteristics of the (3-point smoothed) edge histogram are used to determine whether or not a stable threshold has been reached.

Threshold Detection

Two convergence criteria are used to detect stable threshold(s). These include:

1. An edge histogram with a single significant peak (unimodal) is detected. The edge histogram is classified as unimodal if the area under any secondary peaks is less than 2% of the area under the peak associated with the mode.
2. The mode of the edge histogram has not changed from iteration t_i to t_{i+1}. This occurs near the middle of the edge profile because the edge operator is "pulling" equally from both directions.

If more than one stable threshold is detected, the thresholds are ranked according to their prominence in the image. Ranking criteria are highest mode frequency and total histogram area to secondary peak ratio.

Secondary thresholds are also considered if:

1. $\Sigma\ |t_i - t_{i+1}| < \delta$. This means that there was a small cumulative difference between successive threshold approximations over some interval.
2. $t_i = t_{i+k}$, $k > 1$. This indicates a repetition or oscillation over the same sequence. In both of these cases the value of the secondary threshold is the average over the sequence.

After convergence to a stable threshold is achieved, the edge histogram is smoothed and searched for secondary peaks (local maxima which represent > 2% of the total area under the histogram). Secondary peaks arise due to the spatial connectivity of regions identified by the 1D edge operator and neighboring regions which fall within a different density range. Hence, secondary peaks are strong indicators of potentially stable thresholds which define these neighboring regions. To make use of this information, density values associated with secondary peaks are placed on a stack and investigated for convergence to a separate stable threshold.

The use of secondary peak densities allows the spectrum to be searched in an intelligent manner and greatly reduces the computation which would otherwise result from a methodical investigation of each density in the image. Computation is further reduced by checking successive threshold estimates stemming from secondary peak densities against previously detected stable thresholds or tested values. If the check reveals a difference within $\pm\delta$ ($\delta = 5$), it is assumed that the sequence will lead to no new stable thresholds, and the process is terminated. The same rationale applies if a value in a sequence is repeated, however, in this case the "cycle" is used to output a secondary threshold, as described above.

When the stack is empty a list of stable threshold(s) is ranked and output. The algorithm is given below.

The Threshold Detection Algorithm:

Input: Initial non-zero threshold approximation, t_0.
Output: List of stable threshold(s), T

Data Structures:
a stack S, of potential thresholds
a list L, of previously tested thresholds
a histogram array H.

```
begin
  1. place t0 on S;
  2. t_last <- 0;
     unimode , repeat <- false;
  3. while S is not empty do
     a. remove an entry t from S;
     b. for all Ti and Lj
            if (|t - Ti| < δ)or (|t - Lj| < δ)then
               repeat <- true;
            end_if;
        end_for;
     c. if not repeat and ((t = t_last) or unimode) then
            output t to T;
            output any secondary peak densities to S;
            output all ti from this sequence to L;
            output any secondary thresholds to T;
            unimode , repeat <- false;
     d. else
            contour image at t;
            form edge histogram H;
            t_last <- t;
            t <- mode (H);
            output t to S;
            if H is unimodal then
                 unimode <- true;
            else unimode <- false;
     end_while;
end
```

The objective of the algorithm is to output a list, T, of stable threshold(s) starting from an initial threshold estimate, t_0. The algorithm works off of a stack S of potential thresholds which is fed by the initial estimate, t_0, and secondary peaks in the edge histogram, H. A list, L, of all previously tested thresholds is maintained to expedite processing.

After initialization (1 and 2) a value, t, is removed from S and investigated for convergence to a stable threshold (3a). If any t_i fall within range of a previously tested or stable value, the sequence is abandoned (3b). if this does not happen, subsequent threshold estimates are examined recursively until a stable threshold value is reached which is determined when t = t_last or H is unimodal (3c). At this point t is output to T and threshold candidates stemming from secondary peaks are output to S. Any secondary thresholds as defined by the criteria above are also output to T.

The heart of the algorithm is given in 3d where initially the image is contoured at t and the edge histogram H is formed. The mode of H is then determined and output to S. If H is unimodal or t=t_last, t will be output to T the next time through the loop. The search continues until S is empty and the spectrum is exhausted.

The convergence of the algorithm is illustrated in Figure 4. The top image indicates the points which were sampled based on the initial threshold approximation to form the edge histogram on the right. The marked bimodality of the histogram is an indication that the threshold has not yet stabilized to the middle of the edge profile, whereas the unimodal distribution (bottom) indicates convergence to a stable value. The segmentation based on the detected stable threshold is shown in Figure 5.

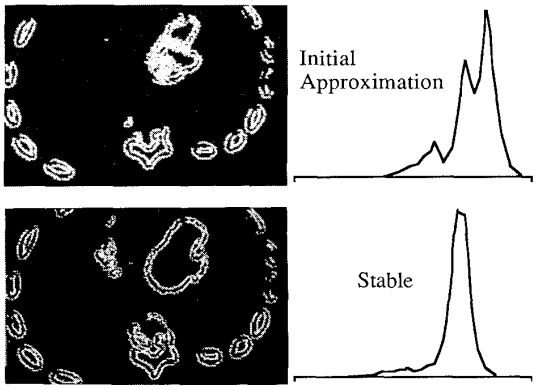

Figure 4. Sampled points with corresponding edge histograms.

Figure 5. Segmentation based on stable threshold with minimum rectangle superimposed.

Determination of Start Point and Enclosing Volume

After a segmentation threshold has been computed, the volume enclosing the anatomy of interest and a start point for surface tracking must be defined for extraction of the endocardial surface. The enclosing volume can be defined from a minimum rectangle surrounding the segmented anatomy at each level (Figure 5). A "signature parsing" technique[3] was developed to compute the bounding rectangles. Maximizing over rectangles at all levels produces a maximum rectangular region of interest which can be projected through all levels to define the enclosing image volume used for surface tracking. The start point for surface tracking can be any point at which the thresholded (white) region is tangent to the bounding rectangle.

Results

The relaxation algorithm was tested on 28 images taken from 7 patients at end-diastole. (Four images from different levels were used from each patient.) The patient population included three normals, an atrial tumor, apical akinesis, an aneurysm, and IHSS.

The blood-contrast mixture was detected in all cases using the threshold detection algorithm. If more than one stable threshold was detected the maximum was always chosen for segmentation. Before applying the algorithm segmentation thresholds were determined manually for each of the 28 images. A comparison of computer and manually-determined thresholds showed good agreement in 20 of the images, while the algorithm underestimated the threshold in 7 of the 8 remaining cases. However, it should be pointed out that 3 of the lower computer-determined values were still sufficient to clearly segment the blood-contrast mixture and in 2 cases the computer-determined values were judged (retrospectively) to be preferable to the original manually-determined values.

Conclusions

An algorithm has been described for segmentation of blood-contrast regions from Cine CT images. Computer-determined thresholds compare well with manually-defined values while providing a mechanism for fully automated extraction of the endocardial surface. Key features of the algorithm include the use of local edge profiles to determine a globally optimum threshold and convergence starting from any initial input density. The automated methodology makes 3D surface detection reproducible, minimizes reprocessing, avoids manual parameter selection, and therefore greatly increases throughput and clinical utility.

Future work includes investigation of adaptive techniques for measurement and normalization of edge profiles and determination of an optimal sampling interval. The algorithm will also be enhanced to detect of subobjects within minimum bounding rectangles. Finally, thresholds from all levels will be compared for consistency.

Bibliography

1. Barrett, W.A. and Udupa, J.K.: Dynamic Display and Quantitative Analysis of Three-Dimensional Left Ventricular Pathology, Proceeding of the IEEE, Computers in Cardiology, Bethesda, MD, 1988.
2. Barrett, W.A.: An Iterative Algorithm for Multiple Threshold Detection, IEEE Proceedings of Pattern Recognition and Image Processing, 1981.
3. Heaton, T, et al.: Fast Automated Object Detection Using Signature Parsing, SPIE Proceedings on Intelligent Robots and computer Vision VIII: Algorithms and Techniques, November 5-10, 1989.