A Scale-Space Based Approach for Deformable Contour Optimization *

Yusuf Sinan Akgul and Chandra Kambhamettu

Video/Image Modeling and Synthesis (VIMS) Lab
Department of Computer and Information Sciences
University of Delaware
Newark, Delaware, 19716
USA
{akgul,chandra}@cis.udel.edu

Abstract. Multiresolution techniques are often used to shorten the execution times of dynamic programming based deformable contour optimization methods by decreasing the image resolution. However, the speedup comes at the expense of contour optimality due to the loss of details and insufficient usage of the external energy in decreased resolutions. In this paper, we present a new scale-space based technique for deformable contour optimization, which achieves faster optimization times and performs better than the current multiresolution methods. The technique employs a multiscale representation of the underlying images to analyze the behavior of the external energy of the deformable contour with respect to the change in the scale dimension. The result of this analysis, which involves information theoretic comparisons between scales, is used in segmentation of the original images. Later, an exhaustive search on these segments is carried out by dynamic programming to optimize the contour energy. A novel gradient descent algorithm is employed to find optimal internal energy for large image segments, where the external energy remains constant due to segmentation.
We present the results of our contour tracking experiments performed on medical images. We also demonstrate the efficiency and the performance of our system by quantitatively comparing the results with the multiresolution methods, which confirm the effectiveness and the accuracy of our method.

1 Introduction

A deformable contour[8] is an energy minimizing model which is popularly used for automatic extraction and tracking of image contours. One of the main reasons of the popularity of deformable contours is their ability to integrate image level bottom up information, task dependent top down knowledge information

* This work is supported by Grant No. R01 DC01758 from NIH and Grant No. IRI 961924 from NSF.

and the desirable contour properties into a single optimization process. A deformable contour model has two types of energies associated with it: an internal energy, which characterizes the desirable attributes of the contour, and an external energy, which ties the contour with the underlying image. The framework is based on minimizing the sum of these energies. Formally, a discretized version of a deformable contour is an ordered set of points $V = [v_1, v_2, ..., v_n]$. Given an image I, the energy associated with a deformable contour, V, can be generally written as

$$E_{Snake}(V) = \sum_{i=1}^{n} \alpha E_{int}(v_i) + \beta E_{ext}(v_i, I) \quad (1)$$

where E_{int} is the internal and E_{ext} is the external energy of the contour element v_i, and α and β are the weighting parameters.

Although for the majority of the applications, the main framework for the formulation stayed more or less the same, there have been numerous proposals for minimization techniques. Original proposal[8] and many others used variations of gradient descent algorithms for the minimization of Equation(1). While the internal energy definitions are suitable for optimizations based on gradient descent, external energies usually include large amounts of noise, which makes gradient descent methods sensitive to convergence to local minima instead of global minima, numerical instability and inaccuracy problems.

Application of dynamic programming (DP) to deformable contour minimization[3][5] addresses these problems. As we will explain in section 2.1, DP solves the optimality, numerical stability and incorporating hard constraints problems. However, although the time complexity is polynomial, DP suffers from long execution times. In order to shorten execution times for practical applications, researchers commonly suggested[5] using a multiresolution framework. The main idea of using multiresolution techniques for DP is to decrease the number of degrees of freedom for each contour element. Since the underlying images are smaller at the lower resolutions, there are less number of image positions that a contour element can take, resulting in faster exhaustive enumeration times. The details of current multiresolution techniques are explained in section 2.2.

There are some problems with the above multiresolution techniques. First, during the construction of lower resolution levels, these techniques utilize the external energy of the deformable contour minimally. This is a serious problem because only the external energy ties the deformable contour to the new resolution image. Another problem with the current multiresolution methods is that, while the image size is decreased, the fact that the new resolution image will be used in an exhaustive enumeration, which is a very costly process, is completely neglected. Neighboring image locations that will produce the same energies should be unified to one location. We describe the details of these problems and a few others in Section 2.2.

This paper addresses the above problems by employing a multiscale representation instead of a multiresolution representation. The method segments the underlying images by analyzing their structures with respect to the external energy in the scale-space. The segments are formed in a way that, in the final segmentation the external energy related information is kept closer to the maximum by measuring the change of this information with respect to the scale change through an information theoretic approach. A special dynamic programming technique[1] is then applied to optimize the energy of Equation(1) by using the centroids of these segments as the degrees of freedom. This paper extends our previous work[1] by employing a different segmentation technique that uses information and scale-space theory.

2 Snakes, DP, and Problems with Multiresolution Methods

In this section we define the deformable contour energies, the details of DP methods and multiresolution methods. We will also describe the problems that we address in detail.

2.1 Snakes and DP

Equation(1) describes the general form of the snake energy. The internal energy of the snake serve to impose smoothness and continuity of the contour. As mentioned earlier, the external energy, on the other hand, ties the contour to the underlying image by pushing the snake toward application dependent image features like edges. One of the biggest advantages of using snakes is that specific applications can change the internal and external energy definitions without affecting the general framework.

We define the internal energy as follows:

$$E_{int}(v_i) = \left(1 - \frac{\overrightarrow{v_{i-1}v_i} \; . \; \overrightarrow{v_i v_{i+1}}}{|\overrightarrow{v_{i-1}v_i}| \, |\overrightarrow{v_i v_{i+1}}|} \right) + \gamma ||v_i - v_{i+1}| - d| \tag{2}$$

where γ is the weighting parameter and d is the distance needed between the contour elements. The first part of this energy formulation, which is the dot product of two vectors(Figure 1), is for imposing smoothness of the contour.

Given an image I, one possible definition for the external energy is

$$E_{ext}(v_i, I) = -\, |\nabla I(v_i)| \tag{3}$$

which is the negative of the image gradient ∇I at v_i. Given the above formulations and an image I, we can extract and track object boundaries by defining a search window around each contour element and selecting the candidates from these search windows that minimizes the snake energy (Figure 1). The desired contour, $V = [v_1, v_2, ..., v_n]$, can be obtained by

$$V = \arg\min_{v} \sum_{i=1}^{n} \alpha E_{int}(v_i) + \beta E_{ext}(v_i, I) \tag{4}$$

Assuming there are m different positions that the contour element v_i can take in a search window W_i, the cost of iteratively testing each possible element configuration is $O(m^n)$, which grows exponentially. Fortunately, the optimization

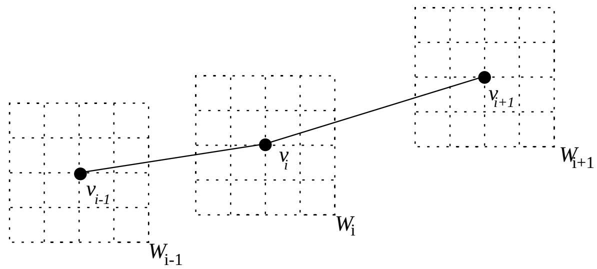

Fig. 1. Each contour element v_i has a search window W_i defined around it.

of the snake energy can be done in polynomial time using dynamic programming. Amini *et. al.*[3] and Geiger *et. al.*[5] proposed DP methods for deformable contour optimization. In this paper, we will use Amini *et. al.* formulation. Our system can easily be ported to the system of Geiger *et. al.*

The main idea under the DP method is that each contour element, v_i, can take only one possible position from the search window W_i. We also observe that the energy formula of Equation(1) can be written in terms of separate energy terms of $E_1, E_2, .., E_{n-2}$, such that each energy term E_{i-1} depends only on v_{i-1}, v_i, v_{i+1}.

$$E_{Snake}(v_1, v_2, ..., v_n) = E_1(v_1, v_2, v_3) + E_2(v_2, v_3, v_4) + ... + E_{n-2}(v_{n-2}, v_{n-1}, v_n)$$

where

$$E_{i-1}(v_{i-1}, v_i, v_{i+1}) = E_{int}(v_i) + E_{ext}(v_i). \tag{5}$$

Next we write a set of optimal value functions that hold the best energy configurations up to the current contour element.

$$\begin{aligned} s_1(v_2, v_3) &= \min_{v_1} E_1(v_1, v_2, v_3) \\ s_2(v_3, v_4) &= \min_{v_2} E_2(v_2, v_3, v_4) + s_1(v_2, v_3) \\ &\cdots \\ s_{n-2}(v_{n-1}, v_n) &= \min_{v_{n-2}} E_{n-2}(v_{n-2}, v_{n-1}, v_n) + s_{n-3}(v_{n-2}, v_{n-1}) \end{aligned}$$

Finally, we can write

$$\min E_{Snake} = \min_{v_{n-1}, v_n} s_{n-2}(v_{n-1}, v_n).$$

Since each optimal value function is calculated by iterating on three contour elements and there are $n-2$ of them, the time complexity of DP algorithm is polynomial and it is $O(nm^3)$. The resulting contour produced by the DP algorithm is optimal since it checks every possible alternative.

Although the time complexity of DP algorithm is polynomial, it is still too slow for some practical applications. Application of DP in combination with multiresolution methods addresses this problem, which is explained in the next section.

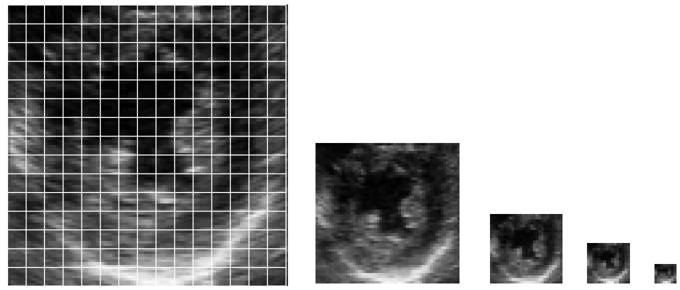

Fig. 2. A multiresolution representation of an echocardiographic image: The leftmost image is the original 240x240 image. The rightmost image is the 15x15 top level image. Each pixel in this 15x15 image represents a square shaped 16x16 segment marked on the original image.

2.2 DP with Multiresolution Methods and Problem Details

We will use the Gaussian pyramid as the basic multiresolution method[4] in explaining the general structure and describing the problems. Geiger *et. al.*[5] uses a different multiresolution scheme[6], which preserves discontinuity between the image resolutions. However, most of the problems with the existing techniques are also present in their method.

A Gaussian pyramid for an image I is a sequence of copies of I, where each successive copy has half the resolution and sample rate. The levels of a Gaussian pyramid for given image I is calculated as

$$\begin{aligned} G_I(ij0) &= I(ij) \\ G_I(ijk) &= \sum_{m,n} w(mn) G_I(2i-m, 2j-n, k-1) \end{aligned} \tag{6}$$

where k is the pyramid level. The motivation in using a multiresolution method for the snake optimization is that lower the image resolution, smaller the search windows, which means lower number of candidate positions in each search window. Decreasing the resolution in a multiresolution representation may be viewed as segmenting the original image into equal sized square segments and representing each segment with a single pixel whose gray-level value is usually given by the average of the area around the segment(Figure 2). Deformable contour optimization algorithms are applied to the highest level of the pyramid. The obtained contour is an approximation of the final contour and it is used as the initial snake position for the next lower level. Using a smaller window size, the optimization is performed at the current level, and the process continues until the contour is optimized at the lowest level, which is the original image level. As expected, a multiresolution based DP does not necessarily produce optimal contours.

We mentioned before that only the external energy ties the deformable contour to the underlying image. However, during resolution decreasing steps, Equation (6) utilizes external energy minimally, which increases the loss of external energy related information. We argue that, unlike in Figure 2, the pixels in the lowest-resolution image should represent different sized segments in the original image. This will give us a possibility of choosing a smaller segment size on areas where external energy shows greater variations, resulting in better representation of external energy and less loss of external energy related information.

Another problem with the above multiresolution method is efficiency related. We know that the purpose of using a multiresolution method is to reduce the number of candidates in a search window so that the enumeration process gets faster. Therefore, during the construction of the newer resolutions, neighboring elements of search windows that will produce about the same energy should be unified into a single element. This modification can also be done by employing different sized segments – We choose a larger segment for areas where external energy remains relatively constant on the original image. This will increase the system efficiency without decreasing the performance because at the upper levels of the pyramid, we are not looking for the final version of the contour but only an approximation.

3 A New Scale-Space Based Approach for Deformable Contour Optimization

In the previous section, discussions on the problems of DP multiresolution methods suggested that in order to utilize external energy properly, each pixel in the lowest resolution pyramid image should represent a variable sized segment in the original image. However, achieving this is very difficult with the multiresolution techniques because of their inherent nature – A pixel in a multiresolution pyramid level can only represent a fixed sized segment in the lower pyramid level. Therefore our new method does not use the multiresolution approach.

Our solution is based on scale-space techniques, which have received a considerable amount of attention in the computer vision field[10]. The main idea of producing a multiscale representation is to simplify the underlying image by removing the fine scale details while continuously increasing the scale. This kind of approach gives us the possibility of analyzing the image structure with respect to scale. In other words, we can analyze the change of the image structure while the image undergoes a simplification transformation. There are major differences between a multiresolution representation and a multiscale representation. Lindeberg has a thorough discussion about the differences in [10] and we use the terminology used by him. As its name implies, a multiresolution representation decreases the image resolution while forming the pyramid levels. On the other hand, a multiscale representation keeps the spatial sampling constant while the scale changes.

Our new method for deformable contour optimization forms a separate scale-space for the search window of each contour element v_i of a snake V. Using an information theoretic approach, we then analyze the behavior of the external energy under the scale change to come up with a set of different sized square shaped segments of the search windows. We apply a special dynamic programming optimization[1] using the centroids of these segments as the possible positions for optimized contour elements v_i. The resulting contour is used as the initial contour position for the same kind of optimization with a smaller scale-space representation and smaller search window sizes. The process continues until the segments of the search windows correspond to an original image pixel, after which no segmentation is meaningful.

3.1 Analyzing the Underlying Images and the Segmentation

The scale-space for a search window is constructed by a repeated convolution of the search window with a Gaussian kernel of increasing standard deviation σ sampled at discrete intervals. Given a search window W_i, we construct the scale space $L_i(x, y; \sigma)$ by

$$L_i(x, y; \sigma) = g(x, y; \sigma) * W_i(x, y) = \int_\alpha \int_\beta \frac{1}{2\pi\sigma^2} e^{-\frac{\alpha^2+\beta^2}{2\sigma^2}} W_i(x-\alpha, y-\beta) d\alpha d\beta \quad (7)$$

where $L_i(x, y; 0) = W_i(x, y)$ and $g(x, y; \sigma)$ is the Gaussian kernel with standard deviation σ. Each sample of σ is called a level of the scale-space. Levels are numbered starting from 0, which is the original image level. The scale of the level l is represented by σ_l.

We first form all scale-spaces L_i, $i = 1..n$, where n is the number of contour elements. Then, we segment each search window W_i by analyzing the behavior of the external energy with respect to change in σ. In other words, we like to know how the external energy changes in various areas of the search window if the underlying image is simplified by increasing σ in the scale-space. If the external energy starts to behave differently, we conclude that the corresponding segment of that area should be chosen smaller in order to be able to reflect the behavioral change better in the final segmentation. On the other hand, if the external energy behaves the same between the scale changes, we conclude that a larger segment for the corresponding area should not decrease the external energy related information in the final segmentation. We prefer larger segments in terms of efficiency because larger segments means less number of segments in a search window. This segmentation process addresses all the problems of the segmentation that we mentioned before.

In order to measure the behavioral change of the external energy with respect to scale σ, we use an information theoretic approach. Let $s^k_{i,j}$ be the j^{th} segment of the search window W_i defined on image $L_i(x, y; \sigma_k)$, which is the k^{th} level of the scale-space L_i. We can measure the amount of external energy related information, $H(s^k_{i,j})$, by the Shannon entropy.

$$H(s^k_{i,j}) = \sum_x \sum_y -p(s^k_{i,j}(x, y)) ln(p(s^k_{i,j}(x, y))) \quad (8)$$

where

$$p(s^k_{i,j}(x, y)) = \frac{E_{Ext}(s^k_{i,j}(x, y))}{\sum_u \sum_v E_{Ext}(s^k_{i,j}(u, v))}.$$

Similar types of information theoretic approaches were used in many scale-space studies by a number of researchers including Niessen *et. al.*[11] and Jagersand[7].

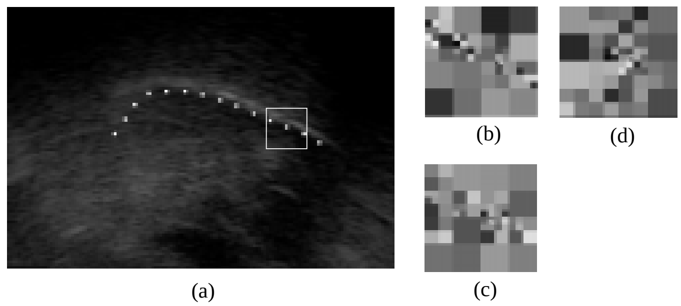

Fig. 3. Segmentation of the search window using different external energies. Please see the text for details.

We can measure the Shannon entropy of the same segment on the immediate upper level $k+1$ by $H(s_{i,j}^{k+1})$. Finally, we get the normalized measure of the behavioral change of the external energy with respect to change in σ by

$$D(s_{i,j}^{k}) = \frac{H(s_{i,j}^{k+1}) - H(s_{i,j}^{k})}{H(s_{i,j}^{\infty})}. \tag{9}$$

Larger the value of $D(s_{i,j}^{k})$, smaller the size of the segment should be.

The details of our segmentation is as follows. For a given m by m search window W_i, we form the scale-space L_i up to level l. The elements of the image at level k of this scale-space can be reached directly by $L_i(x, y; \sigma_k)$. We then form a set of segments S_i with four initial segments at the scale-space level $l-1$.

$$S_i = \{s_{i,1}^{l-1}, s_{i,2}^{l-1}, s_{i,3}^{l-1}, s_{i,4}^{l-1}\}. \tag{10}$$

Each of these segments are $m/2$ by $m/2$ and they are not allowed to overlap. In other words, we segment the scale-space level $l-1$ into four equal sized squares. We then choose the r^{th} segment $s_{i,r}^{l-1}$ in S_i that gives the largest value for Equation (9). This means we are choosing the segment that has the highest behavioral change with respect to change in scale. $s_{i,r}^{l-1}$ is removed from the set S_i and we add four new segments to S_i that are all $m/4$ by $m/4$ and are defined on scale-space level $l-2$ at the position of $s_{i,r}^{l-1}$ without any overlapping. The process continues by removing the segment producing the largest behavioral change value and adding four new square segments defined on the immediate lower scale level. This process continues until the number of segments in S_i reaches a user determined value.

Figure 3-a shows a midsagital ultrasound image of the tongue with the initial contour points superimposed. Figure 3-b shows the search window of the marked contour element segmented using an image intensity based external energy. Figure 3-c shows the same window segmented using the external energy defined by Equation (3). Figure 3-d shows the same search window segmented using an external energy that is sensitive to image gradient magnitude and the tangent angle of the contour at the marked contour element. Each segment is

$\sqrt{Number\ of\ Segments}$	64	32	16	8	4	2
Multiscale	8.1924	6.7940	5.3886	4.0486	2.6926	1.3669
Multiresolution	8.1924	6.8194	5.4500	4.0863	2.7301	1.3669
Constant Image	8.3178	6.9315	5.5452	4.1589	2.7726	1.3863

Table 1. Average Shannon entropy values for the segmentations by multiscale and multiresolution methods.

shaded with a random gray-level for visualization purposes. As the figures show, the final segmentations are different for different types of energy, which should be reflected in the DP optimization process by producing better contours.

Using the information theory, we can measure the information carried by a set of segments S_i by

$$H(S_i) = \sum_j -r(s_{i,j}) ln(r(s_{i,j})) \tag{11}$$

where $r(s_{i,j}) = E_{Ext}(\overline{s_{i,j}})/\sum_v E_{Ext}(\overline{s_{i,v}})$, $s_{i,j}$ is the j^{th} element of the segment set S_i and $\overline{s_{i,j}}$ is the average gray-level value of the segment $s_{i,j}$. In order to demonstrate that our scheme produces sets of segments that have more external energy related information, we performed experiments on medical images by measuring the information of the produced segment sets using Equation (11). We then measured the information produced by the equal sized square shaped segmentation(Figure 2) of the usual multiresolution methods using the same formulation. We also measured the information produced by segmenting a constant gray level image, which has the least possible information. Notice that, the type of segmentation does not matter for the constant image because the resulting information produced by Equation (11) would be the same. Experiments were performed on the ultrasound image shown in Figure 3-a, by taking 64 by 64 search windows of each contour element and by segmenting each search window using our multiscale method and using the multiresolution method. Finally, for each segment set, we measured the information produced and took the average. Table 1 shows these average information values for our multiscale method and for the multiresolution method. As the table shows our method carries more information than the multiresolution methods because the difference between multiscale values and the constant image values are greater than the difference between multiresolution values and the constant image values. Figure 4 shows this visually where we normalized the average information values by dividing it with the constant image information value. As expected, both methods produce the same information amount where the number of segments is 64^2. This is because each segment corresponds to an original image pixel and both methods produce the same segmentation. Similarly, both methods produce the same information value where the number of segments is 2^2. It is because our multiscale method initializes the segment set S_i with equal sized segments as in Equation (10). Figure 4 also shows that our method carries much more external energy related information where the number of segments is around 16^2, which is the most widely used case.

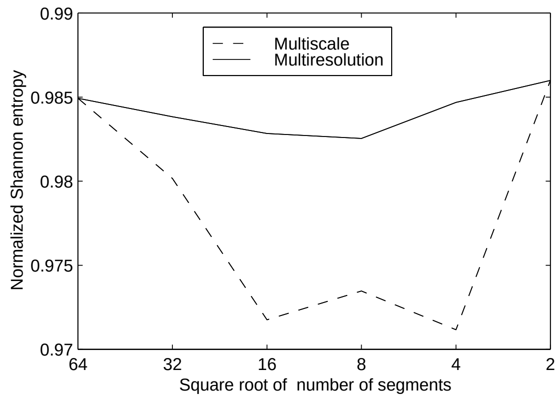

Fig. 4. Normalized average Shannon entropy values for Table 1.

Although there are major differences, our segmentation method shows resemblance to quad-tree type segmentation methods[9]. Our method uses scale-space techniques to analyze the behavior of the external energy with respect to change in the scale to decide which segment to divide. Quad-trees on the other hand do not pay attention to scale changes. They simply use the variance in the image to decide which segment to divide.

4 Experiments

We tested our system by performing experiments on medical images, which are known to be very problematic for contour analysis. In order to show the performance of our system we compared the contours of the multiresolution methods with our multiscale method at the highest level of the pyramid. This is because we know that final results are corrected by the contour optimizations at the lowest levels, which are the same for our method and for the multiresolution methods. This paper presents two of the test sets that we used.

The first test set is a sequence of midsagital ultrasound images of the tongue during speech. In addition to the usual ultrasound imaging problems, open contours and application specific problems makes contour analysis of these sequences very difficult[2]. Figure 5-(a) shows the tracked contours for four frames by our system. Figure 5-(b) shows the tracked contours produced by the multiresolution method. Our method spends about 28 seconds of CPU time for each contour. The multiresolution method spends about 43 seconds. We compared these contours against the ground truth obtained by a non-multiresolution dynamic programming system, which guarantees to give optimal results. The comparison is done by measuring the distances between the corresponding contour element positions of the two contours. Our system produced an average of 6.12 pixel difference. The other method produced an average of 12.91 pixel difference.

The experiments on ultrasound images confirmed the accuracy of our system. Next, we like to see if we can achieve the same performance using a smaller

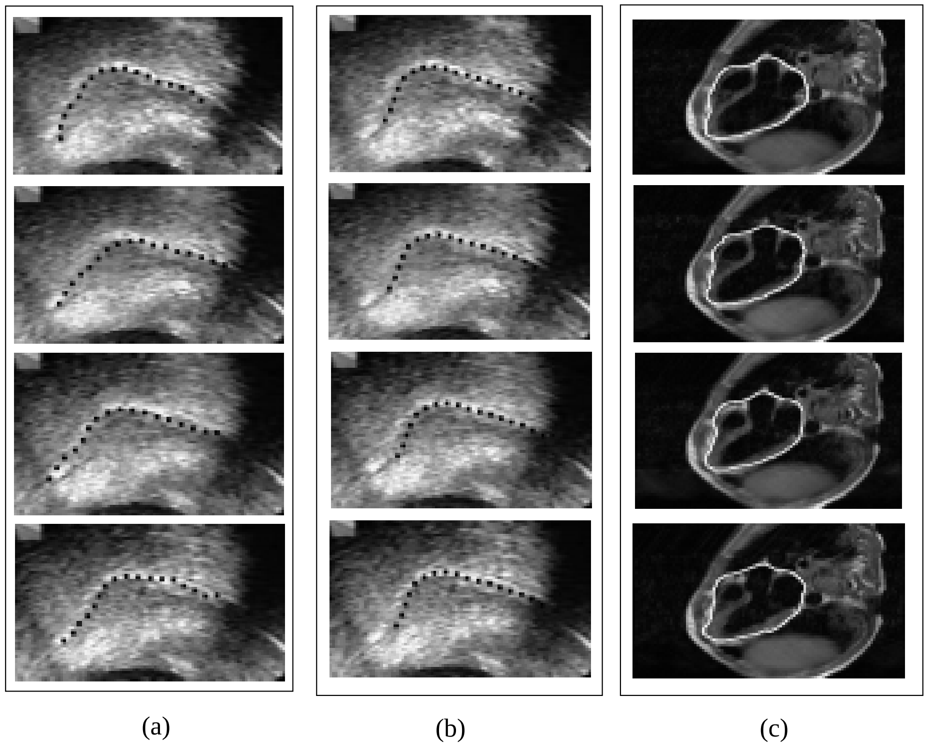

Fig. 5. Tracking results from (a) Multiscale method (b) Multiresolution method. (c) Results of our system applied to an MRI heart image sequence.

number of segments, which would result in faster execution times. For this experiment we used a sequence of four frames of a right anterior oblique (RAO) view contrast ventriculogram (CV) images from a normal human subject. Since they are less noisy and the contours are closed, MRI images are easier to analyze. Manually detected contours were used for verification of our results. First we run the multiresolution method on the sequence using the first frame's manually detected contour as the initial contour positions for all the frames in the sequence. We used 64 points for each 32 by 32 search window. Each contour extraction took an average of 26.78 CPU seconds. The average contour element difference with the manually detected contours was 2.88 pixels. We then did the same experiment using our multiscale method. We used only 25 points(segments) for each search window to speedup the optimization process. The average contour element difference with the manually detected contours was 2.87, which is almost the same with the multiresolution method. However, we saw a big difference in the time taken for each contour optimization: it took only an average of 7.42 CPU seconds for each contour extraction process with our method. Figure 5-(c) shows the tracking results from our system.

5 Conclusions

We presented a new multiscale approach for dynamic programming based deformable contour minimization. The system introduces a number of novel ideas that would be valuable for discovering new uses of scale-spaces in model based analysis of 2D and 3D images and image sequences. We confirmed through the experiments that the new method can achieve faster optimization times and performs better than the current dynamic programming optimization methods that are based on multiresolution techniques.

Our method reduces the number of possible locations that a contour element can take, dramatically shortening the execution time of the optimization. Although multiresolution methods use the same idea, our multiscale approach uses a scale-space approach to come up with a better set of candidate positions that makes the optimization process faster and increases the performance. Using information theory, the system analyzes the behavior of the external energy with respect to the scale change. This analysis gives us information on how to segment the underlying images so that reduced number of candidate positions carries more external energy related information. A previously developed dynamic programming method[1] is used to optimize the contour energy on these points to produce the final contours. The system can be generalized to different deformable contour and deformable model applications by changing the internal and external energies and the segmentation algorithm to fit the specific needs of the application.

References

1. Yusuf Sinan Akgul and Chandra Kambhamettu. A new multi-level framework for deformable contour optimization. In *CVPR99*, volume II, pages 465–470, 1999.
2. Yusuf Sinan Akgul, Chandra Kambhamettu, and Maureen Stone. Extraction and tracking of the tongue surface from ultrasound image sequences. In *CVPR*, pages 298–303, 1998.
3. A. A. Amini, T.E. Weymouth, and R.C. Jain. Using dynamic programming for solving variational problems in vision. *PAMI*, 12(9):855–867, 1990.
4. P.J. Burt and E.H. Adelson. The laplacian pyramid as a compact image code. *IEEE Trans. on Commun.*, 31(4):532–540, April 1983.
5. D. Geiger, A. Gupta, L. A. Costa, and J. Vlontzos. Dynamic programming for detecting, tracking, and matching deformable contours. *PAMI*, 17:294–302, 1995.
6. D. Geiger and J.E. Kogler, Jr. Scaling images and image features via the renormalization group. In *CVPR*, pages 47–53, 1993.
7. M. Jagersand. Saliency maps and attention selection in scale and spatial coordinates: An information theoretic approach. In *ICCV95*, pages 195–202, 1995.
8. M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. In *ICCV*, pages 259–269, 1987.
9. A. Klinger. Pattern and search statistics. In J.S. Rustagi, editor, *Optimizing Methods in Statistics*. Academic Press, 1971.
10. T. Lindeberg. *Scale-Space Theory in Computer Vision*. Kluwer Academic Publishers, 1994.
11. W.J. Niessen, K.L. Vincken, J.A. Weickert, and M.A. Viergever. Nonlinear multiscale representations for image segmentation. *CVIU*, 66(2):233–245, May 1997.