

# A new segmentation method for MRI images of the shoulder joint

Nhat Tan Nguyen      Denis Laurendeau  
Computer Vision and Systems Laboratory  
Laval University  
Quebec, Canada  
{ntnguyen, laurend}@gel.ulaval.ca

Alexandra Branzan-Albu  
Computer Vision Laboratory  
University of Victoria  
Victoria, Canada  
aalbu@ece.uvic.ca

## Abstract

*This paper presents an integrated region-based and gradient-based supervised method for segmentation of a patient magnetic resonance images (MRI) of the shoulder joint. The method is noninvasive, anatomy-based and requires only simple user interaction. It is generic and easily customizable for a variety of routine clinical uses in orthopedic surgery.*

## 1 Introduction

The shoulder joint has the widest range of motion of any joint in the human body. This joint is actually a complex set of bones and soft tissues. It contains several bones, with numerous articular surfaces, ligaments, tendons, and neurovascular structures. Shoulder pain, with or without dysfunction, is a common problem, but the clinical signs are often subtle. Conventional imaging (plain film radiography, scintigraphy, conventional tomography, and arthrography) frequently proves to be inadequate or equivocal in determining the source of the pain. With their high-resolution, high-contrast, and multi-planar imaging capabilities, magnetic resonance imaging (MRI) and computed tomography (CT) can be used to overcome this problem.

In many cases, the source of shoulder pain is due to dislocation or luxation of the humerus and scapula bones. In an attempt to assist radiological diagnosis we developed a segmentation algorithm for the detection of the planar contours of the humerus bone in a sequence of successive slices of a three-dimensional (3-D) image of the shoulder. After detection, a stack of planar contours is obtained for the bone, which describes its boundary surface and from which the orientation and location of the bone, can be determined. Our aim is to use this information for visualization and also for quantification of the location and orientation of the humerus head bone when the arm is placed in different postures.

Most segmentation methods can be divided primarily

into region-based and gradient-based approaches. Region-based methods [4, 5, 7] rely on the homogeneity of spatially localized features such as gray level intensity, texture and other pixel statistics. Homogeneity does not necessarily mean identical pixel values within a particular region; it rather means that the variation within a region is of a smaller extent than that between regions. The advantage of such methods is that since they rely directly on the gray level image, they are less sensitive to noise than methods that use derivative information. Also, if the high frequency information in an image is either missing or is unreliable, the segmented images remain relatively unaffected.

Shape variations, on the other hand, can be better handled using a deformable boundary finding framework [1, 2, 6] when we consider such variations to be generally around an average shape and such information can easily be incorporated as priors.

Our ultimate goal is to develop a framework for integrating boundary finding and region-based segmentation. This would lead to a system where the two modules would operate respectively so that at each step the outputs of each module are updated using information from the outputs of both the modules from the previous iteration. The main objective we present in this paper is aimed at using the results of boundary finding to assist region-based segmentation. Unlike most of the other existing methods [11, 3], it integrates boundary-based segmentation with region growing rather than edge detection, thereby making it resistant to the problems of false and broken edges while at the same time having well localized boundaries.

## 2 Description of algorithm

### 2.1 Choosing the best frame in sequence

In order to choose the best frame in a sequence of images, one image in the sequence must be selected where the shape of the object appears clearly (see Figure 1 for diagram of the approach). The contrast between object and

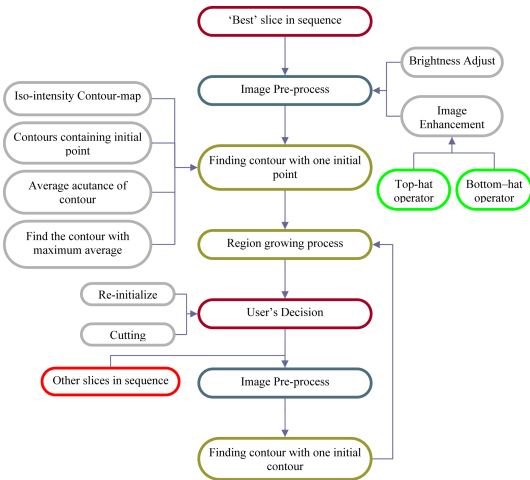


Figure 1. Diagram of the whole algorithm

background should be significantly different. In our case, the frame in the middle of the sequence is usually the desired one.

## 2.2 Applying brightness adjust and image enhancement

Since MR images can be obtained using several different methods, the brightness of the resulting images may vary. Thus, image brightness must be adjusted. In general, the manipulation of gray scale images in terms of a transfer function relating the stored brightness value for each pixel to a displayed value. In some cases, it is advantageous to use transfer functions that are not one-to-one: several stored values are displayed with the same brightness values, so that other stored values can be spread further apart to increase their visual difference. A non-linear relationship can expand one portion of the gray scale range while compressing another. The most common kinds of transfer functions are shown in Figure 2. These are curves of displayed vs. stored brightness following simple mathematical relationships such as power law curves.

$\text{Gamma}(\gamma)$  is in fact a rather mathematical entity, an expression of power law. If  $\gamma$  is less than 1, the mapping is weighted toward higher (brighter) output values. If  $\gamma$  is greater than 1, the mapping is weighted toward lower (darker) output values. The user must input the value of  $\gamma$  to adjust the brightness (as default  $\gamma = 1$ ).

After adjusting the brightness, the given image is enhanced through the application of top-hat and its inverse transformation. For the top-hat transformation, let us first consider  $f \in P_i$  to be a picture and structuring element  $B$  to be the unit disk. Now consider  $f - f_{\rho B}$ . This quantity is again a picture, composed of the residual from the opening by  $\rho B$  and  $f$ . It extracts topographical peaks, isth-

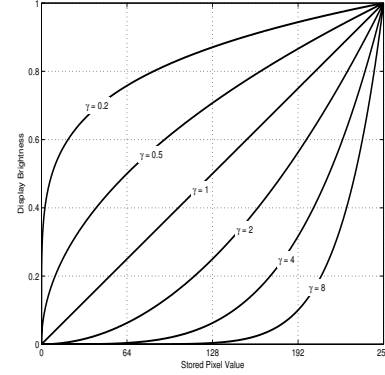


Figure 2. Examples of different transfer functions

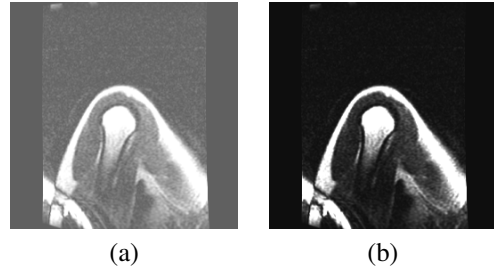


Figure 3. Manipulation of the brightness adjust. (a) An original image. (b) An image after the application of a positive gamma function ( $\gamma = 3$ )

muses and capes. More precisely, consider all of disk  $\rho B_y$  containing the point  $x$ . With each of them associate the sharpest drop from  $x$  to any point of  $\rho B_y$ . The largest of these sharpest drops is  $f - f_{\rho B(x)}$ . In the reciprocal representation  $X_i(f - f_{\rho B(x)})$  is the set of points  $x$  such that in every  $\rho B_y$  containing  $x$ , the maximum drop is equal to at least  $i$ . We obtain the relation:

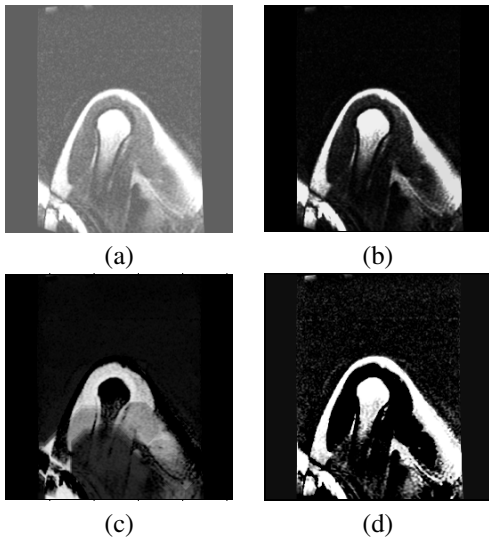
$$X_{i_0}(f - f_{\rho B}) = \bigcup_{i=0}^{i_{max}-i_0} [X_{i+i_0}(f)/X_{i+1}(f_{\rho B})] \quad (1)$$

$$\text{with } 0 \leq i \leq i_{max}$$

which represents the original algorithm given by Meyer [10], when he proposed the top-hat transformation. Since his pioneering work, thousands of experiments have proven the power of the algorithm for contrast extraction in pictures.

The top-hat image contains the topographical peaks, isthmuses and capes of objects that fit the structuring element  $\rho B$ . The bottom-hat transform, the inverse transform

of the top-hat transform, shows the gaps between the object and other parts of the image. In order to maximize the contrast between the objects and the gaps that separate them from each other, we add the top-hat image to the original image, and then subtract the "bottom-hat" image from the result. The impact of this process also reduces the noise around the object. The example in Figure 4 shows a T1 shoulder image (a) original. Image (b) illustrates the top-hat applied. Images (c) shows the bottom hat transform applied on original image. And (d) is the image after this enhancement. We can easily observe that the form of the humerus is clearer and sharper when compared to the original.



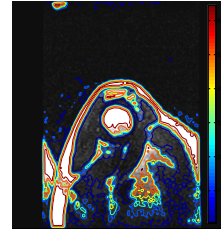
**Figure 4. Application of a top hat filter. (a) An original image. (b) Image after the application of the top hat filter. (c) Bottom hat applied image. (d) Image after enhancement**

### 2.3 Calculating the average acutance and finding the "best" contour

An iso-intensity contour map [12] provides the contours (256 levels) of the entire image (Figure 5). The contour map of the humerus is the subset of this image. To extract the iso-intensity contour map of the humerus, the user is asked to choose one point located in the humerus, called the reference pixel. This information is necessary to locate the humerus in the image. Then all iso-intensity contours containing that point are collected and a set of contours is built. Among these contours, the goal is to extract only one contour that represents the humerus exactly.

The humerus is usually brighter than the background. Then only the average acutance of the contours which con-

tain the reference pixel is determined. On the direction per-



**Figure 5. An image with its iso-intensity contour map**

pendicular to the contour at each point  $j$ , the acutance  $a_j$  is calculated. The acutance  $a_j$  of a point  $j$  on contour is then defined as

$$a_j = \sum_{i=1}^k |I_i - E_i| \quad (2)$$

where  $I_i$  is the value of the intensity of pixel within the contour and  $E_i$  is the value of the intensity of the pixel outside the contour the exterior one. And average acutance  $A$  of contour is

$$A = \frac{1}{N} \sum_{j=1}^N a_j \quad (3)$$

Now, the contour with the maximum average acutance  $A$  is the desired contour.

### 2.4 Using the previous contour to find contour on the successive frame

The contour that was found in the previous step, is now projected on to the adjacent frame, as shown in Figure 6. The normal line of each point on the contour is then drawn.

#### 2.4.1 The gradient of the contour

First an image from the contour is generated and a blur filter is applied. This provides an image as shown in Figure 8. After computing the gradient of this image, 2 images  $FX$  and  $FY$  are obtained which have the same size as the original image. Let  $FX$ ,  $FY$  be the gradient maps of the image.

$FX$  corresponds to  $\partial I / \partial x$ , the differences in the  $x$  (column) direction.

$FY$  corresponds to  $\partial I / \partial y$ , the differences in the  $y$  (row) direction. Normalizing  $FX$ ,  $FY$  provides:

$$(fx, fy) = \frac{(fx, fy)}{\max fx, fy} \quad (4)$$

Consider that  $(x_0, y_0)$  is the point on the contour. The segment of normal line of the contour at  $(x_0, y_0)$  is defined

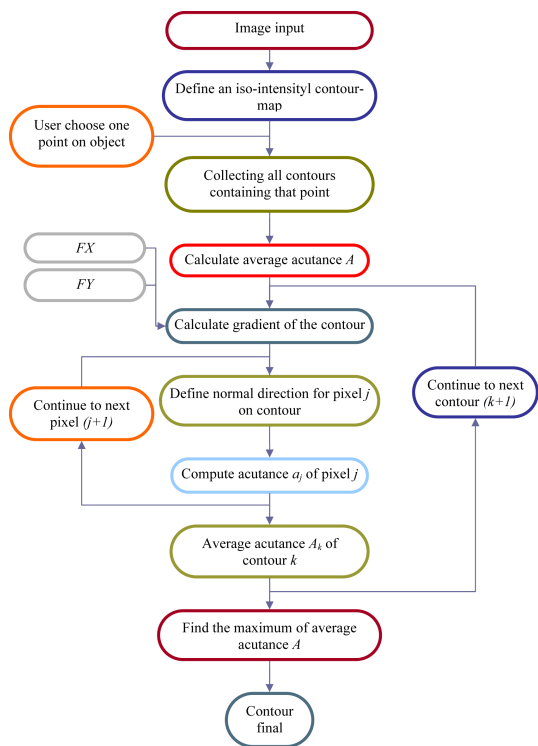


Figure 6. Diagram demonstrating the determination of the boundary with a given point procedure

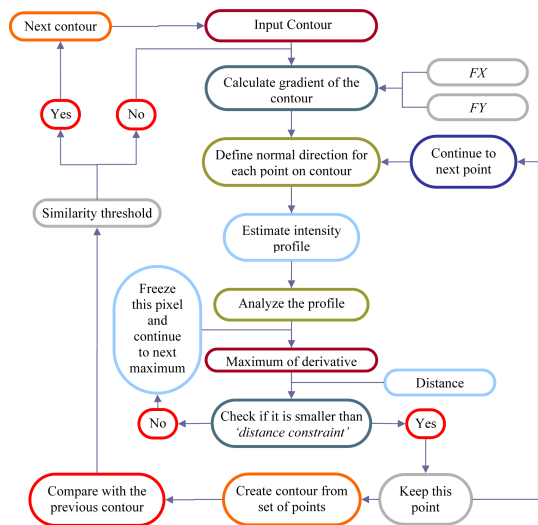


Figure 7. Diagram illustrating the procedure of finding the contour with a contour initial

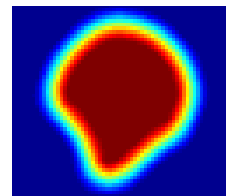


Figure 8. An image generated from a contour

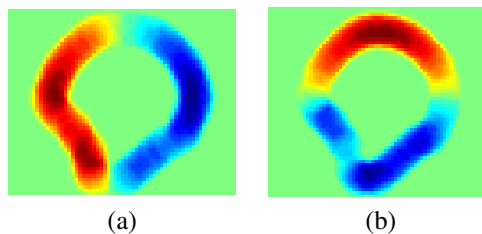


Figure 9. Gradient maps of image. (a) Row direction derivatives  $F_X$ . (b) Column direction derivatives  $F_Y$

by two points  $(x_1, y_1)$  and  $(x_2, y_2)$ :

$$x_1 = x_0 + d * f_x; \quad y_1 = y_0 + d * f_y \quad (5)$$

and

$$x_2 = x_0 - d * f_x; \quad y_2 = y_0 - d * f_y \quad (6)$$

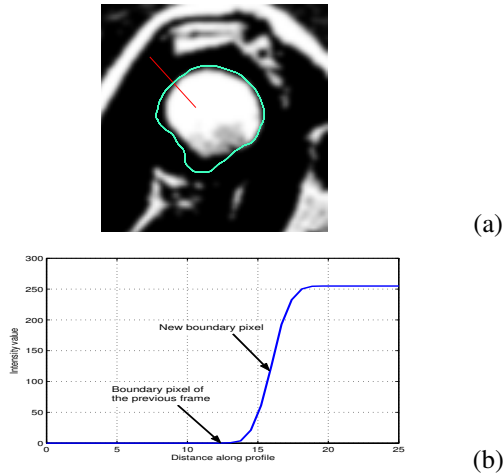
where  $d$  is the length of the normal. This method not only provides 8 possible directions but also gives a precise direction of the normal at each point on the contour.

#### 2.4.2 Analyzing the intensity along the normals of the contour

The profile of the image intensity along the normal line is now analyzed.

By calculating the derivative of the profile intensity, one obtains the slopes (minimums and maximums local). The frontier of the object will fall on the pixel in which the maximum slope is found.

Here, we put a constraint called the *distance constraint*. We suppose that the curvature of the boundary of the object is approximately uniform. Consequently, the distance between 2 found-points from 2 successive analyzes is not far. In the profile corresponding to the currently analysed point, there may be several local maxima of the slope. The local maximum which respects the *distance constraint* is chosen. When a point that should be the edge is identified, then the distance from this point to the previous point is verified. If this distance satisfies the *distance constraint*, then the point



**Figure 10. Analyze the image intensity along a segment. (a) Image of light object on a dark background with a normal section shown; (b) The profile of image intensity along a normal section**

is kept. If not, then the next lower slope is examined to find the other point.

### 2.4.3 Linking the points

The previous process leads to a set of data points. The method of fitting that set of data points with a  $B$ -spline curve will now be discussed. Consider  $n+1$  data points  $D_0, D_1, \dots, D_n$  which are to be fitted with a  $B$ -spline curve of degree  $p$ . First a set of parameters must be found that can "fix" these points at certain values. More precisely, if the data points are  $D_0, D_1, \dots, D_n$ , then  $n+1$  parameters  $t_0, \dots, t_n$  in the domain of the curve must be found so that data point  $D_k$  corresponds to parameter  $t_k$  for  $k$  between 0 and  $n$ . This means that if  $C(u)$  is a curve that passes through all data points in the given order, then  $D_k = C(t_k)$  for all  $0 \leq k \leq n$ . Thus seven data points (i.e.,  $n = 6$ ) and seven parameters must be found to set up the desired correspondence.

The simplest way to select parameters is using the uniformly spaced method. Suppose the domain is  $[a, b]$  and  $n+1$  uniformly spaced parameters are required. The first and the last parameters must be  $a$  and  $b$  because the curve must pass through the first and the last data points. Therefore,  $t_0 = a$  and  $t_n = b$ , since  $n+1$  points divide the interval  $[a, b]$  into  $n$  subinterval evenly,  $a$  and  $b$  included. The length of this interval is  $b - a$ , so the length of each subinterval must be  $(b - a)/n$ .

While the uniformly spaced method is simple, it is known to generate some unpleasant results. For example, when data points are not uniformly spaced, using uniformly

spaced parameters could generate erratic shapes such as big bulges, sharp peaks and loops. Fortunately, our data points from the previous process are quite uniformly spaced, this problem will be verified with the results at next step.

Once a set of parameters is obtained, a knot vector can be generated. For a  $B$ -spline curve of degree  $p$ ,  $m+1$  knots are required, where  $m = n + p + 1$ .

Thus, the first internal knot is the average of  $p$  parameters  $t_1, t_2, \dots, t_p$ ; the second internal knot is the average of the next  $p$  parameters,  $t_2, t_3, \dots, t_{p+1}$ .

Having generated a knot vector, the only missing part is a set of  $n+1$  control points. Suppose that  $C$  is the interpolating  $B$ -spline curve of degree  $p$ .

This  $B$ -spline curve has  $n+1$  unknown control points, since parameter  $t_k$  corresponds to data point  $D_k$ .

Since matrix  $D$  contains the input data points and matrix  $N$  is obtained by evaluating  $B$ -spline basic function at the given parameters,  $D$  and  $N$  are both known, and the only unknown is matrix  $P$ . The above is simply a system of linear equation with unknown  $P$ , solving for  $P$  yields the control points and the desired  $B$ -spline interpolation curve becomes available. In our experiments, the set of data points, the parameters, are equally or almost equally spaced and short spaced, the interpolation curves have to stretch a little longer for the shorter chords. This situation worsens with the higher degree curves because the higher degree curves provide more freedom to wiggle.

### 2.4.4 Smoothing

The response data is specified by input data  $S$  using the moving average method. The number of data points in the average (the span) is five.  $F$  is the smoothed response data. Note that the predictor data need not be specified if it is sorted and uniform.

The first four elements of  $F$  are given by

$$\begin{aligned} F_1 &= S_1 \\ F_2 &= (S_1 + S_2 + S_3)/3 \\ F_3 &= (S_1 + S_2 + S_3 + S_4 + S_5)/5 \\ F_4 &= (S_2 + S_3 + S_4 + S_5 + S_6)/5 \\ F_t &= \frac{1}{5} \sum_{i=t-2}^{t+2} S_i \end{aligned}$$

and continue to end.

### 2.4.5 Similarity

Given two contours defined by their curvilinear coordinates  $C_1(r_j)$  and  $C_2(s_i)$ , a metric distance,  $e(C_1(r_j), C_2(s_i))$ , between the two curves is defined. V. Chalana [9] proposed a methodology using the distance to the closest point (DCP)

for a point  $s_i$  to the contour  $C_1$  as

$$d(s_i, C_1) = \min_j \|r_j - s_i\| \quad (7)$$

The Hausdorff distance between the two contours is defined as the maximum of the DCP's between the two contours

$$e(C_1, C_2) = \max\{\max_i\{d(s_i, C_1)\}, \max_j\{d(r_j, C_2)\}\} \quad (8)$$

The closest point distance associates each point on both curves to a point on the other curve, and the Hausdorff distance finds the largest distance between the associated points.

Instead of using the DCP and the Hausdorff distance [1], the displacement vector is calculated base on the Euclidean distance transform on one contour to overlay of the second contour. Each point  $s_i$  on  $C_2$  has a normal and the intersection of this normal with contour  $C_1$  is determined. These points of intersection with the points  $s_i$  define a set of correspondence between the two contours. The distance from each point  $s_i$  on  $C_2$  to its correspondence point on contour  $C_1$  is calculated as

$$d_{C_1}(s_i, r_i) = \|s_i - r_i\| \quad (9)$$

Then the boundary is traced and the distance values are integrated.

This calculation is not symmetric, since the distances from  $C_1$  to  $C_2$  are not the same as  $C_2$  to  $C_1$ . Therefore a common average is derived by combining the two averages. The mean absolute distance, as opposed to binary overlap, does not depend on the object size.

This calculation is conducted iteratively for every frame. Typically the procedure converges in less than four iterations. Each time a contour is obtained from the iteration, it is compared to the previous contour, and a decision is made as to whether the analysis of this frame should be continued or if one should proceed with the next frame.

In order to do this, a threshold is given to stop the iteration. If the similarity that was calculated above is smaller than the threshold, the procedure will proceed to next frame. In our experiments, this threshold has been set equal to 1 pixel.

In addition to computing common average distance (as discussed above), a mean standard error of distance is also defined. If it is larger than 1 pixel, the procedure will be repeated irrespective of the value of similarity.

## 2.5 Region growing

The region growing algorithm has two objectives: to add missing pixels to the region and remove false pixels from the region. Having obtained the last contour, the pixel

aggregation procedure mentioned in [3] is then conducted with the seed pixels which are pixels on that contour. The pixel aggregation process iteratively not only adds to the region the neighboring pixels satisfying similarity constraints but also subtracts from the region the pixels and its neighborhoods which do not satisfy similarity constraints and stops when no candidate pixel satisfies this constraint.

These similarity constraints can be defined as follows by

$$\text{Highthreshold} = 0.7(\mu - t\sigma_n) \quad (10)$$

$$\text{Lowthreshold} = 0.65(\mu - t\sigma_n) \quad (11)$$

where  $m$  is the mean value of the region and  $s_n$  is the standard error of the mean,  $\sigma_n = \sigma/\sqrt{n}$ ,  $\sigma$  is the standard deviation and  $n$  is the number of pixels in the region. The  $t$  value comes from a table of  $t$  values (Student's  $t$  test). The following diagram illustrates the algorithm.

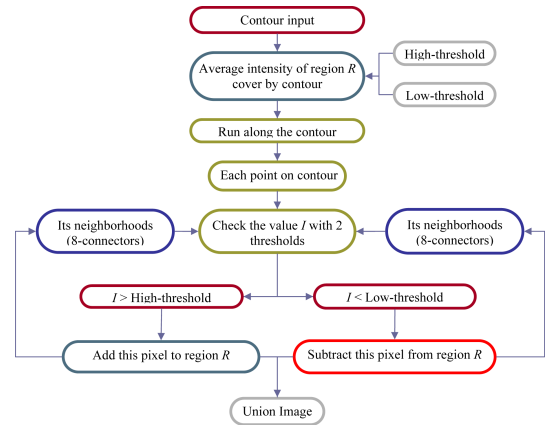


Figure 11. Region growing diagram

## 2.6 User's decision function

Occasionally, some unpleasant results are obtained, such as in the Figure 12.

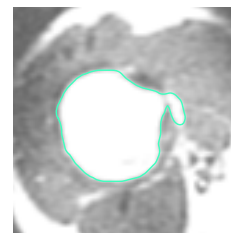


Figure 12. Example of an unexpected result

In this case, user input is required to overcome this problem. The user is provided with 3 options: "Re-initialize", "Cutting", and "Continue". "Re-initialize" - allows the user

to discard the information from the preceding frame and initialize the procedure at the present frame as described in section 2.3.

”Cutting” - will be described in the next section.

”Continue”- allows the user to accept the result and continue the process to the next frame.

The whole procedure is described by the illustration in Figure 13.

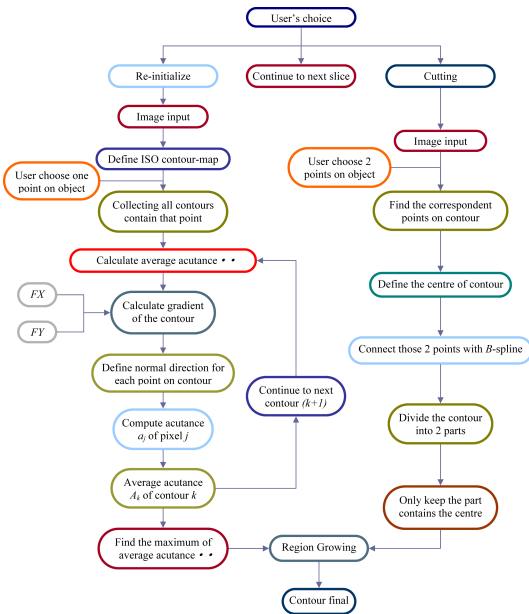


Figure 13. The user's function diagram

## 2.7 Cutting the contour with 2 points and linking these points with the spline

If the user wishes to cut the contour, he will be asked to pick 2 points on the contour. Here, we encounter one problem: with these 2 points, the contour will be separated into 2 parts; the computer does not know which part shall be kept as the final contour. Thus the center of the contour must be determined and only the part which contains the center must be kept.

The center of the contour is defined by

$$x_c = \frac{1}{N} \sum_{i=1}^N x_i \quad y_c = \frac{1}{N} \sum_{i=1}^N y_i$$

where  $N$  is the number of points on the contour and  $x_i$  represents the point  $i$  on contour. Then the two points are connected with the  $B$ -spline interpolation.

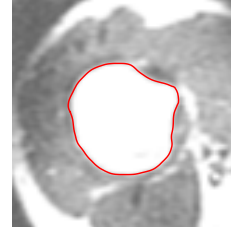


Figure 14. Contour after using cutting's function

## 3 Experimental results and evaluation

Experiments were carried out with the T1-weighted 1.5 Tesla images of the shoulder to verify the performance of the above mentioned method. This method is implemented in Matlab in Windows 2000 platform. The images were acquired on a General Electric Sigma Horizon open field MR system at the Intervention Magnetic Resonance Imaging Unit of the Centre Hospitalier Universitaire de Québec (CHUQ). The reference segmentations were taken by a radiologist.

For the purpose of evaluation, a traditional measure of volumetric overlap was used between two regions in the same way as used in [8]. Consider  $S$  as the region generated by our segmentation algorithm and  $R$  as the reference region. Images are analysed pixel by pixel to determine the false positives, false negatives, true positive and true negative pixel.

Let  $N_P$  be the number of false positive pixels,  $N_N$  the number of false negative pixels,  $N_R$  the total number of reference pixels and  $N_S$  the total number of subject pixels. Then consider the following correspondence measure:

$$C(S, R) = 1 - \frac{N_N + N_P}{N_R + N_S} \quad (12)$$

where  $N_S + N_N = N_R + N_P$ . The measure gives a score of 1 for perfect agreement and 0 for complete disagreement. The Figure 15 shows a graphical representation of the cor-

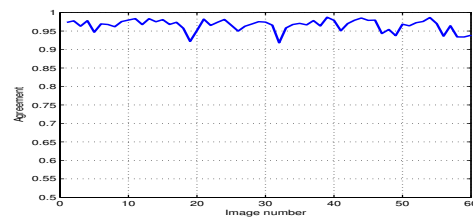
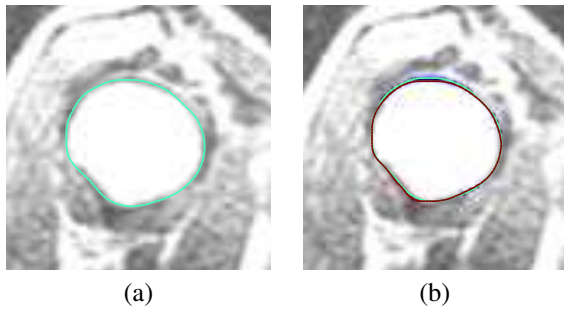


Figure 15. Correspondence measure between automatic and manual segmentation



**Figure 16. Result of humerus segmentation generated by experts and this algorithm. (a) A contour generated by this algorithm. (b) A contour generated by an expert**

respondence measure computed for a whole database. As demonstrated in this figure, the results of the algorithm and the human tracings are in close agreement. The average value of the measure computed over the entire database is 96.32%. Figure 16 demonstrates the result of the final contour of the humerus generated by the algorithm and experts alternatively.

## 4 Conclusion

The algorithm proposed here has the ability to perform on images with no specific symmetry and it can also isolate the bony region of MRI slices with a high level of noise, which will prove useful in many tasks of early vision. The experimental results indicate that this method yields excellent performances when compared to manual segmentation performed by radiologists.

While this work has focused on a 2D approach conceptually, future work will involve carrying the approach over to 3-D.

## References

- [1] C. Alberola-López, M. Martín-Fernández, and J. Ruiz-Alzola. Comments on: A methodology for evaluation of boundary detection algorithms on medical images. *IEEE Trans. Med. Imaging*, 23(5):658–660, May 2004.
- [2] J. Batista and R. Freitas. An adaptive gradient-based boundary detector for mri images of the brain. In *Proc. of the 7th Intl. Conf. on Image Processing and Its Applications, Manchester, UK*, pages 440–444. IEE, July 1999.
- [3] A. Branzan-Albu, D. Laurendeau, and C. Moisan. Tumour detection in mr liver images by integrating edge and region information. *Modelling & Simulation for Computer-aided Medicine and Surgery*, INRIA Rocquencourt:12–15, November 2002.
- [4] A. Chakraborty, L. H. Staib, and J. S. Duncan. An integrated approach to surface finding for medical images. *Proceedings of the 1996 Workshop on Mathematical Methods in Biomedical Image Analysis*, pages 253–262, March 1996.
- [5] A. Chakraborty, M. Worring, and J. S. Duncan. On multi-feature integration for deformable boundary finding. In *Proc. Int. Conf., Comput. Vision, ICCV*, pages 846–851, June 1995.
- [6] V. Chalana and Y. Kim. A methodology for evaluation of boundary detection algorithms on medical images. *IEEE Trans. Med. Imag.*, 16(5):642–652, October 1997.
- [7] R. Chellappa. Two-dimensional discrete gaussian markov random field models for image processing. In *Proc. SPIE, Digital Image Processing Applications, Ying-wei Lin; Ram Srinivasan; Eds.*, volume 1075, page 336, April 1985.
- [8] G. Gerig, M. Jomier, and M. Chakos. Valmet: A new validation tool for assessing and improving 3d object segmentation. *Proc. MICCAI Conf., Medical Image Computing and Computer-Assisted Intervention, Utrecht, the Netherland*:516–523, October 2001.
- [9] D. K. Hodge, C. F. Beaulieu, and G. H. T. III. Dynamic mr imaging and stress testing in glenohumeral instability: Comparison with normal shoulders and clinical/surgical findings. *Journal of Magnetic Resonance Imaging*, 13:748–756, 2001.
- [10] F. Meyer. Contrast feature extraction. *Proc. 2nd European Symp. Quantitative Analysis of Microstruct.*, Materials Science, Biology, and Medicine, October 1977.
- [11] B. Serra and M. Berthod. Optimal subpixel matching of contour chains and segments. In *IEEE Proc. Int. Conf., Comput. Vision, ICCV*, pages 402–407, June 1995.
- [12] S. Shiffman, G. D. Rubin, and S. Napel. Medical image segmentation using analysis of isolable-contour maps. *IEEE Trans. Med. Imag.*, 19(11):1064–1074, November 2000.
- [13] M.-E. Tremblay, A. Branzan-Albu, D. Laurendeau, and L. Hébert. Integrating region and edge information for the automatic segmentation for interventional magnetic resonance images of the shoulder complex. In *Proceedings of the 1st Canadian Conference on Computer and Robot Vision (CRV'04)*, pages 279–286, London, Ontario, May 2004.