AUTOMATIC CORONARY ARTERY WALL CONTOUR DETECTION

L Moura[1] and R Kitney[2]

ABSTRACT – The paper describes an image processing technique for the detection and labelling of arterial wall contours from cross-sectional slides in pathology. A gradient operator is applied to the digitised image and generates points of interest at likely edges. These points are then used as starting points for a contour follower algorithm leading to the formation of one-pixel-wide line segments. Heuristic search is used to locate closed sets of segments. A cost-funtion is used which is minimised by a circle so that the algorithm tends to find roundish structures. Further processing involves closed curve filtering and contextual contour labelling.

Results have shown that the method's strength lies in its ability to correctly find the wall contours even when major gaps are present, provided that some long segments are formed. The method is suitable for locating and labelling other round-shaped structures such as blood vessel cross-sections and cell outlines.

INTRODUCTION

Coronary occlusion is a common cause of myocardial infarction and is often fatal. Ischaemic heart disease was responsible for approximately 160,000 deaths in England and Wales in 1985, nearly 27% of all deaths that year (OPCS,1987).

Although much is known about the pathogenesis of coronary artery occlusion, there are some aspects yet to be unveilled (Davies, 1984, Davies, 1985, Hangartner, 1986). Investigation of coronary artery pathology may be performed by examining arterial cross-sections under the optical microscope. The biological material is prepared so that some particular types of tissue will be stained (Gray, 1973).

Occlusive disease is caused by atheromatous plaques which form underneath the intima. Enlargement of these plaques leads to constriction of the arterial lumen — a condition known as *stenosis*.

The degree of stenosis, measured as reduction in lumen size, has been a key concept for the study of arterial diseases in general. Methods for estimating cross-sectional surface area have varied from counting squares enclosed by outlines (Sissons, 1963) to paper-weighing techniques (Bell, 1967), and from planimeters (Aherne, 1971) to image processing (Gore, 1979, Hougardy, 1976). Understandably, computer techniques have been increasingly used in the last few years. Nevertheless, whichever methods are used they concentrate essentially on the slide-by-slide study of arterial segments.

[1] Coordenadoria de Informática - InCor HCFMUSP, São Paulo

[2] Biomedical Systems Group - Imperial College, London, UK.

PRELIMINARY CONSIDERATIONS

In this paper we describe the algorithm for automatic arterial wall contour detection that we have developed as part of a computer system for the reconstruction of coronary artery segments from series of cross-sectional slices (Moura, 1988).

An example of such a series can be seen in Figure 1. It consists of ten slides taken at equally spaced depths. This series of images has been selected as very representative of the type of data this work is concerned with.

All the coronary artery slides used in this paper have been obtained using the procedure described in detail in (Moura, 1988). Basically, the coronary arteries are dissected from hearts obtained at autopsy, fixed, processed and then thinly sliced using a microtome. Each slice is then attached to a rectangular piece of clear glass, stained, and covered with another piece of glass, smaller and much thinner.

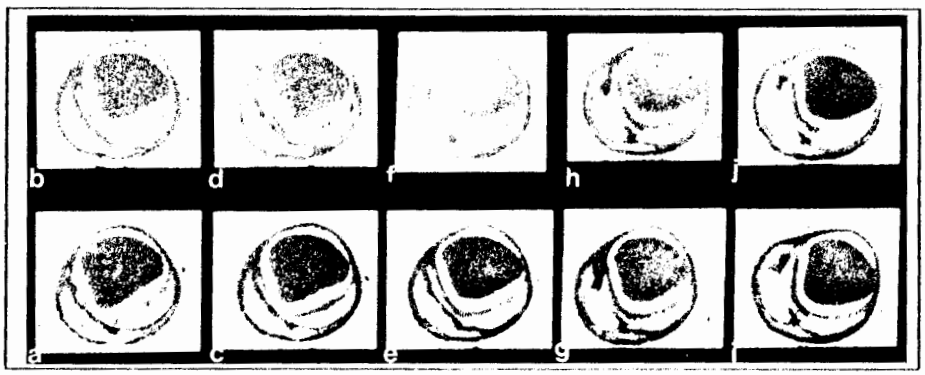

Figure 1 A Series of Cross-Sections. *The images from (a) to (j) are images of Coronary Artery cross-sections after being digitised.*

A glance at the slides in Figure 1 reveals that there are several levels of detail in the vessel structure. Although these details can convey important information to the pathologist's eye and the inner compartments may be important for some studies (Davies, 1984, Hangartner, 1986), this work will be restricted to the structures defined by the external and internal arterial walls.

The reasons for this restriction are that: *a)* wall contours are more tractable than general, sometimes very fuzzy, compartments, *b)* arterial walls are the most important arterial structures and will always be the first structures to be searched for, *c)* in functional terms blood flow through the vessel is critically dependent on the lumen size, and *d)* the level of detail the present computer system can represent is resctrited by the 3D processor. That resolution is below that required to represent finer details accurately.

DATA ACQUISITION.

Each slide is photographed under the optical microscope using a 35-mm camera and 100 ASA black and white negative film. The optical magnification is chosen in such a way that the largest arterial diameter in the series is nearly half as large as the visual field diameter. This allows that some space be left between the arterial wall and the photograph margins. Once the magnification is chosen it is kept constant throughout the series. In order to have a spatial reference for later scaling, a microscale is photographed at that same magnification. The distance between slices and the specimen's reference number are also recorded.

The 35-mm negatives are enlarged to the size of 5"x7" on glossy paper and the photographs are digitised into the computer by means of a Panasonic 1600 camera with a Vidicom cathod-ray tube. Although the camera's resolution is of 770 columns and 575 lines, these are decimated so that the usable final picture consists of 222^2 pixels at 8-bit depth resolution. The reason for decimation is to make the image fit the computer system's standard image size, as described by (Moura, 1988).

PRE-PROCESSING

Once the images have been digitised into the computer they are pre-processed in order to enhance image quality. In this work pre-processing consists basically of two steps: background removal and histogram strecth.

Background Removal.

Background removal is intended to compensate for variations in lighting that may have occurred when digitising the image or photographing the slides. It is assumed that any possible trend in illumination is linear in i and j. In other words, it is assumed that images digitised using the image acquisition subsystem are given by

$$I(i,j) = I_0(i,j) + B(i,j) \tag{1}$$

with

$$B(i,j) = a.i + b.j + c \tag{2}$$

where $I(i,j)$ is the digitised image, $I_0(i,j)$ is the original image, a,b and c are constants and $B(i,j)$ is the trend in background lighting.

If $B(i,j)$ could be estimated from the data it would then be possible to reconstruct $I_0(i,j)$ by making

$$I_0(i,j) = I(i,j) - B(i,j). \tag{3}$$

In this paper $B(i,j)$ is estimated using Least Squares fitting. The type of image under consideration is characterised by dark objects in a brighter background. Therefore, the bright regions of the image are the ones to be scanned when looking for trends in background illumination.

The method we have developed for background estimation consists of fitting a plane to the set of image pixels whose intensity is greater than a given threshold T. The plane is fitted in so as to minimise the square error, given by

$$E = \sum_{i=1}^{N}\sum_{j=1}^{N} \left[a.i + b.j + c - I(i,j)\right]^2, \quad \text{for} \quad I(i,j) \geq T, \tag{4}$$

where N is size of the image. The threshold T can be chosen arbitrarily but for the type of image under consideration this threshold has been set to the image's mean intensity value.

Histogram Stretch.

Once trends in background have been removed the histogram of intensities is stretched. The aim of this operation is twofold. Histogram stretch not only enhances visual contrast but it can also be used to bring together some data standardisation.

AUTOMATIC CONTOUR DETECTION

The input data to this stage are the pre-processed images. These are well-balanced and contrast-enhanced images in which coronary artery cross-sections appear against a lighter background. Images in Figures 1 are very representative of the type of images to be processed and must be analysed visually in order to create a feeling about the task to be executed automatically.

Here, it is more important to comprehend the general difficulties involved in characterising the contours for automatic detection than to be strictly accurate.

In these images, for example, the external and internal wall contours are *roundish* and the contour lines appear *reasonably clear* both in the normal images as well as in a gradient magnitude version of them — Figure 3. However, some *large gaps* may exist due to either superimposition of structures or image noise.

Any edge detection method which can take advantage of this *a priori* knowledge of contour characterisics is a good candidate for the application in view. A particularly strong boundary detection method when contours are of known shape is given by Heuristic Search.

The application of Heuristic Search is basically a three-step method. In the first, the original image is smoothed in order to remove unwanted details and a gradient operator is applied to it. The brighter the pixels in the resulting gradient magnitude image are the higher their likelihood of being part of edges. The second step explores this fact using bright gradient pixels as starting points for a contour-follower algorithm so that long one-pixel-wide edge segments are formed. The final step in the procedure is to connect the edge segments using some geometric criteria to form closed set of segments, called paths. A cost function is associated to each path and a contour is a minimum cost path. In order to succeed the cost-function must be chosen in such a way that it is minimised by contours with the desired forms — in the case in view circle-like shapes.

Smoothing and Gradient Operator.

Before a gradient operator is applied, the original image is smoothed in order to reduce the influence of high-frequency and low-amplitude noise.

A gradient operator is then applied to the resulting image in order to reveal probable edge pixels. The gradient operator in use has been described by Grattoni (Grattoni, 1985) and consists of two first-difference matrices A_x and A_y being convolved with the image. These two matrices are shown below.

$$A_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \text{and} \quad A_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{5}$$

They provide good estimators for the image's partial derivatives in x and y. These derivatives are given by

$$X = A_x * I \quad \text{and} \quad Y = A_y * I, \tag{6}$$

where the symbol '*' denotes convolution. The gradient magnitude is then computed as

$$M(i,j) = \sqrt{[X(i,j)]^2 + [Y(i,j)]^2} \tag{7}$$

and the gradient direction as

$$D(i,j) = arctan\left[\frac{Y(i,j)}{X(i,j)}\right]. \tag{8}$$

The gradient direction gives the direction of maximum local increase in pixel intensity. Therefore, the gradient direction at any image point tends to be perpendicular to the direction of any existing edge, a fact that will be employed in locating edges as well as testing the coherence of potential contour paths.

Segment Formation.

Once the gradient operator has been applied, segment formation is to take place. The gradient magnitude image is scanned for points of interest. This image is divided into square windows of L^2 pixels. If the intensity of the maximum pixel value within a window is greater than a certain application-dependent threshold T, then the pixel intensity and position is recorded in a list *POI* of points of interest.

The list *POI* is sorted in decreasing order so that the brigther pixels will be closer to the top of the list. The list *POI* is then scanned from top to bottom and every point of interest is used as a starting point for a contour follower algorithm which performs the operations depicted in Figure 2.

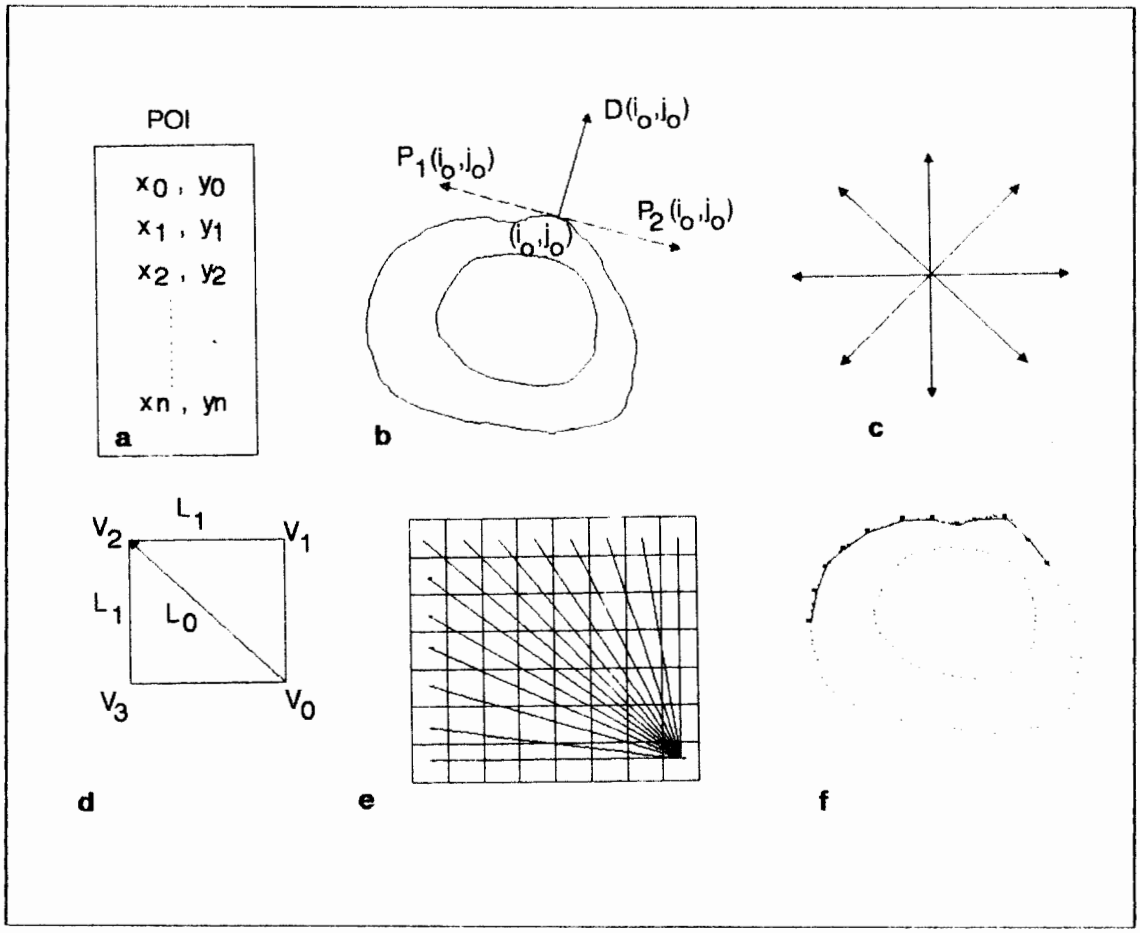

Figure 2 ***The Contour Follower Algorithm I.*** *A list of points of interest (a) is created on the grounds of intensity of the gradient magnitude pixels. These points are used as starting points for the search of edge segments whose direction tend to be perpendicular to the gradient direction at each pixel (b). The search, however, is done according to one of eight diretions (c). After the searching region is defined (d) straight lines departing from the current pixel are matched for edges (e). The end of each edgy straight line is used as the starting point for the next., thus forming long edge segments.*

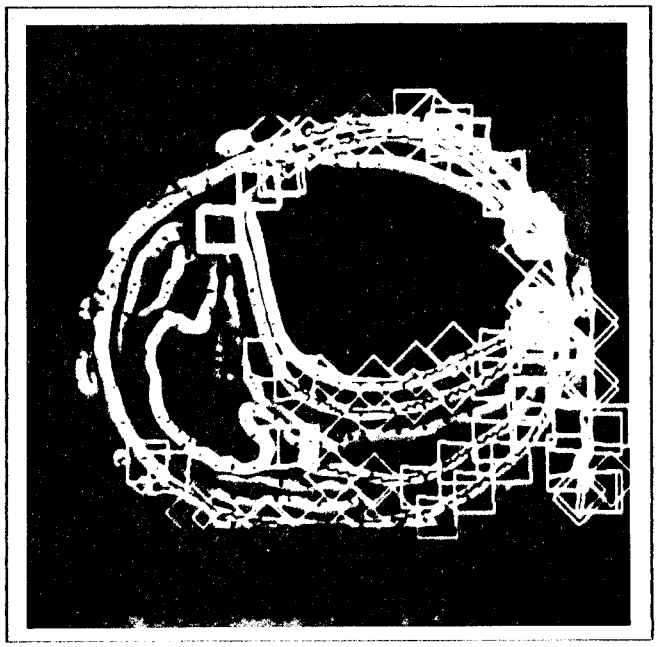

Figure 3
The Contour Follower Algorithm II. The dots are the interest points. The polygons define the search region and the lines are the segments thus formed.[3]

Heuristic Search (Minimum Cost Search).

The input to this stage are the segments found using the contour follower algorithm, the gradient magnitude and the gradient direction.

Heuristic search will be used to connect segments together and declare as a valid contour any *geometrically coherent* path whose cost is a minimum for the segments in it.

A path is a closed contour formed by segments and their connections. Although segments can be easily connected using straight lines we have decided to use Ferguson's curves (Ferguson, 1964) so that the connections tend to follow the segments behaviour, as described in (Moura, 1988).

For the purpose of this work, a *geometrically coherent* path is any path for which *a)* a path does not intersect any segment, *b)* the gradient direction along the path keeps always the same orientation (inwards/outwards) in relation to the path itself *c)* the angle formed between any two consecutive segments on the path is always less than an application-defined tolerance β and *d)* any two consecutive segments on the path are within a square window of size W.

Heuristic search can be seen as a graph search where the segments are the graph nodes and the graph branches are all the connections which lead to paths that satisfy conditions a to d above. A cost is associated to each path and a contour is any valid path with a minimum cost for the segments in it.

This cost-function has been chosen in order to favour the formation of paths whose segments are close to each other and also to favour the formation of *roundish* shapes. The distance-related cost component can be achieved by making

$$C_d = \frac{1}{P}\sum_{i=1}^{N_s}\sqrt{(Sx_i - Ex_{i-1})^2 + (Sy_i - Ey_{i-1})^2} \qquad (9)$$

where (Sx_i, Sy_i) are the coordinates of the starting point of the i-th segment in the path, (Ex_i, Ey_i) are the coordinates of the ending point of the i-th segment in the path, N_s is the number of segments in the path and P is the path's perimeter.

[3]Some photographs have been retouched manually in order to improve visibility in black and white.

AUTOMATIC CONTOUR LABELLING

The technique described in the previous section has been successfully used to detect the arterial wall contours. However, it has also detected a third contour. This contour — marked by an arrow in Figure 4 — is caused by coagulated blood which has shrunken during specimen preparation and is usually present in the type of slide under consideration. This contour must be disregarded as it does not bring any useful information.

This fact highlights the need for some sort of semantic contour classification which allows that the external and internal wall contours be properly recognised and labelled.

The approach we have devised for automatic contour labelling is based on two contour features, both contextual. The first is the gradient direction — inwards/outwards — in relation to the contour centre. We define this direction as inwards if the contour defines a structure which is lighter than its surroundings and outwards otherwise. The second contour feature is its position relative to other contours.

The gradient direction can be detected by analysing the straight line which goes from the *contour centre* to the contour pixel with the highest gradient magnitude value. The relative contour position can be extracted by straight lines that go from the bottom of the image to its top through each *contour centre*. If a number is assigned to each contour and if these numbers are recorded as the contours are intersected by the straight lines, the resulting *Contour Position Strings* will yield information regarding their relative position, as described in (Moura, 1988).

The two contextual features are combined so that the required contours can be found, as shown in Figure 5. The Contour Position String which describes the two arterial wall contours is *the one* which starts with an outward-gradient contour number and is followed by an inward-gradient contour number.

If more than one Contour Position String satisfy these conditions for two different sets of contours then the contours cannot be properly classified and the algorithm will have failed. In the current system, the program asks for operator assistance if it is unable to classify the contours.

The two procedures for contour feature extraction may not work for some markedly non-convex contours. Such contours, however, are most unlikely to happen in practice and they do not fit our previous assumption that wall contours are roundish. A more important factor for these technique's success is the choice of the *contour centre* required for both algorithms. If this point is too close to the edges it may cause errors in both cases.

We have developed a very reliable technique for fitting a circle to a set of points using Least Squares. This technique is described in full in (Moura, 1988) and is used to determine the contour centres for contour classification.

CONTOUR FILTERING.

Wall contours detected by the automatic procedure tend to be somewhat rough, as can be seen in Figures 5a and 5c. If the edges in the original image are noisy then the contour may be inaccurately detected, as in Figure 5a. However, as pointed out earlier in this paper, arterial wall contours are expected to be smooth. This property can be used to correct regional roughness.

The cost component designed to favour roundish shapes is based on the fact that the ratio between contour area and square perimeter is maximised by a circle. This component is given by

$$C_s = 1 - 4\pi \frac{A}{P^2}, \tag{10}$$

where P is the path's perimeter and A is the area enclosed by the path.

From its own definition it is easy to verify that the minimum C_s occurs for a circle — in which case it is null — and tends to 1 when the area defined by the shape tends to 0. Thus, C_s can be seen as a measure of *roundness*. A shape can be said to be *roundish* if C_s is greater than some given threshold.

The total cost function is computed as

$$C_t = a_1 C_d + a_2 C_s, \tag{11}$$

where a_1 and a_2 are application dependent weighing constants.

For each segment whose length is greater than a given S_{min}, the heuristic search algorithm finds all the possible paths and records in a list MCPATHS the one which leads to the minimum cost, if any. If this cost is greater than a certain C_{max} the path is disregarded.

After all the segments longer than S_{min} have been processed, the minimum cost path among all the minimum-cost paths in list MCPATHS is declared the first valid wall contour. All other paths which include any of the segments in this first wall contour are written off the list MCPATHS. The minimum-cost path among the remaining paths in the list is declared the second valid wall contour and the process is repeated until the list MCPATHS is empty.

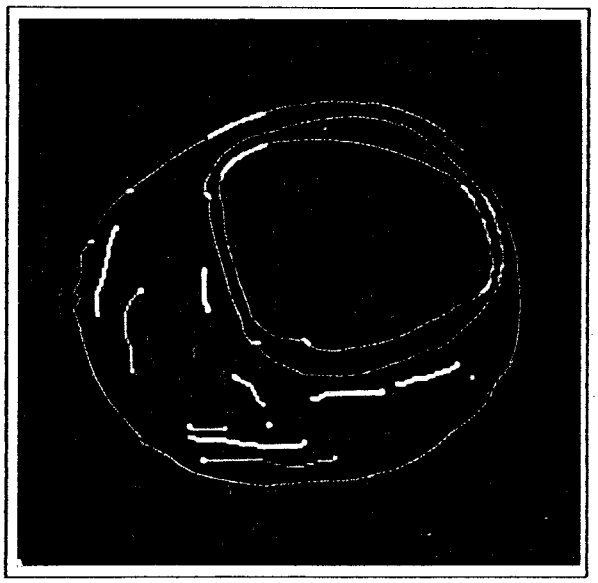

Figure 4
The Detected Contours.
The three detected contours are the minimum cost paths for the segments they use. The thicker lines on the contours represent Ferguson Curves connections.

Figure 4 shows an example of contours detected using the algorithm just described. Note that the algorithm has been able to bridge very wide gaps automatically.

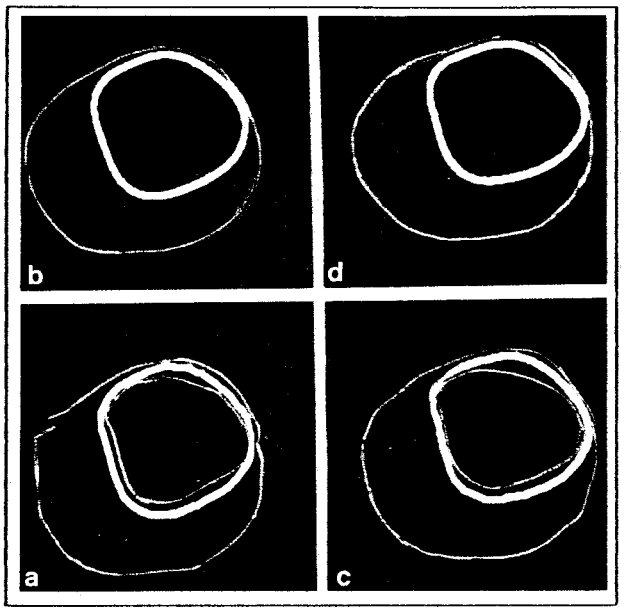

Figure 5
Contour Filtering.
This technique can be used to generate smooth contours (b) and (d), from the detected contours. It can remove regional roughness (b). Note that only the two relevant contours have been filtered.

We have developed a technique for closed contour filtering which produces smooth contours fitted to the original curve in the Least Squares sense. This contour filtering technique is fully described and discussed in (Moura, 1988).

The basic concept involved in this technique is that *well behaved* contours can be expressed in polar coordinates in terms of a contour centre and a function $R(\theta)$. If $R(\theta)$ is defined by piece-wise polynomials in θ then it is possible to force the contour and its derivatives to be continuous. Such a contour, continuous up to the second derivative is defined in (Moura, 1988) as a piece-wise polynomial-radius contour, or simply a *pw-contour*.

A pw-contour can be fitted to a set of data points so that a minimum square error is achieved. In order to standardise the data, every pw-contour is described by its centre (x_c, y_c) and a set of 360 radii R_i, $i = 0, 1, 2, \ldots 359$, taken at equally-spaced angles.

The pw-contour fitting can be seen as a non-linear filter whose output is a smooth representation of the input contour. There are two parameters which control the fit. The first is the number of intervals N_I into which the pw-contour is divided and the second is the order M_P of the polynomial used to represent $R(\theta)$. Although increases in M_P and N_I tend to improve the fit, N_I is more related to general fitting whereas increases in M_P tend to improve the fit at contour details.

The results shown in Figure 5b and 5d have been achieved with $N_I = 12$ and $M_P = 3$. These values have been set up for all contour filtering in this thesis.

CONCLUSIONS

In this paper we have described an implementation for the automatic detection of the arterial wall contours. These are assumed to be *roundish* or — more precisely — the *most roundish* structures in an image. We have described a cost-function which measures roundness and that is used to locate the wall contours.

This assumption restricts the classes of images the system will successfully process but on the other hand allows that wall contours with very wide gaps be detected.

The use of the contour filtering technique we have developed has allowed that the condition of roundness be monitored and automatically corrected. The use of pw-contours warrants that wall contours be smooth.

ACKNOWLEDGEMENTS

This work was partly funded by Fundação E J Zerbini, FAPESP and CNPq. Their financial support is hereby heartily acknowledged.

REFERENCES

01. Aherne et al. *Muscle Size Fibre In Normal Infants, Children And Adolescents.* Journal Of Neurological Science, 1971, (14), 171.

02. Bell CD and Conen PE *Change In Fibre Size In Duchenne Muscular Dystrophy.* Neurology (Minneapolis), 1967, (17), 902.

03. Davies MJ and Thomas A. *Thrombosis And Acute Coronary-Artery Lesions In Sudden Cardiac Ischemic Death.* New England Journal of Medicine, 1984, (310), 5, 1137-1140.

04. Davies MJ and Thomas AC. *Plaque Fissuring - The Cause Of Acute Myocardial Infarction, Sudden Ischaemic Death, And Crescendo Angina.* British Heart Journal, 1985, (53), 363-373.

05. Ferguson J.C. *Multivariate Curve Interpolation.* Journal ACM 1964, 11, 221–8.

06. Gore LF. *The Measurement Of The Microscope Image.* Medical Laboratory Science, 1979, (36), 63.

07. Grattoni P and Bonamini R. *Contour Detection Of The Left Ventricular Cavity From Angiographic Images.* IEEE Transactions On Medical Imaging, 1985, (MI—4), 6, 72-78.

08. Gray P. *The Encyclopaedia Of Microscopy And Microtechnique.* Van Nostrand Rheinhold Company, 1973.

09. Hangartner JRW et al. *Morphological Characteristics Of Clinically Significant Coronary Artery Stenosis In Stable Angina.* British Heart Journal, 1986, (56), 501-508.

10. Hougardy HP. *Recent Progress In Automatic Image Analysis Instruments.* Microscope, 1976, (24), 1.

11. Moura L. *A System For Reconstruction, Handling And Display Of 3D Medical Structures.* PhD Thesis, University of London, Nov 1988, 306 pp.

12. Office of Population Censuses and Surveys. *Mortality Statistics: Cause — England and Wales* 1985. HMSO, 1987.

13. Sissons HA. *Investigations Of Muscle Fibre Size.* In Research in Muscular Dystrophy: Proc. 2nd Symposium, 1963, 89.