Segmentation of Coronary Arteries using Radial Basis Function Neural-Network

Alok Sarwal[1], Atam P. Dhawan[2]

[1] Lockheed Martin Corp., Denver, USA
[2] Department of Bioengineering, University of Toledo, Toledo, USA

Biplane and digital subtraction angiography (DSA) have brought about important advances in the diagnosis and treatment of cardiovascular anomalies by allowing for blood flow measurements, estimation of the regional wall stress and study of myocardium motion. Segmentation of the coronary arteries is a critical first step towards an automated interpretation of angiographs. We present an analysis of neural network methods based on a Radial Basis Function (RBF) and back-propagation (BP) network applied to segmentation of the coronary arterial tree. The results of the neural network based segmentation are compared with segmentation techniques based on a delineation algorithm. Features like vessel diameter and centerline coordinates are extracted for segmented images and compared for the various segmentation methods. The network methods are based on first evaluating the best number of cluster partitions and then automatically obtaining the vectors for training. The pixel gray-level values in a small neighborhood along with information of ridges are utilized to provide the training vectors. The ridge locations indicate high likelihood of continuous points on the artery. A discussion of the learning and generalization characteristics for segmentation, by the networks, is presented for multi-view DSA images and tube phantom simulations.

1. Introduction

Quantification of the three dimensional (3-D) properties of coronary arteries has been of significant interest. The state of these vessels has traditionally been obtained by visual examination from multiple views. Each angiogram provides a two dimensional projection of the three dimensional arterial tree. A clinician then correlates these views through experience and assesses the degree of disease of the arteries. However, this task can be quite difficult and subjective. Computer analysis of angiograms and two dimensional representation of the arteries through automatic segmentation and edge detection, has emerged as an important area of research. Accurate segmentation of the coronary arteries, allows for improved interpretation and subsequent 3-D reconstruction of the arterial tree.

Angiograms are acquired as a sequence of x-ray images during the injection of a contrast agents into the circulatory system to be imaged. Blood vessels must be correctly identified while false positive detection of noise and interference by irrelevant anatomical details must be avoided. The need for automated vessel detection remains critical in computer based systems performing complex tasks, such as quantifying the coronary blood flow, reconstruction of the 3-D vascular structures and tracking vessels in the temporal sequences of angiograms. Thus, the calculation of the diameter of lumen along the vessel is clinically significant. However, the segmentation of the coronary arterial vessels is a difficult problem due to presence of motion artifacts, contrast inhomogeneity, and the unstationary nature of the background. The vessel-diameter typically decreases at the lower levels of the arterial tree. Emphasis has recently shifted to processing of the entire arterial tree, as the entire tree provides the relationship between branches and other valuable arterial information

Some relevant work has been done for obtaining accurate lumen measurements for coronary arterial branches [1,2]. However, such work has provided results for a branch or sub-section of the arterial tree. The coronary vessels exhibit a ridge-like profile for the gray-scale values acquired in the angiogram. A ridge detection algorithm can provide the ridge points present in

the image. A cubic facet model can model the local region in the image, and then use the information to obtain the direction maximizing the second derivative. The ridge extrema can then be taken to be present along the zero-crossing of the first derivative [3]. The magnitude of the second derivative provides information on the sharpness and height of the ridge, while the zero-crossing provides sub-pixel information on the location of the ridge. Such ridge information is relevant for indicating presence of a pixel on an artery at its peak or ridge-point location.

Maximum Likelihood Classification (MLC) [4,5] and delineation based edge-detection [6] have been previously applied to this problem, along with numerous other techniques involving statistical and heuristic methods. Clustering techniques can be applied for the segmentation problem as various regions can be partitioned based on their gray-level values. Fuzzy clustering approaches have been applied for unsupervised clustering and classification without a priori information of the number of clusters [7]. A validity measure is provided for fuzzy c-partitions (FCM) [8] based on compactness and separability of various clusters. It has been shown that fuzziness offers an advantage compared to the hard-clustering approaches by offering some flexibility of membership selection for the data vectors belonging to more than one cluster.

The neural networks provide another classification paradigm, with considerable potential. Neural networks do not require underlying class probability distribution for accurate classification. Rather, the decision boundaries are adapted through an iterative training process. The variation in image quality among various angiograms, and introduction of noise in the course of image acquisition, emphasizes the importance of such an adaptive scheme. The selection of training points can be automatic, and so training or calculation of new weights can be inexpensively performed for each instance of segmentation. This is further supported by the parallelism inherent in feed-forward neural networks. Recently, radial basis function (RBF) networks have provided good results for classification [8] and are gaining wide acceptance. The RBF method has been previously used for interpolation for non-linear multi-dimensional functions, by using a large number of randomly selected RBF center locations. Alternate learning methods, based on orthogonal least squares, have been applied [9]. The RBF network provides a suitable substitute for other two layer architectures. The method used by Broomfield and Lowe [10] provides a suitable basis for the application of RBF to signal processing applications.

2. Methods

We evaluate the performance of neural network classifiers utilizing image gray-scale pixel neighborhood values and pixel contrast information as features for clustering. The contrast value of a pixel is the largest difference in gray-scale value between a pixel and its 8-connected neighbors. The second derivative values at ridge points, is a likely indicator for a point on the artery, and is used to provide training points for the network. A multi-resolution approach is applied and it offers advantage in improved accuracy and speed, for this problem. We present results of applying various networks, with different configurations, for our problem. Specifically, we present a novel method of applying an RBF network, based on initial fuzzy clustering of the data, for coronary arterial segmentation, and compare its performance with a back-propagation network and a delineation method. The effect of network configuration on classification performance is analyzed in this paper. The data used for validation consist of simulated tube phantoms and pig coronary arterial images.

2.1. Selection of Training Vectors Based on Fuzzy Clustering

The feature vector has up to 10 elements and consists of the gray-level value of a pixel and surrounding eight neighbors, combined with the contrast value at the pixel. The initial coarse image segmentation, however, uses only the contrast information in the 3×3 neighborhood for a 9 element vector. The ridge points in the image are evaluated by using a cubic facet model fit in the local neighborhood of the pixel. The bivariate cubic function is fit in a local window surrounding a pixel at row and column location

(r, c) and is given by:

$$f(r,c) = C1+C2r+C3c+C4r^2+C_5rc+C_6c^2 + C_7r^3 + C_8r^2c + C_9rc^2 + C_{10}c^3 \quad \text{(Equation 1)}$$

The coefficients are estimated based on the gray-level values of the underlying image within a window using least square fitting. The function provides information of the first and second derivatives around the central pixel. The pixel is labeled a ridge if the first derivative taken along the extreme direction of the second derivative has a zero crossing sufficiently close to the central pixel [3]. The conditions to be satisfied for a ridge point are: $f'(\rho) = 0$ and $f''(\rho) < 0$ where ρ is small distance from the center of the pixel taken along a direction given by:

$$\text{Sin}\,\theta = \frac{C_2}{\sqrt{C_1 + C_2}} \quad \text{(Equation 2)}$$

$$\text{Cos}\,\theta = \frac{C_3}{\sqrt{C_2 + C_3}} \quad \text{(Equation 3)}$$

This feature detects the presence of a continuous set of points on the arterial branch, for a given window. This is an important attribute for discriminating an artery from noise. A 5×5 window was taken for this work. Cluster analysis is based on partitioning a set of data into similar groups [11]. The fuzzy c-means clustering minimizes the following objective function.

$$J_m = \sum_{i=1}^{c}\sum_{j=1}^{n} \mu_{ij}^m d^2(X_j, V_i) \quad \text{(Equation 4)}$$

$$d^2(X_j, V_i) = (X_j - V_i)^T A(X_j - Vi) \quad \text{(Equation 5)}$$

Where, A is a pXp positive definite matrix and p is the dimension of the vectors X_j $(j = 1, 2 \ldots n)$, V_i the centroid of cluster i, c is the number of clusters, n the number of data vectors, μ_{ij} the fuzzy membership and m is the fuzziness index. The fuzzy c-means algorithm has the following steps [12]: (i) Randomly initialize membership μ_{ij} for vector X_j belonging to cluster i such that $\sum_{i=1}^{c}$, (ii) compute the fuzzy centroid, and; (iii) update the fuzzy membership. The above steps are repeated till the cost function J_m does not change any further. The fuzzy clustering validity set function is utilized for measuring the overall average compactness and separation of the fuzzy c-partitions. The ratio $\Pi_i = \sigma/n_i$ is the compactness of data vectors in class i, where σ is the variance of the data from the cluster center and n_i is the number of vectors in the class. The separation $s = d^2_{min}$ represents the minimum distance between cluster centroids [9].

$d_{min} = \min_{i,j} |Vi - Vj|$ such that a capital s indicates that the clusters are well separated. The best partition of the data vectors represents the smallest measure S, where:

$$S = \frac{\sum_{i=1}^{c}\sum_{j=1}^{n} \mu_{ij}^2 |V_i - X_j|^2}{n \min_{(i,j)} |V_i - V_j|^2 - \frac{1}{n}\sum_{i=1, j\neq i}^{c} |V_i - V_j|^2} \quad \text{(Equation 6)}$$

The denominator has been modified from that in [9] and the deviation from the average set of shortest distances (for all clusters) is considered instead of the smallest distance between any one pair of clusters. The above cost function will provide improved overall solution by considering the global average separation between clusters instead of the worst case cluster pair separation. The above method provides information of the best number of clusters for a given set of feature vectors. These data are obtained from the coarse image obtained by taking average gray-scale value of 4×4 set of pixels. This information is subsequently used to provide an initial segmentation based on the pixel contrast data in a local (3×3) window around pixels. Thus, improved computational efficiency is obtained by using the scaled image for the initial segmentation. The foreground training vectors are taken at the ridge point on the segmented arterial region of the coarse image, and the background-training vector is taken from the immediate background vicinity of this point. The training and test features are then taken to be gray-scale values in the local window, along with the contrast value at that particular pixel in the original image. The selection of appropriate training points is critical for accurate segmentation, and so the following conditions have to be satisfied for foreground point P_i^f to be part of the training point set $(P_1^f, P_2^f \ldots P_n^f)$

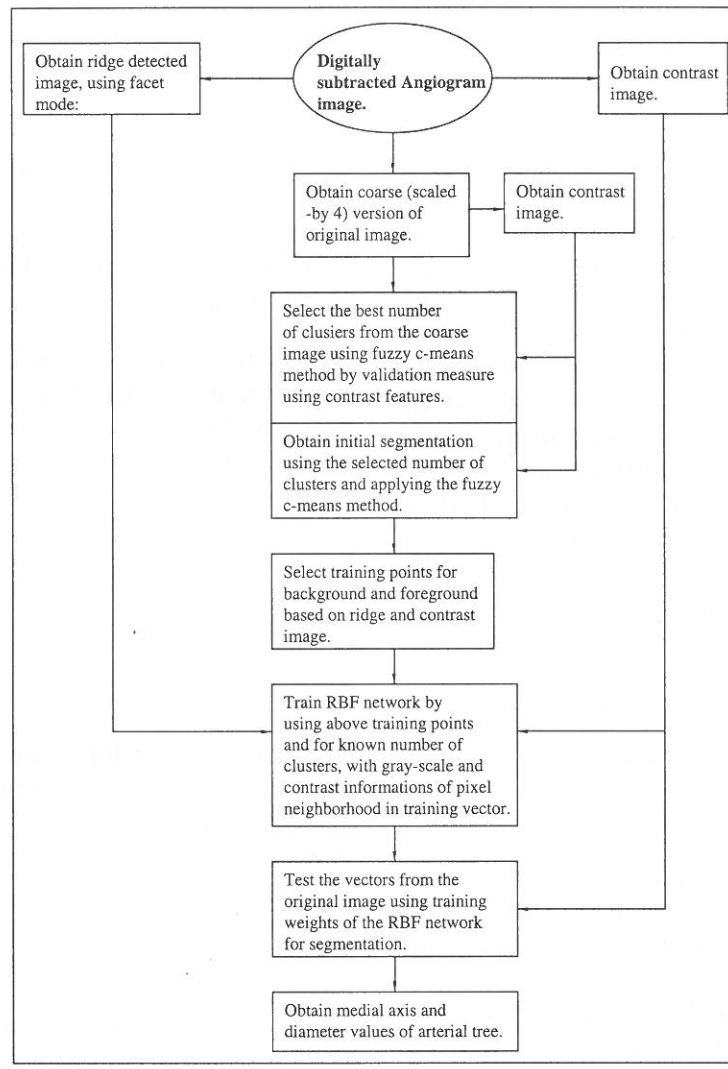

Fig. 1. Arterial Segmentation Algorithm.

- The points should not belong to the background of the segmented coarse image,
- $C_i > \sigma_c$
- $P_i^f \in (R_1, R_2, R_3 \ldots R_n)$

Where C_i is the contrast value of pixel at location i, σ_c is the standard deviation of the contrast values of the entire image and $(R_1, R_2, R_3 \ldots R_n)$ is the set of ridge points detected by the cubic facet model. The background vectors are taken in the immediate vicinity of the foreground point such that: $P_j^{fb} \notin (R_1, R_2, R_3 \ldots R_n)$ and for point j with minimum contrast value C_j s.t. $j \in N(i)$, or neighborhood of point i.

The feature values are all normalized to be unit variance, and as such, the range of values for features with large variability is scaled and increased for features with smaller variability. Thus, the overall relative significance of each feature is preserved. Equal number of training vectors are selected from the foreground and background of the original image so that typically, no more than 1% of the points of the complete image are taken as training points. The segmentation of the entire arterial tree is investigated [14]. Details are shown in Figure 1.

2.2. Radial Basis Function Network Classification

The principle of RBF derives from the theory of functional approximation. Given **N** pairs

$$(\vec{x}_i, y_i)\ (\vec{x} \in \Re^n, y_i \in \Re)$$

We require a function, of the form $f(\vec{x}) = \sum_{i=1}^{K} w_i \phi(|\vec{x} - \vec{c}_i|)$ for K RBF centers, where

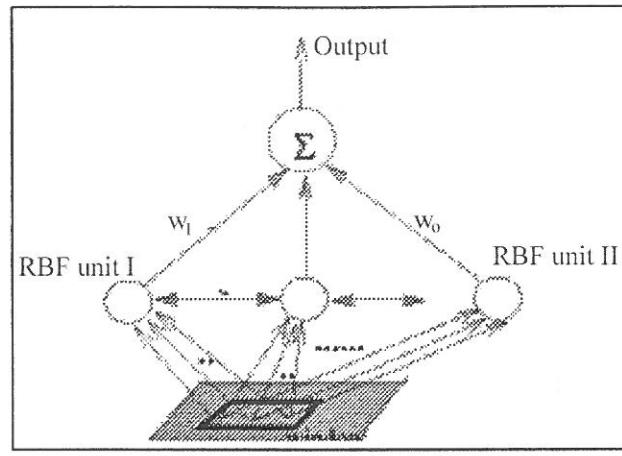

Fig. 2. Radial-Basis-Function neural network.

$w_{,i}\,\phi(|\vec{x}-\vec{c_i}|)$ is the weighted RBF. The RBF is defined as $\phi(x,i)=\exp\left(\frac{|\vec{x}-\vec{c_i}|^2}{2\sigma^2}\right)$.

This represents a function of the Euclidean distance between the components of the function to be approximated, and the location of each of the i centers of the RBF, in a n dimensional vector space. The maximum value of 1.0, is realized for the RBF, when the value $\vec{x}$ is at one of c_i center locations, and it becomes 0 when this separation is large. The function $f(\vec{x})$ should approximate the N pairs and minimize the error function,

$$E(f)=\sum_{i=1}^{N}(y_i-f(\vec{x_i}))^2+\lambda\|Pf\|^2 \qquad \text{(Equation 7)}$$

The first part of the error equation minimizes the total error of the approximation and the second part is a stabilizer that forces the function to be as *smooth* as possible.

The RBF neural network is applied here as a single layer of linearly weighted, radially-symmetric basis functions summed together to provide the output. The variables in this network are the position, variance, and the weights for the RBF node output in the network. Typically, RBF has approximation characteristics which are comparable to multi-layer sigmoid neural networks and which have been shown to be universal approximators.

The j^{th} instance of input vector x with components $(1,2\ldots N)$ is presented to each input. Each RBF node has center c_i, as the location of the RBF, the variance σ^2 or the width and the output of each unit weighted by w_{ij}, as shown in Figure 2.

To account for large variances, the output of the Gaussian nodes is normalized to sum to 1.

$$f(x,i)=\frac{\sum_{i=1}^{K} w_{ij}\phi(x,i)}{\sum_{i=1}^{K}\phi(x,i)} \qquad \text{(Equation 7)}$$

The network weights can be calculated by: $y=F_{(nXp)}W$, where the elements of $F_{(nXp)}$ are the activation functions. Typically, $F_{(nXp)}$ is over-determined and so the least mean square error is computed for the solution. The weights can be calculated by a pseudo-inverse,

$$W=[F^T.F+\alpha I]^{-1}.F^T.y \qquad \text{(Equation 8)}$$

The constant α is a regularization constant, present to avoid singularity for matrix F [12]. The center of the nodes is determined by the fuzzy c-means clustering, as mentioned previously. The variance or width of an RBF node is based on the variance of neighbor clusters. The network consists of a single hidden layer and an output summation unit. The cluster centers are taken to be the position of RBF center [12]. Variance of each RBF is taken by considering the nearest neighbors of the particular cluster and weights calculated as mentioned above. The application of this network is shown in Figure 1. Each RBF is placed at the location of a cluster and ideally we require background and foreground clusters. However, we need additional RBF nodes to improve the conditioning

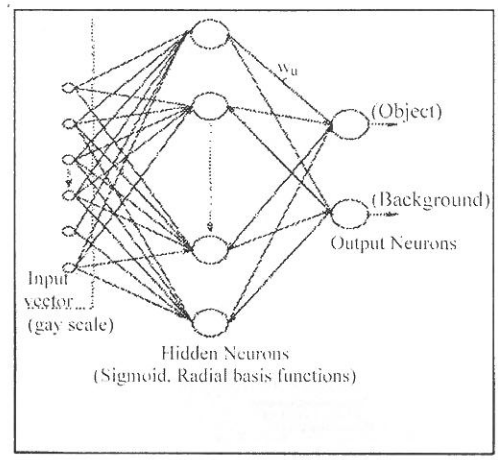

Fig. 3. Back-propagation trained neural network architecture using 15 training vectors, 5 intermediate and 2 output nodes and 0.028 RMS convergence error.

of the solution. Moreover, noise and other artifacts are observed in angiogram data, indicating presence of additional classes.

The training set consists of input vector and the expected output so that discrimination can be made between a test point being an object point and a background point. Training and testing are performed on a local window of size 3×3. The window slides, sequentially, over all the pixels of the image. The gray-scale values of pixels in the window along with the contrast value of the pixel constitute elements of the training and test vectors. The output of the network is modulated by passing it through a sigmoid in the $(-1, +1)$ range and the segmentation threshold is set at 0.0, during the test phase. The architecture of the network is quite important, as the quality of segmentation is directly dependent on the number and size of clusters taken. The effect of number of clusters is shown in Figure 6 and Figure 7.

2.3. Back-propagation Trained Network Segmentation for Comparative Evaluation

It has been shown that back-propagation trained feed-forward neural network (BPNN) with one hidden layer can approximate any continuous function over the compact space Rn [15,16]. The universal approximation property is an existence property. Such a feed-forward neural network classifier (with back-propagation training) for detecting vascular structures in angiograms has been previously implemented by Nekovei et al. [4]. The classifier consisted of a multi-layer feed-forward network window in which the center pixel of a window was classified using gray-scale information within the window. The network was trained by using the back-propagation algorithm with a momentum term. Factors like network topology, rate parameters, training sample set and initial weights contribute to the accuracy of segmentation.

The BPNN is the input data vector similar to the RBF case. The momentum variable is used in the back-propagation algorithm to increase the speed of learning. The network is applied to the entire image, after training for convergence to a suitable error. The background and foreground are each represented by an output, and a pixel is classified as a background or foreground pixel depending on which of the two outputs has the strongest response [4]. Window sizes between 3×3 and 9×9 are used. Two hidden layers form higher dimensional features and can provide improved classification. So, the testing involved up to 2 hidden layers. The learning rate of 0.05 and a momentum term of 0.25 was used, as they provided appropriate learning. The initial weights were randomly selected in the range [-0.5, 0.5]. The number of inputs required are obtained by the window size.

Two outputs are present, for the background and foreground class respectively. The number of hidden layer neurons is selected as a user parameter. The two output nodes are trained to provide the object and the background signals respectively so that the object neuron is set to 1.0 when the input training vector is a window from the object, and the background neuron is set to

1.0 when the vector from background window is input. The node with greater intensity is taken to be the output during testing (segmentation) and this approach resembles *a winner take all* scheme. However, back-propagation, like other gradient descent algorithms, is prone to getting stuck in local minima. Another problem is determining the minimal yet optimal architecture for the best possible classification. This step involves some trial and error.

2.4. Probabilistic Linear Delineation for Comparative Evaluation

The Probabilistic Linear Delineation (PLD) is able to detect a centerline of the coronary arterial tree, without first detecting the associated edge points [6]. The proposed method consists of multiple stages as mentioned below:

First, automatically delineate the artery pixels from the background pixels and create one-dimensional description of the arterial segments which represents the centerline of each segment; next, assign confidence levels to the pixels along these centerline descriptions. Finally, the centerline pixels are aggregated so as to form a fully connected arterial tree skeleton. The algorithm mentioned below:

- Convolve the image with a blurring function.
- Along each row, search for local maxima in the horizontal direction (in the left and right directions). This search is accomplished by performing the following steps at each pixel location:
  - ▷ Starting at a pixel continue to move to the right by single pixel increments provided the new pixel has a smaller grey-level value than the current pixel. Record the minimum found $\min_{\text{right}}$.
  - ▷ Repeat the above step for the opposite direction and record minimum found $\min_{\text{left}}$.
  - ▷ Choose maximum of the two minima found, $\min_{\text{horiz}} = \max(\min_{\text{right}}, \min_{\text{left}})$.
- Repeat the above steps for the vertical direction, such that $\min_{\text{hv}} = \min(\min_{\text{horiz}}, \min_{\text{vert}})$.
- Define the magnitude of the local maximum, β, to be the absolute value of the difference between pixels.
- Grey-level value, α and the background instance, $\min_{\text{hv}}$, $\beta = |\alpha - \min_{\text{hv}}|$.

The above procedure provides an effective method for finding points along the centerline and edges of a given artery. As a side benefit, the minimum value found would be an instance of the background. This is an important feature since it provides the mean value of the background at each point in the image. This implies that we should be able to derive a global variance for the image. The global variance, σ^2 can be defined as follows:

$$\sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(\min_{hv}^{i} - \mu_{bg}^{i}) \qquad \text{(Equation 9)}$$

where N is the number of pixel locations, $\min_{hv}$ is the instance of the background, and μ_{bg} is the local mean estimate of the background taken for all pixel locations. A 1-D median filter applied to the cross section of each artery segment will provide an estimate of the background. The orientation of the arterial cross sections is not known *a priori*. To circumvent this problem, a series of 1-D median filters arranged in four different orientations (0, 45,90, 135 degrees,) are applied at each pixel location. It is then assumed that at least one of these filters should be approximately aligned with the cross section of the segment (orthogonal to the segment) at a particular position and the median value will be lowest along the cross section. The background value for pixels on and off the arteries can be obtained similarly.

Thus, for each pixel location, the local mean of the background in a window of approximately 15 by 15 pixel size, the magnitude of the local maximum at the pixel location, and the global standard deviation of the background can be obtained. This information can be utilized for detecting the edges and centerline of the arterial tree [6].

3. Validation Data

A tube phantom was created by taking a standard 5/16 inch tube and filling it with a mixture of gelatin, milk and blue ink, to approximate a viscous fluid with some level of opacity. This mixture was then set in the twisted flexible tube along with glass capillaries. The capillaries have 0.214 inches width across its widest

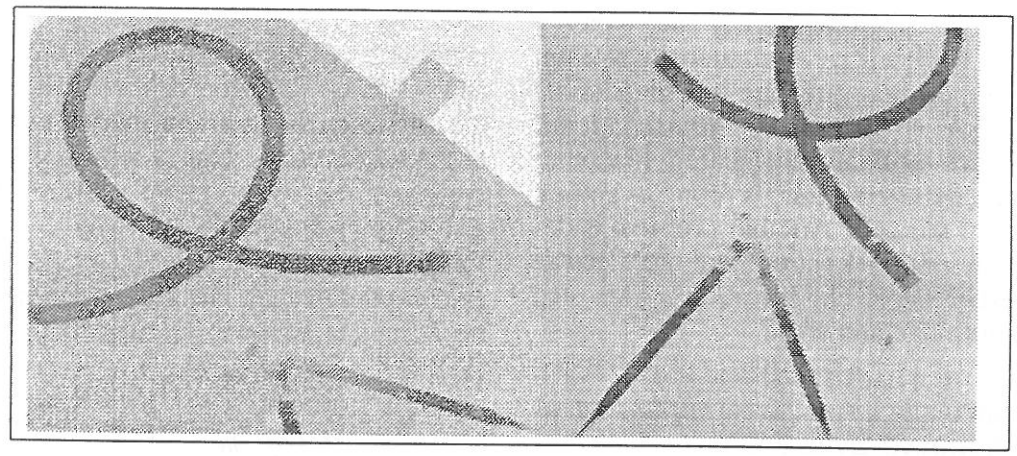

Fig. 4. Tube phantom simulating Arterial Vessels.

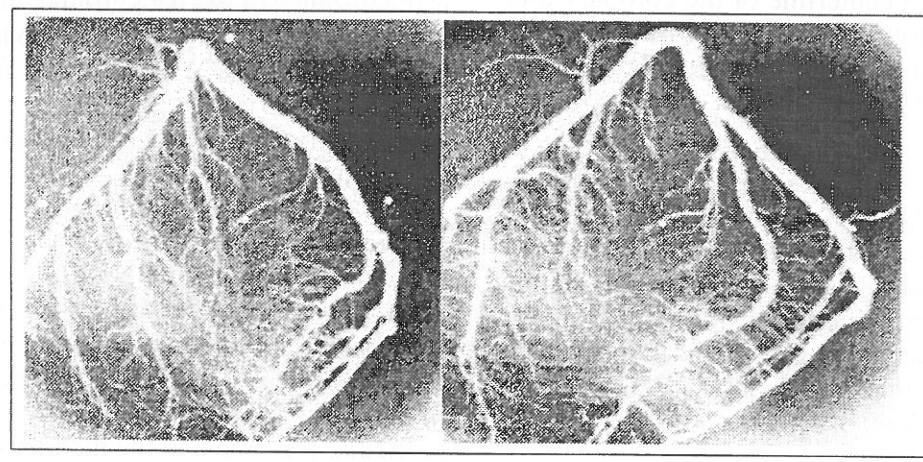

Fig. 5. Pig-cast coronary arterial tree with 5.6 variance white Gaussian Noise.

portion. This phantom was then imaged with a Hitachi CCD camera (DK-7000) and digitized to a 512 by 512 pixel image with 8 bit representation per pixel. The resolution was 0.46 mm. per pixel. The gray-scale images are shown in Figure 4 and Figure 5. Pig arterial-tree cast was used for the angiogram images. Additional noise was added for critically testing the algorithms. A sequence of pig-cast DSA images at three views was used for arterial tree segmentation analysis. The imaging parameters were as follows: polar angles were 66° and 102° and azimuth angles were 30° and 5° for view-1 and view-2 respectively; source to image-intensifier distance was 900 mm.; and pixel resolution was 0.41 mm. per pixel for both views and the image size was 512 by 512 pixels.

4. Algorithm Results

Results are shown for phantom and pig-cast segmentation with the number of clusters selected by the modified fuzzy c-means algorithm. The number of clusters greater than this number will provide comparable results, but increase the computation cost. In case the number of clusters becomes too high, then the noise is in effect learnt as a valid class and segmentation is poor. Up to 16 clusters were adequate for this problem. The results did not show large variability between 12 and 14 clusters. The cluster evaluation curve is shown in Figure 9, from which the value of about 10 clusters is selected to correspond to the point at which the curve starts increasing again. Also shown are the ridge points detected by the facet model ridge detector, superimposed on the angiogram. As mentioned, these ridge points guide the selection of training points. A small value for the

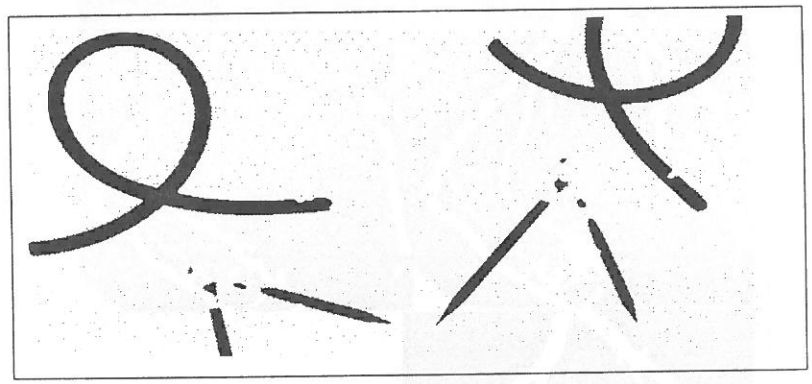

Fig. 6. RBF Segmentation of Phantom Data simulating vascular structures; 5 cluster and 5 clusters with different test cases.

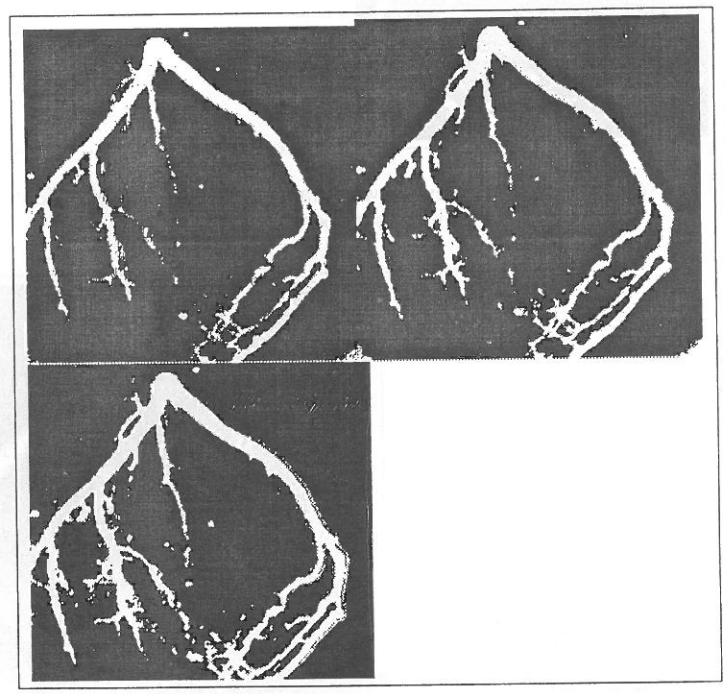

Fig. 7. RBF Segmentation of Angiogram Data of Pig-cast (view 1) Phantom with training and testing on the same image with 10, 12 and 14 clusters respectively.

regularization constant is necessary, so we do not introduce excessive smoothing. Therefore a value of 0.001 was taken for weight calculation for the RBF network. The tube phantom results are quite good as compared to actual or true measurements. White Gaussian noise with 5.6 variance was added for providing a hard data set. The training cases are taken by the automatic point selection method mentioned previously. It should be mentioned that an advantage of this network is that the architecture of the RBFNN depends on the number of clusters only and the vector length is fixed at 10.

While different BP networks (BPNN) topologies showed no significant effect on performance, both the learning process and the classification performance were sensitive to the rate parameters. The best result was obtained with a small learning rate of 0.05 and a momentum of 0.25. A three layer network (with 9-5-2 neurons) was adequate for the problem and showed good generalization to the entire angiogram and other images.

The results are good and can be compared on the basis of the number of pixels classified correctly. It was observed that over generalization was possible in case the training was prolonged. As expected, a biased training data set resulted in a bias towards the dominant class. Training was relatively short in the range of 200 to 600

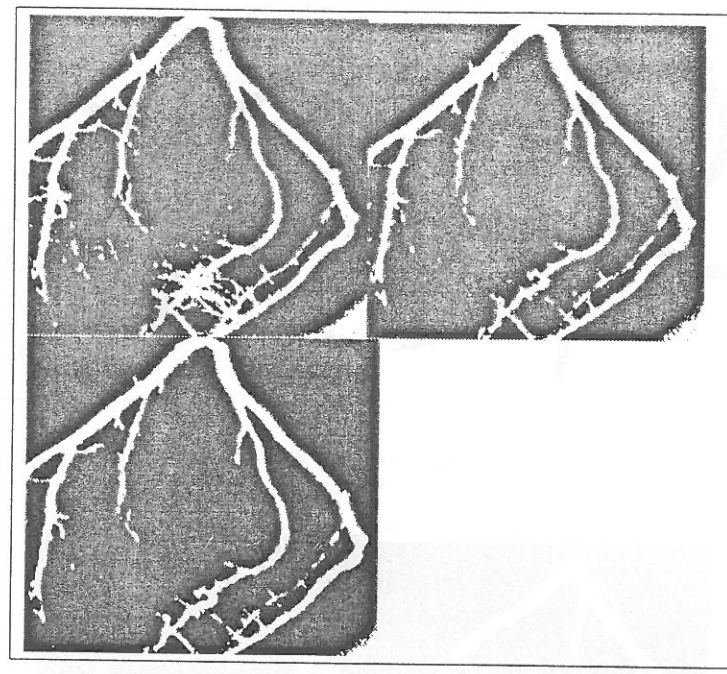

Fig. 8. RBF Segmentation of Pig-cast Phantom (view 2) with training and testing on different images with 10, 12 and 14 clusters respectively.

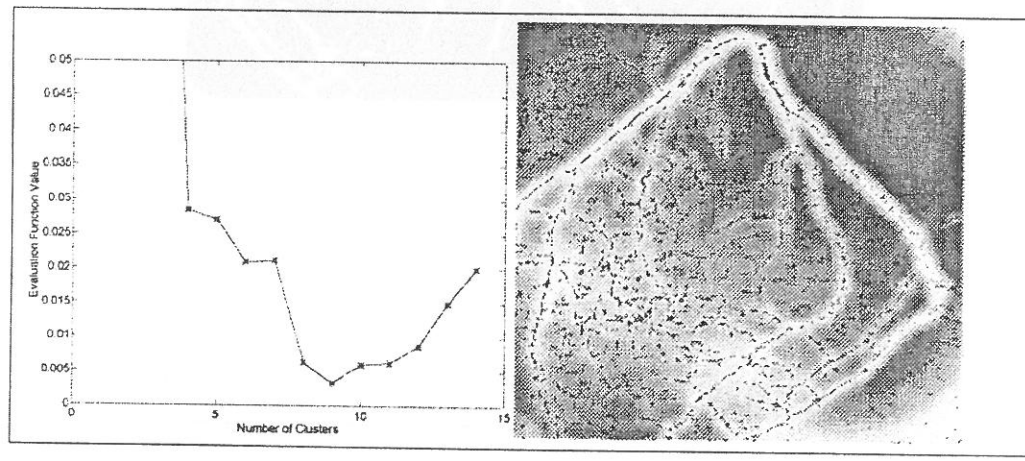

Fig. 9. Ridge Points (by facet model) shown on original image; value of Fuzzy Evaluation Function for different number of clusters, with ridge points and Cluster Evaluation.

epochs. The segmentation did well from one image to the next because a relative classification scheme was used with the maximum and minimum gray-scale values of background and object used to normalize the input to the image. The larger the window, the more details seemed to be extracted, but higher level of noise was also present as in Figure 11.

It should be mentioned that the training set for the BPNN method was selected manually by visual inspection and not by the automated method as for the RBF network based algorithm.

The probabilistic delineation algorithm was also applied on the pig arterial phantoms and the results are shown in Figure 13. The objective of this algorithm was to obtain a width coded centerline of the object that would provide a diameter for comparison with the RBFNN method.

5. Discussion

The larger the window, the more details seemed to be extracted, but greater noise is present for the neural networks methods. A comparison of the phantom image segmented diameter with the actual measured values, indicates that the

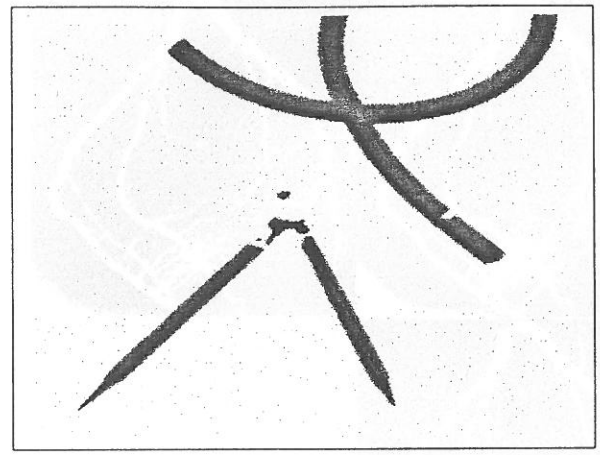

Fig. 10. BPNN Segmentation of Phantom data simulating vascular structures, using 15 training vectors, 5 intermediate and 2 output nodes and 0.028 RMS convergence error, using a 3 by 3 window.

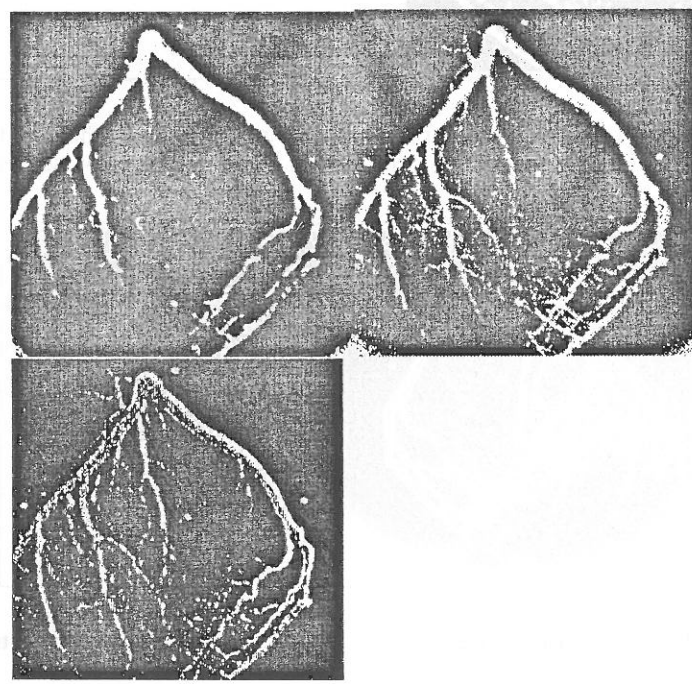

Fig. 11. 11 BPNN Segmentation of Angiogram Data of Pig-cast (view 1) Phantom for 75 training cases, with training and testing on the same image, 5 intermediate and 2 output nodes, 0.028 RMS error of convergence, using a 3 by 3, 5 by 5 and 7 by 7 window respectively.

results for the RBFNN method are quite encouraging. A comparison of the error of segmentation by RBFNN and a BPNN is shown in Figure 14. A fixed window size of 3×3 was taken for both networks and 5 partitions were the best selection from the modified FCM algorithm, for the RBFNN. The RMS error for the RBFNN was 0.76, and 0.97 for BPNN. The error was computed for various points lying along the rows of comparable image (Tube Phantom 1). The diameter was computed along direction perpendicular to the medial axis for the selected point. The results indicate that the RBF network performed more accurate segmentation for the window size considered here, as shown in Figure 14.

A comparison between the diameter from the RBFNN segmentation method and PLD was performed. The diameter is calculated of the arterial tree shown in Figure 13, at periodic row intervals. The RBFNN method appears to provide better continuity and less variability among adjacent diameter measurements along branches, compared to the PLD method. The

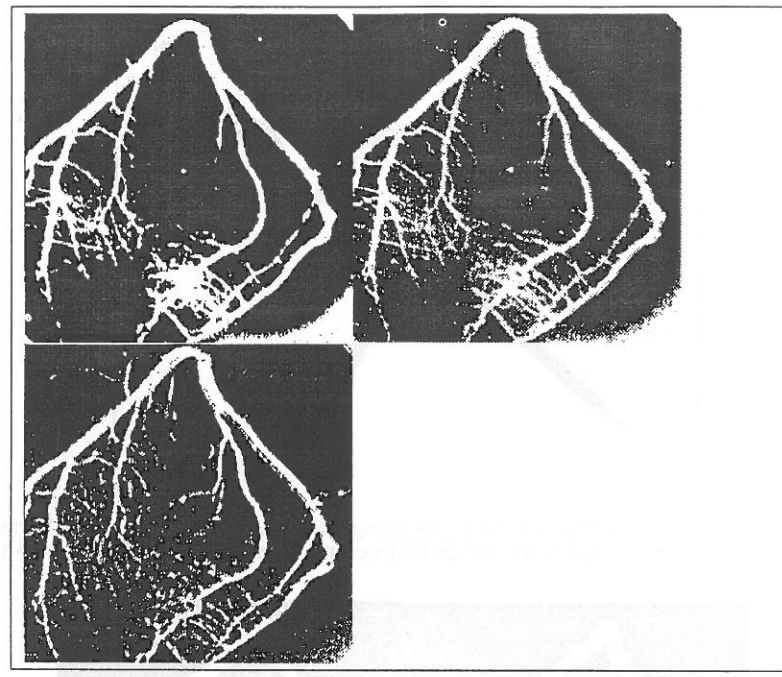

Fig. 12. 12 BPNN Segmentation of View 2 Angiogram Data of Pig-cast (view 2) Phantom for 75 training cases, with training and testing on different images, with 3 by 3, 5 by 5 and 7 by 7 windows respectively.

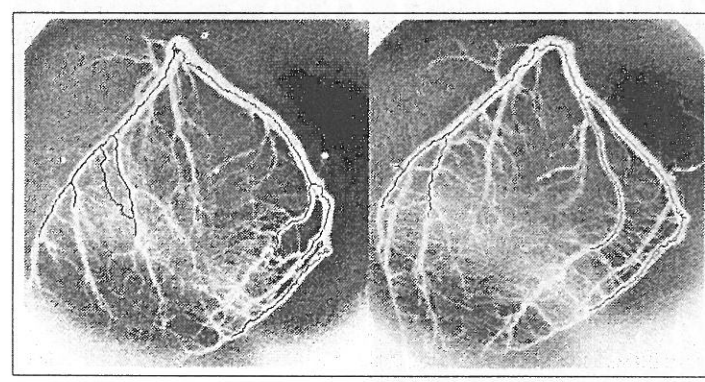

Fig. 13. Pig-cast Coronary Arterial tree (View 1, View 2) width and centerline extraction by PLD.

PLD method can cause some loss of accuracy due to the smoothing (averaging) of gray-level values in the local window being considered. Pearson's product-moment correlation coefficient [15] P_c, was calculated to be 0.872 for the two sets of diameter measurements shown in Figure 15. Assuming x_i and y_i are the two diameter sets, P_c is:

$$P_c = \frac{\sum_i (x_i - \mu_x)(y_i - \mu_y)}{\sqrt{\sum_i (x_i - \mu_x)^2} \sqrt{\sum_i (y_i - \mu_y)^2}} \quad \text{(Equation 10)}$$

6. Conclusion

This paper has presented a RBF neural network based segmentation algorithm for coronary arteries. The automated selection of training vectors from one of a set of images is adequate for generalized segmentation. However, even though we have demonstrated generalization, training and final segmentation for the same image are possible by this algorithm, as the computation cost is not too high. The complete algorithm took 18 seconds to process on a Ultra-1 workstation for one pigcast angiogram (view 1 or view 2.) Fuzzy clustering by the modified evaluation function, combined with a

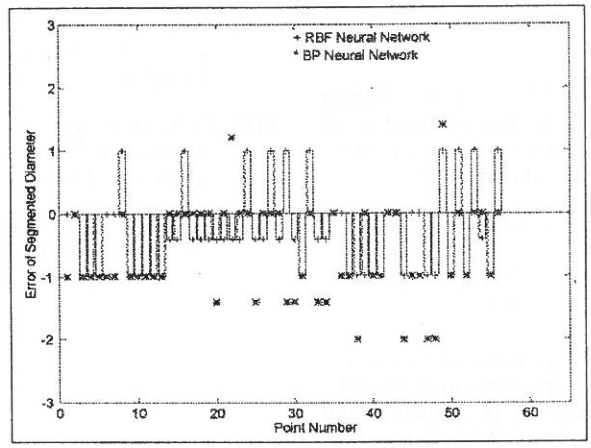

Fig. 14. Diameter error comparison of Radial Basis Function and Back-propagation trained neural networks for a 3×3 window. Diameter computed by using Euclidean pixel distances for tube phantom by training on the first and testing on the second.

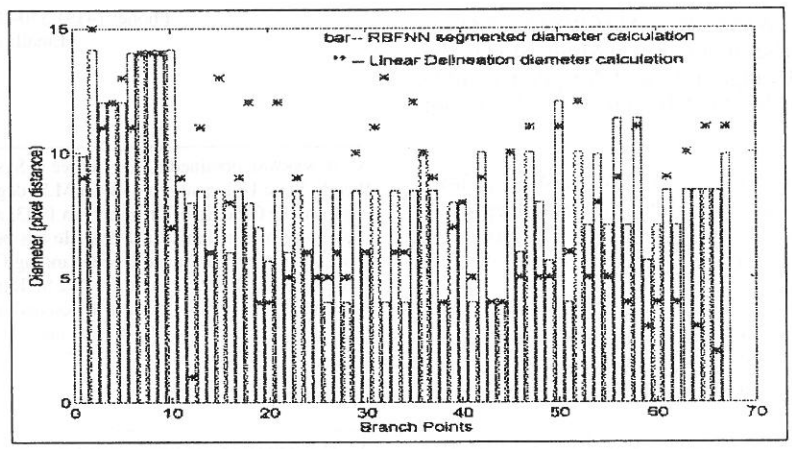

Fig. 15. Diameter comparison of Radial Basis Function neural networks and Linear Delineation. Diameter is computed by using Euclidean pixel distances for coronary arteries by training on the first view and testing on the second.

multi-resolution approach for image feature selection offers an algorithm capable of giving accurate results. It should be noted, that it also offers parallel processing characteristics similar to other neural networks and that the network size is fixed by the number of fuzzy cluster partitions.

Back-propagation NN provided good results using a 3 or 4 layer architecture with 2 output neurons. The BPNN training is iterative and so computation time is proportional to the number of epochs necessary. Appropriate selection of the number of layers and the neurons per layer is typically experience based. The RBFNN performs a pseudo-inverse and so the time for training is a variable of the number of clusters.

The accuracy of segmentation is shown in the previous section and the RBFNN results are quite promising, as compared to other approaches. Moreover, the RBFNN seems to offer more robustness to architecture changes and a 3×3 window was quite adequate for this problem using the automated algorithm described in Figure 1. The RBFNN results were compared to those from PLD algorithm, applied to *tube like* structures approximating coronary arteries. However, PLD has limitations due to its strong dependence on manual selection of parameters like median filter size, size of mask for smoothing regions, distance between the branches to connect etc. intuitively based on the noise level and contrast of angiogram.

References

[1] SONKA, M.D. WINNIFORD, S.M. COLLINS, Robust Simultaneous Detection of Coronary Borders in Complex Images, IEEE Trans. on Medical Imaging, vol 14(1), pp. 151–161, March 1995.

[2] FIGUEIREDO, J.M.N. LEITAO, A Nonsmoothing Approach to the Estimation of Vessel Contours in Angiograms, IEEE Trans. on Medical Imaging, vol 14(1), pp. 162–172, March 1995.

[3] HARALICK, Cubic Facet Model Edge Detector and Ridge–Valley Detector: Implementation Details, Pattern recognition in Practice II, Gelsema and Kanal, Editors, Elsevier Science Publishers.

[4] NEKOVEI, Y. SUN, Back-Propagation Network and its Configuration for Blood Vessel Detection in Angiograms, IEEE Trans. on Neural Networks, vol 6(1), pp. 64–72, Jan. 1995.

[5] OZKAN, B.M. DAWANT, R.J. MACIUNAS, Neural-Network-Based Segmentation of Multi-Modal Medical Images: A Comparative and Prospective Study, IEEE Trans. on Medical Imaging, vol 12(3), pp. Sept. 1993.

[6] DUFRESNE., A. SARWAL, A.P. DHAWAN, A Gray-level Thinning Method for Delineation and Representation of Arteries, in Journal of Computerized Medical Imaging and Graphics, vol. 18 (5), pp. 343–355, 1994.

[7] GATH, A.B. GEVA, Unsupervised Optimal Fuzzy Clustering, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 11(7), July 1989.

[8] BEZDEK, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, New York, 1981.

[9] CHEN, C.F.N. COWAN, P.M. GRANT, Orthogonal least squares learning for radial basis function networks, IEEE Trans. On Neural Networks, 2(2): pp. 302–309, 1991.

[10] BROOMHEAD, D. LOWE, Multivariate functional interpolation and adaptive networks, Complex Systems, 2: pp. 231–355, 1988.

[11] DUDA, P.E. HART, Pattern Classification and Scene Analysis, John Wiley Sons, Inc.

[12] WHEELER, A.P. DHAWAN, C.M. MEYER, Space Shuttle Main Engine Sensor Modeling Using Radial Basis Function Neural Networks, AIAA Journal of Spacecraft and Rockets, vol. 31 (6), pp. 1054–1060.

[13] XUANLI, G. BENI, A Validity Measure for Fuzzy Clustering, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 133(8), pp. August 1991.

[14] SARWAL, A.P. DHAWAN, Y. CHITRE, 3–D Reconstruction of Coronary Arteries using Estimation techniques, SPIE Conf. of Medical Imaging, San–Diego, March 1995.

[15] PARK, I.W. SANDBERG, Universal Approximation Using Radial-Basis-Function Networks, Neural Computation, vol. 3 (2), 1991, pp. 246–257.

[16] CYBENKO, Approximation by Superposition of a Sigmoidal Function, Mathematics of Controls, Signals and Systems, vol. 2, 1989, pp. 303–314.

[17] PRESS, S.A. TEUKOLSKY, W.T. VETTERLING, B.P. FLANNERY, Numerical Recipes in C, 1990, Cambridge University Press.

Received: July, 1997
Revised: April, 1998
Accepted: April, 1998

Contact address:
Dr. Atam P. Dhawan
Dept. Of Bioengineering
5020 Nitschke Hall
University of Toledo
2801 W Bancroft Street
Toledo, OH 43606.
Phone: (419) 530–8267, Fax: (419) 530–8076
Email: adhawan@eng.utoledo.edu
USA

ALOK SARWAL obtained a M.S. degree in Systems Engineering from the Wright State University in 1984, a M.S. degree in Electrical Engineering from the Ohio State University in 1993, and a Ph.D. degree from the University of Cincinnati in 1996. He has been employed at Lockheed Martin Corp., in the Image Understanding Group at Denver, Colorado, since February 1996. His interests are SAR image understanding, object recognition, 3-D object reconstruction and high-performance computation. He is a member of IEEE computer society.

ATAM P. DHAWAN obtained his B.Eng. and M.Eng. degrees in Electrical Engineering from the University of Roorkee, Roorkee, India. He was a Canadian Commonwealth Fellow at the University of Manitoba where he completed his Ph.D. in Electrical Engineering with specialization in medical imaging and image analysis. In 1984, he won the first prize and the Martin Epstein Award at the Symposium of Computer Application in Medical Care Paper Competition at the Eighth SCAMC Annual Congress in Washington, D.C., for his work on developing a three-dimensional (3D) imaging technique to detect early skin-cancer called melanoma. From 1985 to 1988, he was an Assistant Professor at the Department of Electrical Engineering at the University of Houston. Later, in 1988, he joined the University of Cincinnati as an Assistant Professor where he became Professor of Electrical and Computer Engineering and Computer Science, and Radiology (joint appointment). From 1990-96, he was the Director of the Center for Intelligent Vision and Information System. From 1996-98, he was Professor of Electrical Engineering at the University of Texas at Arlington and Adjunct Professor of Radiology at the UT Southwestern Medical Center. Currently, he is Professor of Bioengineering at the University of Toledo and Adjunct Professor of Radiology at the Medical College of Ohio. Dr. Dhawan is a recipient of Martin Epstein Award (1984), NIH-FIRST Award (1988), Sigma-Xi Young Investigator Award (1992), UC Faculty Achievement Award (1994) and the prestigious IEEE Engineering in Medicine and Biology Early Career Achievement Award (1995). He is an Associate Editor of IEEE Transactions on Biomedical Engineering and Assistant Editor of IEEE Transactions on Rehabilitation Engineering.

His current research interests are medical imaging, multimodality brain mapping, intelligent image analysis, image reconstruction, wavelets, genetic algorithms, neural networks, adaptive learning and pattern recognition.