Proceedings of the 5th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, Madrid, Spain, February 15-17, 2006 (pp280-285)

Novel Intelligent Edge Detector for Sonographical Images

Ali Rafiee
Islamic Azad University – Kazeroun Branch ; Kazeroun Iran ,

Abstract--Most image processing, such as image registration, image segmentation, region separation, object description, and recognition, use edge detection as a preprocessing stage. Real ultrasound images, such as sonography images, can be corrupted with speckle noise. The real problem is how to extract the edges and simultaneously preserve image details. In this paper a new genetic-neuro-fuzzy system is suggested for edge detector in ultrasound images. The competitive neural network (NN) is used for this system. Data processing will be done by a winner-take-all competition process is applied to subnetworks in NN and neurons in each subnetwork. The fuzzy transformer system is used to convert the neighborhood window of input pixels to three decision fuzzy parameters. The on-line genetic algorithm (OGA) is used to optimize and regulate the system parameters. A binary pattern of neighborhood window is obtained based on winner subnetwork and neuron. After detecting the first set of edge pixels, next structural algorithm will be applied according to the location of edge pixels to eliminate some of the noisy edges and add some weak real edge pixels. System performance is compared with the standard methods such as Sobel and zero-crossing edge detector. Results show that the genetic-neuro-fuzzy edge detector is a powerful edge detector, whose performance is better than standard edge detectors.

***Index Terms*:** Genetic-neuro-fuzzy, Competitive neural network, Edge detection, Sonography images,

I. Introduction

THE noninvasive nature, low cost, portability, and real-time image formation make ultrasound (US) image an essential tool for medical diagnosis. Most ultrasound image processing applications, such as image registration, image segmentation, region separation, object description, and recognition, use edge detection as a preprocessing stage for feature extraction. This imaging modality, when used noninvasive, allows high acquisition rates and provides images in real-time, but the images is corrupted by a high level of speckle noise [6]. The problem of isolating intensity changes in US imagery is exacerbated by the presence of speckle, which appears as a jumble of randomly placed bright and dark spots. This noise makes it difficult to accurately identify edges, since in some regions the noise produces artificial edges, while in other regions there are no echoes present and the edges seem ambiguous. In such low-quality images (which are very common in US imaging), generic algorithms do not identify the border accurately. Several algorithms have been reported, which could help identify edges in US images [2][8][9].

The golden standard algorithm for detection of edges in images was reported several years ago by Canny [7]. In that case, the edge is defined as a step function embedded in white noise. But in US image data, the noise is speckle, which has a high degree of correlation with the data. The edge cannot be described by a step function, and the difference in the average gray levels of the various regions is high.

There have also been many studies of edge detection with learning models that mimic one style or the other. This class includes, for example, computational neural networks[1][3] [11][13][19][22][24], fuzzy reasoning systems [4][5][10][12][17][20][21][23] and neuro- fuzzy systems[14].

The more recent past techniques concerned with NNs have been inspired by features of NNs such as fault tolerance, computational simplicity, capability to learn from examples for determining correct threshold and ability to process in a highly parallel fashion that yield a rich variety of edge images.

Ho [10], Russo [17] and Tizhoosh [21] suggested several models of fuzzy edge detector. Fuzzy edge detectors are flexible and robust methods, while heuristic membership functions, simple fuzzy rules and many interference methods can be used in these systems.

Lu and Wang [14] used a fuzzy neural network for edge detector that includes two stages: adaptive fuzzification and detection. The main idea in their system is division input patterns to 8 groups and classifies these patterns to edge or non-edge classes. They claimed that this edge detector could be acts very well in additive noises.

Considering the drawbacks of the edge detector systems mentioned, we have constructed our edge detection method for ultrasound images based on an intelligent system. This system is composed of neural network that is the main structure of edge detector system, fuzzy system that is used for solving the ambiguity problems in edge definition and genetic algorithm that is an optimum algorithm in NN learning and setting the system structure. There are two problems that are solved by using the intelligent system: thresholding problem in edge variation range and presence of noisy edges in images. Neuro-fuzzy network uses the learning ability of the neural networks, for which the form of information in this system is fuzzy. The on-line genetic algorithm (OGA) is used to optimize and regulate the system parameters [15][16][18].
This paper organized as follows. Section II contains the genetic-neuro-fuzzy edge detection system; Section III describes the final results and their comparison to the other filters. Finally, Section IV contains our conclusions

II. Genetic-neuro-fuzzy system

Our edge detector system is shown in Fig. 1. This edge detector is based on a 3x3 neighborhood window, which is the input for edge detector system. The following three parameters can be obtained from the neighborhood window by using fuzzy converter system; $\bar{X}$: Fuzzy means of gray levels in neighborhood window.

X_{min}: fuzzy means of gray levels lower than $\bar{X}$ in neighborhood window.

X_{max}: fuzzy means of gray levels more than $\bar{X}$ in neighborhood window.

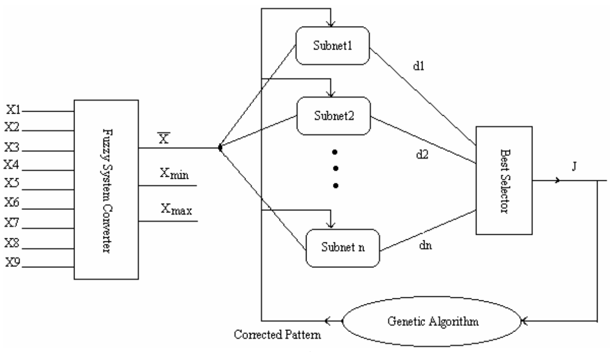

Fig. 1: Genetic-neuro-fuzzy edge detector.

Fuzzy system is used to weight each pixel, based on its gray level to obtain the above parameters. Fuzzy set has two members and its membership function is trapezoidal. Membership function for each window is defined adaptively based on its mean values. Maximum operator is used for fuzzy interference. $\bar{X}$, X_{min} and X_{max} are calculated below:

$$X_{\min}=\frac{\sum_{i=1}^{9} X_i\, I(X_i)}{\sum_{i=1}^{9} I(X_i)} \qquad X_{\max}=\frac{\sum_{i=1}^{9} X_i\, \bar{I}(X_i)}{\sum_{i=1}^{9} \bar{I}(X_i)} \tag{1}$$

$$I(X)=\begin{cases}0 & X \phi X_{avg}\\ 1 & X \le X_{avg}\end{cases}, \quad \bar{I}(X)=\begin{cases}0 & X \pi X_{avg}\\ 1 & X \ge X_{avg}\end{cases} \tag{2}$$

$$\bar{X}=\frac{X_{min}+X_{max}}{2} \qquad X_{avg}=\frac{\sum_{i=1}^{9} X_i}{9} \tag{3}$$

$\bar{X}$ Parameter is used to select the winner subnetwork and X_{min} and X_{max} are for selecting winner neuron in this subnetwork.

A. Context pattern in edge detector

This system architecture arouse from our observation that in edge detection, it would be more effective to adapt multiple sets of thresholding decision parameters corresponding to different local contexts. The idea behind this scheme is that each subnetwork is associated with an edge template corresponded to a different context and each neuron in the subnetwork encodes variation of edge prototypes under corresponding background elimination. A pattern of this categorization method is shown in Fig. 2. P parameters show the central point of each context (corresponding to each subnetwork) in this figure. These parameters are correctable and will be optimized in NN training steps based on $\bar{X}$ parameters in each window. In training steps $\bar{X}$ will be calculated for each window and the subnetwork with a closer P value to $\bar{X}$ and in the range [X_{min} X_{max}] will be selected and the subnetworks P value is corrected.

B. Neural network

The competitive neural network is used for this system. Data processing will be done by a winner-take-all competition process is applied to subnetworks in NN and neurons in each subnetwork. This process is done by use of input neighborhood window. Each neuron is corresponding to the threshold value for edge detection. Two weights are defined for each neuron, that threshold and gray level change ranges are shown by the different between these weights.

Each subnetwork consist two neurons in order that one of the neurons associate to the local prototype for weakly edges and the other

Proceedings of the 5th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, Madrid, Spain, February 15-17, 2006 (pp280-285)

associate with prototype for strongly edges. Therefore this is a form of hystersis thresholding that is used by more edge detectors (Fig. 3). For each input neighborhood window, the $\bar{x}$, X_{min} and X_{max} parameters are obtained and winner subnetwork is selected. Distance between parameters is adopted to subnetwork selection as shown below:

$$d_j = \min_{i=1}^{n} \left(\left| \bar{X} - P_i \right| \right) \qquad j = 1,.., \ n \quad (4)$$

d_j is minimum distance and j is winner subnetwork. Each subnetwork consists two neurons and each neuron includes two weights that is defined as follows:

$$W_j = \lfloor W_{j1} \ \ W_{j2} \rfloor \qquad j = 1,2 \qquad (5)$$

The winner-take-all nature of the competition process also favors that only the local winner within each subnetwork is allowed to update its weight vector. The local neuron output is evaluated in term of the Euclidean distance between the edge prototype and the current edge, which is defined as follows:

$$n_k = \min_{i=1}^{2} \left\| X - W_j \right\|^2, X = [X_{min} \ \ X_{max}], \ k = 1,2 \ (6)$$

Xn_k is minimum Euclidean distance and k is winner neuron number.

C. Training stage

This NN has 6 subnetworks, which this number is selected based on gray level gradient corresponding to edges (about 20) and variation range of edge gray level (about 40). These values can be sensed visually. Network parameters are first selected randomly then updated in training stages according to the input images. As there are 4 weights and one P parameter for each subnetwork, 24 parameters exist for NN. Online genetic algorithm (OGA) is used for training network.

OGA algorithm is a single-member algorithm. Each generation has a queen (member) and just mutation operator produces next generation. Genetic string includes subnetwork P parameters and neuron weights. P parameters is an 8-bits number (for 256 gray level) and neuron weights are each an 8-bits number (4 weights in each subnetwork); totally a genetic string is a 40-bits as shown in Fig. 4. There are 6 genetic strings in NN, which are optimized indepently by winner-take-all process. The input image is scanned from top to down and left to right; neighborhood window is performed for each pixel. Neighborhood window is fed into the edge detector system and the NN weight and P parameters are updated. In this window $\bar{x}$, X_{min} and X_{max} are calculated and winner subnetwork and also the winner neuron in this subnetwork are selected to update their parameters.

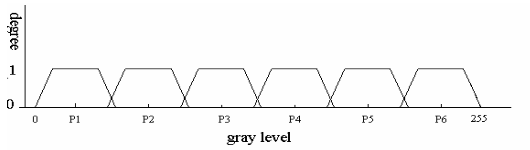

Fig. 2: An example of background elimination categorization

Mutation operator is applied by selecting random bits in genetic string and complementing them. The new subnetwork and neuron parameters create by this new string (son). If the fitness function of this subnetwork is better than before, the new generation is considered as the queen otherwise it is rejected and the previous queen is used to generate new genetic string.

The fitness function used for P parameters is the inverse of the distance between $\bar{x}$ and this parameters and the same function used for weights is the inverse of Euclidean distance between weight vector and $X=[X_{min} , X_{max}]$.

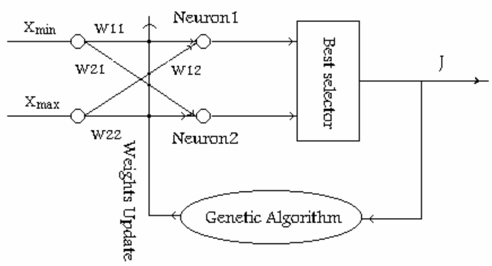

Fig. 3: subnetwork structure.

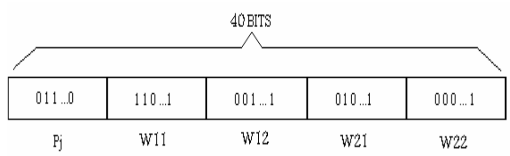

Fig. 4: Genetic binary string for weights and P parameter.

According to the limited input variations, the network will be optimized by at last 20 generations. After training steps the input image is entered to the edge detector system once again and a binary pattern is obtained for each neighborhood window. In this patterns existence or inexistency of edges can be seen visually. The edge patterns are fed into the edge set to use in recognition stages. Binary patterns are created by dividing the neuron gray level range into the two equal sections so that every pixel in the upper

Proceedings of the 5th WSEAS Int. Conf. on Artificial Intelligence, Knowledge Engineering and Data Bases, Madrid, Spain, February 15-17, 2006 (pp280-285)

area is set to 1 and the others set to 0. The following relation shows this transformation:

$$b_i = \begin{cases} 0 & |X_i - W_{k1}| < |X_i - W_{k2}| \\ 1 & \text{otherwise} \end{cases} \quad (7)$$

X_i is the pixel in neighborhood window and b_i is the binary pixel. 7 others binary patterns can be created and entered to edges set to eliminated the rotationally effects in edge detection, by 45° clockwise rotation as shown an example in Fig. 5.

0	0	0
1	0	0
1	1	1

0	0	0
0	0	1
1	1	1

0	0	1
0	0	1
0	1	1

0	1	1
0	0	1
0	0	1

1	1	1
0	0	1
0	0	0

1	1	1
1	0	0
0	0	0

1	1	0
1	0	0
1	0	0

1	0	0
1	0	0
1	1	0

Fig. 5: An example for main binary and 45 degree rotated patterns

D. Recognition stage

In system test process, new images are applied to the system, for each pixel a neighborhood window is entered to the fuzzy transformer. In this window $\bar{X}$, X_{min} and X_{max} values are calculated and by Equations (4) and (5) winner subnetwork and neurons are selected. If the following condition satisfies, this pixel is suggested as the edge:

a) $X_{max} - X_{min} \geq \min(w_{11} - w_{12}, w_{21} - w_{22})$
(Distance more than low threshold).

b) The input binary pattern exists in edges set.

Binary image can be created by on-edge pixels setting to one and off-edge pixels setting to zero in which context and edges are completely obvious.

E. Post processing

After detecting the first set of edge pixels, next process will take effect to eliminate some of the noisy edges and add some weak real edge pixels. So a structural algorithm will be applied according to the location of edge pixels. This algorithm that is called thinning, is performed by the binary post-processing follows a few simple rules, which remove spurious or unwanted edge points and add in edge points where they should be reported but have not been. They fall into three categories; those removing spurious or unwanted edge points, those adding new edge points and those shifting edge points to new positions.

III. Experimental results

Input images in this system are ultrasound images that are corrupted with speckle noise. This noise has a high correlation with the main images and is a multiplicative noise. Genetic-neuro-fuzzy edge detector is used in this paper. Fuzzy system acts as an input converter that converts neighborhood window to a 3 fuzzy parameters $\bar{X}$, X_{min} and X_{max}. These parameters select the subnetwork and the neurons in this subnetwork corresponding with the input neighborhood window.

The NN used here is a competitive network that includes 6 subnetworks and each subnetwork represents a gray level range. This network structure covers the edge variation in different contexts. There are 2 neurons In each subnetwork that one of them corresponded to the low threshold and the others one corresponded to high threshold. Therefore a hystersis model of threshold is created in this edge detector.

A binary pattern of neighborhood window is obtained based on winner subnetwork and neuron. In case this binary pattern exists in edges reference set and the difference between X_{max} and X_{min} is more than low threshold, the central pixel will be assigned to edge pixels. The structural postprocessing will be done to eliminate noisy edges and add real weak edges.

Ultrasound images have 256 gray level and best criteria for edge detection performance is visual observation. System performance is compared with the standard methods such as Sobel, zero-crossing and Canny edge detector. The genetic-neuro-fuzzy edge detector was applied to sonography images. A typical Bowel image that used here is depicted in Fig. 6(a). The genetic-neuro-fuzzy edge detector output is displayed in fig. 6(b). The results are compared with the Sobel operator and the zero-crossing edge detector that is shown in Fig. 6(c) and. 6(d). The Canny output also is given and depicted in Fig. 6(e). As shown in Fig. 6 the genetic-neuro-fuzzy edge detector is a powerful edge detector, which its performance is better than standard edge detectors. In Fig. 7(a), a sample renal image and in Fig. 7(b) the noisy version that is corrupted by speckle noise (0.1 variance) is depicted. The genetic-neuro-fuzzy output is depicted in Fig. 7(c). The Sobel, zero-crossing and Canny output are shown in Fig. 7(d), 7(e) and 7(f).

IV. Conclusion

We have suggested a new genetic-neuro-fuzzy system for edge detector in ultrasound images. The competitive NN is used for this system. The fuzzy converter system is used to convert the input pixels to decision fuzzy parameters. The OGA algorithm is used to learn the system parameters. A set of binary patterns of edges is used for edge detection. After detecting the first set of edge pixels, the postprocessing algorithm will be applied to eliminate the noisy edges and add real weak edges. System performance is compared with the standard methods such as Sobel, zero-crossing and Canny edge detector. A malignant tumor has been illustrated in figure 6. But how much is the extension and invasion of this tumor? By comparing the revealed borders and also co-observation of fig. a and b it can be shown that right border of tumor extends up to the edge of picture in figure b but it seems that right border of tumor has several centimeter distance with the edge of picture a. The real borders of tumor could not be shown in none of figures c, d and e. Sonographic shadow of a kidney is shown in figure 7. Is there any cyst inside this kidney? It's difficult to confirm or rule out this issue by using figures a and b. It would be a higher possibility of renal cyst by virtual edges in fig c and e although it's not true. The architecture is completely disfigured in figure d. Fig. f is clearly revealing how an apparent renal cyst on a particular angle could be cleared as normal renal tissue by edge clarification. Results show that the performance of genetic-neuro-fuzzy edge detector is better than standard edge detectors.

References

[1] I.N Aizenberg, N.N Aizenberg, J. Vandewalle," Precise edge detection: representation by boolean functions, implementation on the CNN", 1998 Fifth Inter. Workshop on Cellular NNs and their Applications, pp. 301-306, London, 14-17, April 1998.

[2] A. A. Amini, T. E.Weymouth, and R. C. Jain, "Using dynamic programming for solving variational problems in vision," IEEE Trans. Pattern Anal. Machine Intell., vol. 12, pp. 855–867, Sept. 1990.

[3] J. C. Bezdek and D. Kerr, "Training edge detecting neural networks with model-based examples," in Proc. 3rd IEEE Int. Conf. Fuzzy Syst., pp. 894-901, Piscataway, NJ, 1994.

[4] J. C. Bezdek and M. Shirvaikar, "Edge detection using the fuzzy control paradigm," in Proc. 2nd Eur. Congress Intell. Tech. Soft Computing, vol. 1, pp.1-12, Aachen, Germany, 1994.

[5] J. C. Bezdek, R. Chandrasekhar, "A geometric approach to edge detection ", IEEE Transactions on Fuzzy Systems, vol. 6, no. 1, pp. 52-75, FEBRUARY 1998.

[6] A.C. Bovik, "Detection of object boundaries in synthetic aperture radar imagery using a human visual model ",ICASSP'86, pp. 2055-2058, Tokyo Japan, 1986.

[7] J. Canny, "A computational approach to edge detection," IEEE Trans.Pattern Anal. Machine Intell., vol. 8, no. 6, pp. 679–698, 1986.

[8] A. Chakraborty, L. H. Satib, and J. S. Duncan, "Deformable boundary finding in medical images by integrating gradient and region information," IEEE Trans. Med. Imag., vol. 15, pp. 859–870, June 1996.

[9] J. M. B. Dias and J. M. N. Lei"Wall Position and thickness estimation from sequences of echocardiographic images," IEEE Trans. Med. Imag., vol. 15, no. 1, pp. 25–38, 1996.

[10] K.H.L Ho., "FEDGE- fuzzy edge detection by fuzzy categorization and classification of edges," IJCAI'95 Workshop, pp. 182-196, Montral, Canada, 95.

[11] T. Kristensen, R. Patel," Edge detection in a lateral inhibition network ," Proceedings of the 2002 International Joint Conference on Neural Networks,vol. 2 , pp. 1620-1624, 12-17 May 2002.

[12] T. Law, H. Itoh, and H. Seki, "Image filtering, edge detection, and edge tracing using fuzzy reasoning," IEEE Trans. Pattern Anal. Mach. Intell., vol. 18, no. 5, pp. 481–491, 1996.

[13] S. Lu and A. Szeto, "Hierarchical artificial neural networks for edge enhancement," Pattern Recognition, vol. 26, no. 8, pp. 1149–1163, 1993.

[14] S. Lu, Z. Wang, " Fuzzy neural networks for edge detection ," IEEE 1997 Canadian Conference on Electrical and Computer Engineering, vol. 2 , pp. 446-449, 25-28 May 1997.

[15] A. Rafiee Kerachi, M.H. Moradi, M.R. Farzaneh," Impulse noise reduction in sonography images by using neuro-fuzzy filters", The Second Iranian Conference on Machine Vision, Image Processing & Applications, MVIP2003, KhajehNasir Al-Din Toosi University Tehran Iran 2003.

[16] A. Rafiee Kerachi, M.H. Moradi, M.R. Farzaneh, " Speckle noise reduction in sonography images by using online genetic-neuro-fuzzy filters", Proceedong of the 4th seminars on fuzzy sets and it's applications, pp. 70-77, 28-29, MAY Babolsar, Iran, 2003.

[17] F. Russo and G. Ramponi, "Edge extraction by FIRE operators," in Proc. 3rd IEEE Int. Conf. Fuzzy Syst. P. 249, Piscataway, NJ, 1994.

[18] N. Saifipour, On-line genetic algorithm in optimization and control, Ph.D. Thesis Amirkabir university, 2002.

[19] C.R. De Silva, L.C. De Silva, S. Ranganath, " An edge detection scheme using radial basis function networks ",Proceedings of the 2000 IEEE Signal Processing Society Workshop Neural Networks for Signal Processing X, vol. 2, pp. 604-613,11-13 Dec. 2000.

[20] C. W. Tao, W. E. Thompson, and J. S. Taur, "A fuzzy if–then approach to edge detection," in Proc. 2nd IEEE Int. Conf. Fuzzy Syst., Piscataway, 1993.

[21] H.R.Tizhoosh, "Fast fuzzy edge detection," Proceedings NAFIPS 2002 Annual Meeting of the North American Fuzzy Information Processing Society, pp. 239-242, 27-29 June 2002.

[22] C. T. Tsai, Y. N. Sun, P. C. Chung, and J. S. Lee, "Endocardial boundary detection using a neural network," Pattern Recognition, vol. 26, no. 7, pp. 1057–1068, 1993.

[23] C. Tyan and P. Wang, "Image processing—Enhancement, filtering, and edge detection using the

fuzzy logic approach," in Proc. 2nd IEEE Conf. Fuzzy Syst., Piscataway, NJ, pp. 600–605, 1993.

[24] Y.Uchiyama,M. Haseyama, H. Kitajima.,"Hopfield neural networks for edge detection ," The 2001 IEEE International Symposium on Circuits and Systems, vol. 3 , pp. 608-611, 6-9 May 2001. WA, 1991.

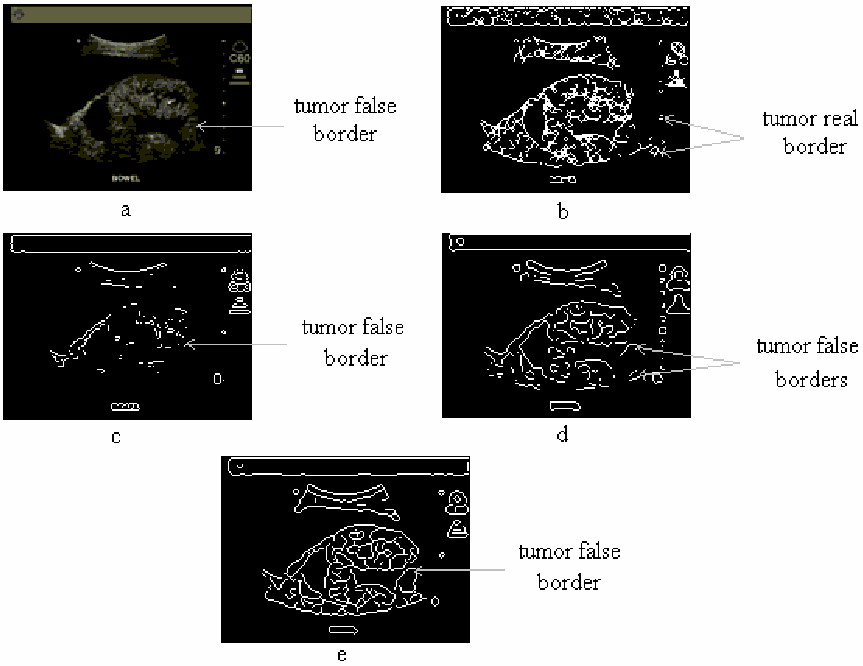

Fig. 6: a) Main Bowel image, b) Genetic-neuro-fuzzy edge detector output, c) Sobel operator's output, d) Zero-crossing edge detector output, e) Canny edge detector output.

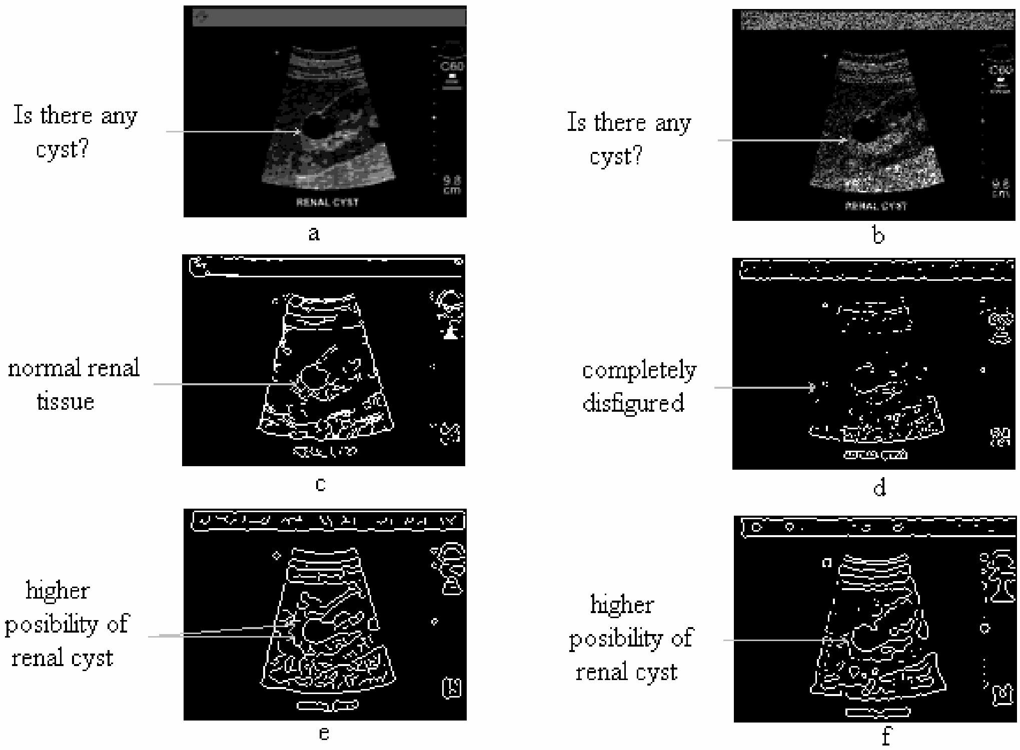

Fig. 7: a) Main Renal image, b) image corrupted by 0.1 speckle noise. c) Genetic-neuro-fuzzy edge detector output, d) Sobel operator's output, e) Zero-crossing output, f) Canny edge detector output.