IMAGE SEGMENTATION AND EDGE DETECTION USING AN AUTOADAPTIVE NEURO-FUZZY SYSTEM

Shweta Lawanya Rao*

Ravi Mishra**

C.L. Chandrakar**

ABSTRACT

An autoadaptive neuro-fuzzy segmentation and edge detection architecture is presented. The system consist of a multilayer perceptron (MLP)-like network that performs image segmentation by adaptive thresholding of the input image using labels automatically pre-selected by a fuzzy clustering technique. The proposed architecture is feedforward, but unlike the conventional MLP the learning is unsupervised. The output status of the network is described as a fuzzy set. Fuzzy entropy is used as a measure of the error of the segmentation system as well as a criterion for determining potential edge pixels. The proposed system is capable to perform automatic multilevel segmentation of images, based solely on information contained by the image itself. No a priori assumptions whatsoever are made about the image (type, features, contents, stochastic model, etc.). Such an "universal" algorithm is most useful for applications that are supposed to work with different (and possibly initially unknown) types of images. The proposed system can be readily employed, "as is," or as a basic building block by a more sophisticated and/or application-specific image segmentation algorithm. By monitoring the fuzzy entropy relaxation process, the system is able to detect edge pixels.

Keyword: *Adaptive Thresholding, Fuzzy Entropy, Image Segmentation, Neuro-Fuzzy System, Self-Organizing System.*

*ME, VLSI, SSCET, Bhilai.

**Senior Assistant Professor, SSCET, Bhilai.

INTRODUCTION

IMAGE segmentation is the technique of decomposing an image into meaningful parts, or objects. It results in a segmented image, where each object is labeled in a way that facilitates the description of the original image so that it can be interpreted by the system that handles the image. In general,the classification of an image's pixels as belonging to one of the "objects" (i.e., classes) composing the image is based on some common feature(s), or resemblance to some pattern.In order to determine which are the features that can lead to a successful classification, some *a priori* knowledge or/and assumptions about the image are usually required. It is due to this fact that the best results are obtained by segmentation algorithms "tailored" for specific applications (but which will perform poorly on applications other than the one they were designed for).

The majority of the segmentation algorithms produce two level, or "object and background" segmentation. While such a result is appropriate for some of the 'classical' image processing applications such as the automatic image analysis of documents or industrial parts, it is not satisfactory for applications dealing with more complex scenes, where several objects have to be detected.

In this paper, we propose a system capable to perform multilevel segmentation of images in an automatic/unsupervised way. No *a priori* assumptions whatsoever are made about the image (type, features, contents, stochastic model, etc.). Such an "universal" algorithm is most useful for applications that are supposed to work with different (and possibly initially unknown) types of images (e.g., searching for images on the Internet or in the photo archive of a magazine). The proposed system can be readily employed, "as is," or as a basic building block by a more sophisticated image segmentation algorithm (that incorporates additional "knowledge" into different parts of the system).The proposed neuro-fuzzy segmentation system is self organizing.It consists of a multilayer perceptron (MLP)-like network that performs image segmentation by adaptive thresholding of the input image using labels automatically pre-selected by a fuzzy clustering technique. The proposed architecture is feed forward, but unlike the conventional MLP, the learning is unsupervised. The output status of the network is described as a fuzzy set. Fuzzy entropy is used as a measure of the error of the segmentation system. Because this measure handles only one aspect of the quality of the segmentation, and because no satisfactory quality measures were proposed in the literature, the results are analyzed mainly by visual inspection and comparison with results obtained by other algorithms.

One of the main roadblocks toward full automation of the segmentation system is the problem of automatically choosing the correct number of labels. This parameter, of crucial importance to most labeling methods, is usually very hard to determine automatically,and in most cases it is left as a parameter which has to be provided by the user. Some methods of employing cluster validity measures to solve this problem were tested, and some preliminary results are presented in this paper.

METHODOLOGY

Overview of the Proposed System The proposed system consists of a multilayer neural network that performs adaptive, ultilevel thresholding of the image using labels automatically preselected by a fuzzy clustering technique. The learning technique employed is self-supervised allowing, therefore, automatic adaptation of the NN. The output status of the network is described as a fuzzy partition. Fuzzy entropy is used as a measure of the error of the system as well as a criterion for determining potential edge pixels. Given an input image, the system is forced to evolve toward a minimum fuzzy entropy state in order to obtain image segmentation. Pixels most affected by the consecutive training iterations (due to the amount of their contribution to the fuzzy entropy of the system) are labelled as edge pixels.

System Description

A general flowchart of the proposed algorithm is depicted in Fig. 1. First, labels are found by applying the FCM algorithm to the image histogram. Then, the information about the labels is employed to build the network's activation and error functions. The input to a neuron in the input layer is normalized between [0–1], proportionally to the gray value of the correspondent pixel. The image information is first propagated forward using (11) to get the output status of the network. The output value of each neuron lies in the interval [0–1]. Then, the output error is calculated and then back-propagated to update the weights [(14)]. Training continues either until a minimum error or until a maximum number of iterations is reached. The output of the system at this stage constitutes the segmented image. Integrating (summing) the thresholded (binarized) differences between the output at consecutive epochs yields the edge image.

Viewed as a system, the proposed algorithm consists of three main processing blocks :

The (fuzzy) error function definition block (A),

The adaptive thresholding block (B) and

The optional edge detection block (C).

1) **Error Function Definition (Block A):** The purpose of this stage is to provide the objective error function to be used by the adaptive thresholding stage (for unsupervised training). First, the input image is fuzzified based on its gray-level histogram and then the error function is obtained by determining the contribution of each graylevel to the fuzzy entropy of the partition (any fuzziness measure can be employed). Two such error functions are depicted in Fig. 3.

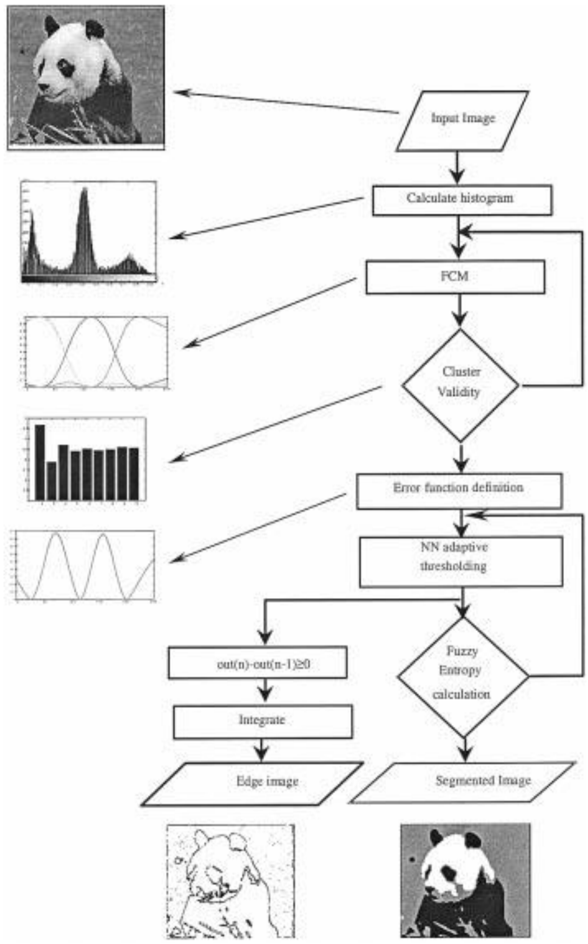

Fig. 1. General flowchart.

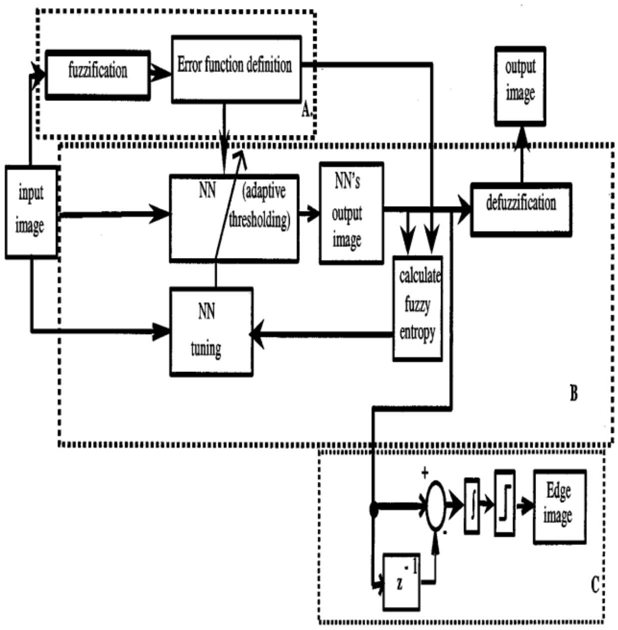

Fig. 2. Block diagram of proposed system.

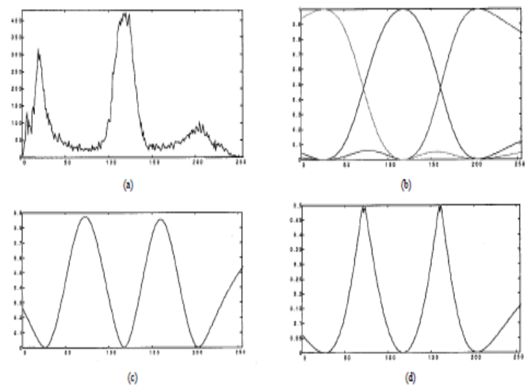

Fig. 3. Error functions. (a) Histogram of the Panda image (b) Partition found by FCM. (c) Fuzzy entropy [(8)]. (d) Linear

In our study, the FCM algorithm was employed to create a fuzzy partition that properly describes the image, using only the pixel intensity feature. Other clustering algorithms might be used as well, but because that in the case of 1-D data the shape of the cluster boundaries is of no importance, variations of the objective function minimized by the algorithm (such as Fuzzy-Shells or the more general Gustafson–Kessel algorithms) are of no specific interest here. Some test were run using possibilistic clustering, but no significant difference in the cluster boundaries was found, while the more irregular shape of fuzzy sets obtained eventually slowed down the convergence of the system. In order to keep the system totally autonomous, an automatic way to determine , the right number of clusters, is needed. In the proposed system, this was done by iterating the FCM algorithm for a range of hypothesized numbers of clusters and choosing the best option based on a cluster validity measure.

Although in general cluster validity measures are not considered very reliable, some of them (e.g., the partition coefficient and the partition entropy, Bezdek [28]) yielded surprisingly good results for some of the test images. This might be partially due to the special nature of the data, which is not common in clustering problems: the data is 1-D and at least one entry exists at each possible point. Still, in general, the problems involved in this choice are (yet) unsolvable, thus leaving this parameter as the only part of the system requiring the (yet) non replaceable human intervention. Supposing that someone does rely on one of the cluster validity measures, another decision has to be made: on one hand, the correct can be chosen as the one providing the extremal value for the specific validity measure employed. On the other hand, accepting the fact that different values of might fit the given data (e.g., for segmentation at different detail), a threshold on the validity measure should be chosen below/above which is accepted. There is (yet) no theoretical way to define such a threshold. Attempts to do this are done based on application/data specific heuristics, like how many different 's are acceptable, or how far can the threshold be from the extremal value of the validity measure.

2) Adaptive Thresholding (Block B): This stage contains the adaptive thresholding system itself, the fuzzy entropy calculation block and the training/tuning algorithm of the adaptive thresholding system. Its inputs are the input image and the error function determined by stage A. and its output is the segmented image. The adaptive thresholding system is based on a self-organizing neural network (Fig. 4).

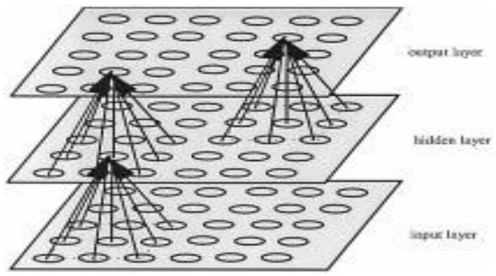

Fig. 4. Network architecture.

In the proposed system, the neuron activation function and the error measuring function were completely modified in order to deal with multilevel segmentation. The network consists of an input layer, an output layer and at least one hidden layer (Fig. 4). Each layer consists of M*N neurons, every neuron corresponding to an image pixel. Each neuron in one layer is only connected to the corresponding neuron in the previous layer and the neurons in its d-th order neighborhood (Fig. 5).

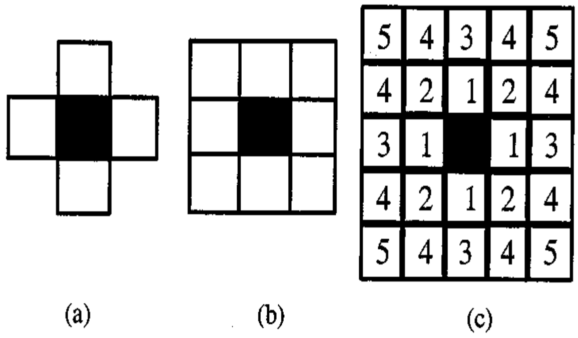

Fig. 5. Neighborhood of a pixel. (a) First order neighborhood; (b) Second order neighborhood; (c) Sequence of neighborhoods.

Activation function: In order to allow more than two stable states of the neuron output (i.e., more than two allowable intensity levels), a special activation function was developed. It consists of a sigmoid-like function with multiple levels. The *multisigmoid* is obtained by the superposition of shifted sigmoid functions (being the number of classes). The rationale behind using such a function was to enable stability around the multiple targeted pixel values. We define the *multisigmoid* function as (Fig. 6)

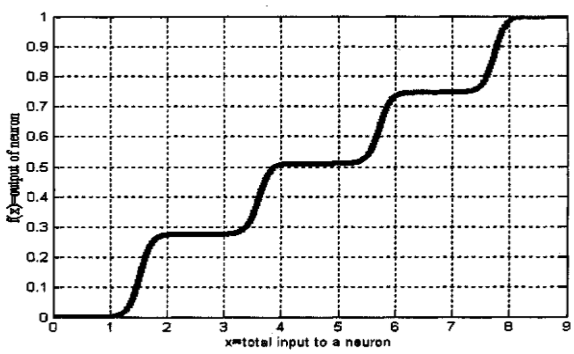

Fig. 6. Multisigmoid (example).

$$f(x) = \sum_k \left(\frac{y_k - y_{k-1}}{1 + e^{-(x-\theta_k)/\theta_0}} + y_{k-1} \right) \times \left[u(x - y_{k-1} * d^2) - u(x - y_k * d^2) \right] \quad (11)$$

where

u step function;

θ_k thresholds;

y_k target level of each sigmoid, will constitute the systems' labels;

θ_0 steepness parameter;

d size of the neighborhood, as defined in the previous section.

The thresholds and the target values are obtained from the error function, as the graylevels with the maximal and with minimal levels of fuzziness respectively. Because the range of the neuron input levels depends on the number of neurons in the previous layer to which it is connected (the size of the neighborhood), the threshold values are adapted to reflect this dependency (by multiplying them by , the number of input links).

4**) Training**: The back-propagation algorithm is employed fortraining . The weights are updated as follows:

$$\Delta w_{ji} = \begin{cases} \eta \left(-\frac{\partial E}{\partial o_j} \right) \frac{\partial o_j}{\partial I_j} o_i & \text{output layer} \\ \eta \left(\sum_k \left(-\frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial I_k} w_{kj} \right) \right) \frac{\partial o_j}{\partial I_j} o_i & \text{other layers} \end{cases} \quad (12)$$

where

I_i total input to the ith neuron;

w_{ji} weight of link from neuron i in one layer to neuron j in the next layer;

o_i output of the ith neuron in the previous layer;

E error in the network's output (relative to the desired target image);

η learning rate.

Fuzziness measure/Error function: The error function defined at the first stage of the system is evaluated for the image output at each training epoch. As discussed before, the aim of the network is to reduce the degree of fuzziness of the input image.

Defuzzification: As it should be understood from the descriptions of the neural network, it is "working" directly on the graylevel values, and not on their fuzzy membership values, as would be the case in a usual fuzzy image processing algorithm. In other words, the network does not change the fuzzy membership values of the pixels in order to reduce the error. Instead, it changes the initial pixel values to values that will decrease the amount of fuzziness, according to the initial fuzzy partition. Thus, the output of the neural network is initially obtained in terms of graylevels, which are then "fuzzyfied" in order to determine the error. In the ideal case when the network converges with no error at all, the outputs have only values who's membership values are "1" or "0," defuzzification is not necessary. When the network does not converge completely (whether stopped intentionally or not), the fuzzification of the output image does not result in merely crisp membership values. The information about the membership values of the pixels might be useful for further processing, depending on the application at hand. If crisp labeling is required, a defuzzification stage must be added. For display purposes, the simplest defuzzification method is thresholding the fuzzy partition, so that each pixel is uniquely assigned to the class in which it has the highest membership value. After being assigned to a single class, the pixels can be labeled either with the value of the prototype of their class, or with the value of the prototype weighted by the membership values of the pixels in their class (thus preserving some of the "fuzzy" information).

Edge Detection (Block C)

The edge detection subsystem was added during the research in order to investigate the possibility to exploit the fuzzy entropy information for other applications. This subsystem is based on the assumption that the edge pixels have the most ambiguous values in the image, i.e., they give the largest contribution to the fuzzy entropy of the output image at each iteration. Thus,these pixels are those that undergo the greatest changes during the training/tuning of the system. Here, the edge image is obtained by monitoring the changes that take place in the pixels' values between two consecutive iterations and integrating these changes over the whole training period.

EXPECTED OUTPUT

Segmentation Output-

The comparison between the segmented image obtained by means of the proposed system and some other algorithms is shown in Fig. 7. Fig. 7(c). shows the results of hard-thresholding the image using the thresholds determined by the FCM algorithm. Figs. 7(d)(f) show the results produced by three algorithms available in the Khoros environment. It can be observedthat the proposed system produces much smoother results (less "noisy" regions), which is a desirable feature, even at the price of loosing some information. The region growing algorithm used here, which might not be the best one of it's kind, produced a clearly inferior result in this case, because it gives different labels to separated (nonconvex) regions even if they are of the same kind. Note that due to the good behavior (well defined clusters) of the example, the results obtained by using the FCM algorithm (Fig. 7(c)) and the (Hard) - Means algorithm are almost the same (only a few pixels differ).

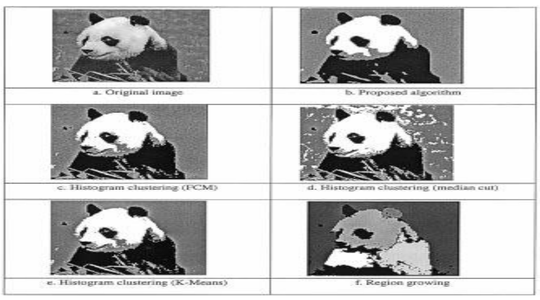

Fig. 7. Segmentation results and comparison.

Fig. 8. Effects of increasing the size of the neighborhood.

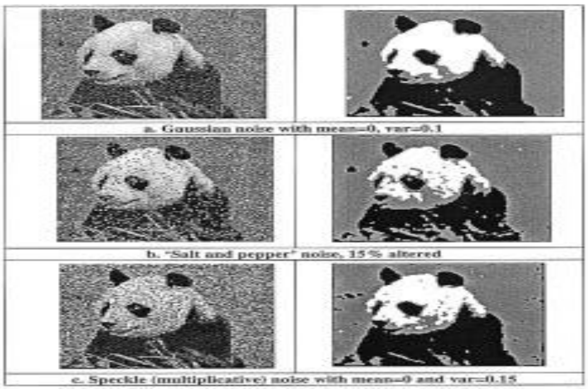

Fig. 9. Noise immunity.

Fig. 8 shows the effects of increasing the size of the neighborhood. It can be seen that as the neighborhood is increased,better smoothing is obtained, but more detailed information is lost. As a conclusion, when dealing with noisy images the use of a larger neighborhood size is desirable. Otherwise, smallerneighborhoods should be used to obtain more detail. Other parameters,like the type of error function used (partition entropy,linear index of fuzziness, etc.) and the steepness factor of the multisigmoid activation function did not show any significant influence on the outcome of the segmentation algorithm (they do influence the rate of convergence. The proposed system was found to be robust to some real life 'complications' like the addition of noise, and changing illumination conditions. For example, Fig. 9 shows the segmentation results of some quite heavily noised versions of the Panda image. Fig. 10 shows a darkened version of the Panda image, its Histogram and the segmented image.

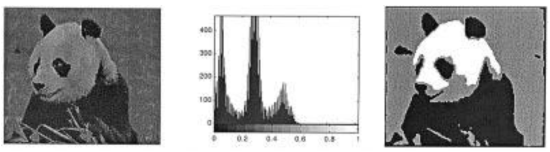

Fig. 10. Invariance with illumination conditions.

Edge Detection Results

The edge detection subsystem was found to perform poorly compared to some of the better edge detection algorithms existing today, sometimes even worse than the classical, gradient type edge detectors (Prewitt, Sobel, etc.). Yet, the results are good enough to prove the point that the process of fuzzy entropy minimization does provide information about the 'edginess' of the pixels in an image, in addition to the segmentation. Furthermore, the algorithm proposed here has the advantage of being totally automatic. Also, some typical post-processing (like edge linking) might enhance its performance. Fig. 11 shows the results of the edge detection subsystem applied to the Panda image.,compared to the results of the (thresholded) Sobel operator and of the optimal Difference Recursive Filter (DRF) for edge detection algorithm. This filter, proposed by Shen and Castan [33], is an optimal smoothing filter (symmetric exponential filter of infinitely large window size) based on a one step model (a step edge and the white noise) and the multi-edge model.

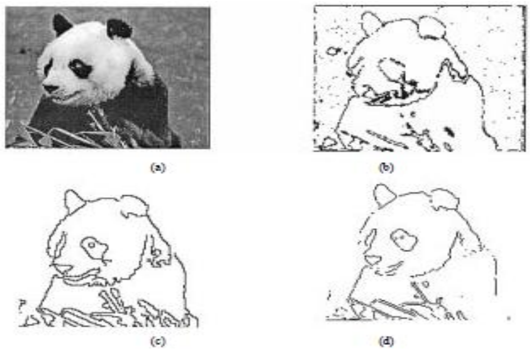

Fig. 11. Edge detection results. (a) Original image. (b) Proposed algorithm. (c) DRF. (d) Sobel (thresholded).

CONCLUSION

An adaptive system for the automatic segmentation of intensity level images was presented in this paper. In order to achieve the ability to robustly perform across the wide diversity of images encountered in real world applications, resemblance to the human visual system was sought during the design of the system. The "heart" of the proposed system, the neural network that performs the adaptive thresholding, is biologically inspired (beyond the simple modeling of artificial neurons): the pyramidal averaging effect, which is a result of its special architecture, is a known feature of the human visual system; also, the learning process based on the minimization of the fuzzy entropy function is a kind of relaxation process, which is typical of physical systems. The FCM algorithm, used to determine the thresholds and the error function, was designed based on human heuristics of what a "good" partition should be like. Although our aim was to concentrate on the research of the concepts described above rather than to achieve optimal segmentation results, the results we obtained are comparable with those of other known methods and sometimes/from certain points of view might even outperform them. In addition, the proposed system has the advantage that it does not require any human expert intervention, nor any *a priori* information about the input image.

In this study, minimization of an image's fuzzy entropy was employed as the target of an MLP neural network's training,thus transforming it into an unsupervised training scheme. Obtaining image segmentation by means of reducing the (fuzzy) entropy of the image can be argued "philosophically" considering both the physical and the information theory significance of "entropy." The segmentation process may be regarded as one that reduces the uncertainty in the image (about the belonging of pixels to objects), as well as one that results in gain of information (about the structure of the image). In addition to using fuzzy entropy as a measure of the error of the segmentation system, this reasoning also led to its use as a criterion for determining potential edge pixels (here it might be interesting to mention the fact that the segmentation and edge detection operations are complementary aspects of the object extraction problem). It was found that some of the "classical" cluster validity measures can be relied upon in some cases (e.g., industrial applications) to automatically choose the number of clusters (segments) that appear in the image. The system provided good results when tested on noisy images (different types of noise with low SNR were tested) and under changing illumination conditions. Convergence of the adaptive thresholding system was not proven analytically, but it was found experimentally in all the (over 100) test cases. The research issues/directions that might provide the greatest benefit are:

• Experimenting with clustering algorithms other than the FCM algorithm, in order to find one that can detect 'small' clusters (the FCM algorithm tends to form partitions where the number of data points in all the clusters is similar). Fig. 12. No caption needed.

• Working with more features at the different processing stages (data fusion). The label choosing system (i.e., the clustering algorithm) can be easily adapted to deal with more features, but in the case of the neural network the implementation of this task is less trivial. In this latter case, some changes to the network architecture are probably required (like linking multiple weights in parallel between the neurons, one for each feature), besides adapting the activation and the error functions. Of most importance are the inclusions of color information (RGB, HUV, etc.) and of local features (gradient value, mean local value, etc.).

• Conduct multi-resolution analysis (look for segments at different resolutions) either selecting an optimal global or local window sizes or by integrating the results obtained at different resolutions.

• Further research on validity and convergence of the segmentation system, especially in difficult real-life images such as Fig. 12.

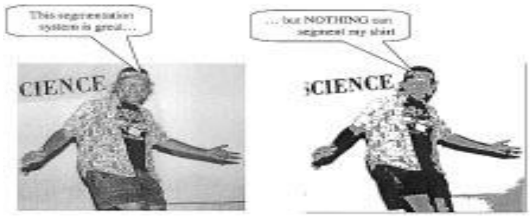

Fig. 28. No caption needed.

REFERENCES

1. B. Jahne, *Digital Image Processing*, 2nd ed. New York:Springer-Verlag, 1993.
2. K. Jain, *Fundamentals of Digital Image Processing*. Upper SaddleRiver, NJ: Prentice Hall, 1989.
3. A. Rosenfeld and A. Kak, *Digital Picture Processing*. New York: Academic,1982, vol. II.
4. N. R. Pal and S. K. Pal, “A review on image segmentation techniques,”*Pattern Recog.*, vol. 26, pp. 1277–1294, 1993.
5. B. Kosko, *Neural Networks for Signal Processing*. Upper SaddleRiver, NJ: Prentice-Hall, 1992.
6. B. J. Krose and P. van der Smagt, *An Introduction to Neural Networks*,5th ed.Amsterdam, The Netherlands: Univ. of Amsterdam, 1993.
7. T. Kohonen, *Self-Organization and Associative Memory*. New York:Springer-Verlag, 1989.
8. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*.New York: Wiley, 1973.
9. A. Ghosh, N. R. Pal, and S. K. Pal, “Self-organization for object extractionusing a multilayer neural network and fuzziness measures,” *IEEETrans. Fuzzy Syst.*, vol. 1, pp. 54–69, Feb. 1993.
10. A. Ghosh, “Use of fuzziness measures in layered networks for object extraction: A generalization,” *Fuzzy Sets Syst.*, vol. 72, pp. 331–348, 1995.
11. L. O. Hall, A. Bensaid, L. Clarke, R. Velthuizen, M. Silbiger, and J. Bezdek, “A comparison of neural network and fuzzy clustering techniques in segmenting magnetic resonance images of the brain,” *IEEE Trans. Neural Networks*, vol. 3, pp. 672–682, Sept. 1992.
12. J. C. Bezdek, L. O. Hall, and L. P. Clarke, “Review of MR image segmentation techniques using pattern recognition,” *Med. Phys.*, vol. 20, pp. 1033–1048, 1993.

13. N. Pal, J. Bezdek, and E. Tsao, "Generalized clustering networks and Kohonen's self-organizing scheme," *IEEE Trans. Neural Networks*, vol.4, pp. 549–557, July 1993. 262 IEEE TRANSACTIONS ON FUZZY SYSTEMS, VOL. 10, NO. 2, APRIL 2002
14. G. Carpenter and S. Grossberg, "A massively parallel architecture for a self-organizing neural pattern recognition machine," *Comp. Vision Graph. Image Process*, vol. 37, pp. 54–115, 1987.
15. S. Hidemoto and J. Toriwaki, "Automatic segmentation of head MRI images by knowledge guided thresholding," *Comp. Med. Imag. Graph.*, vol. 15, pp. 233–240, 1991.
16. S. G. Romaniuk and L. O. Hall, "Dynamic neural networks with the use of divide and conquer," *Proc. IEEE Int. Joint Conf. Neural Networks*, vol. I, pp. 658–663, 1992.
17. H. R. Tizhoosh, *Fuzzy Image Processing*. Berlin, Germany: Springer-Verlag, 1997. in German.
18. J. C. Bezdek and S. K. Pal, Eds., *Fuzzy Models for Pattern Recognition*. New York: IEEE Press, 1992.
19. A. M. Bensaid, L. O. Hall, and J. C. Bezdek *et al.*, "Validity-guided (Re)clustering with applications to image segmentation," *IEEE Trans. Fuzzy Syst.*, vol. 4, pp. 113–123, 1996.
20. Y. A. Tolias and S. M. Panas, "Image segmentation by a fuzzy clustering algorithm using adaptive spatially constrained membership functions," *IEEE Trans. Syst., Man, Cybern. A*, vol. 28, pp. 359–369, May 1998.
21. R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering,"*IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 98–110, May 1993.
22. S. Di Zenzo, L. Cinque, and S. Levialdi, "Image thresholding using fuzzy entropies," *IEEE Trans. Syst., Man, Cybern. Part B*, vol. 28, pp. 15–23, Feb. 1998.
23. N. R. Pal and S. K. Pal, "Object-background segmentation using new definitions of entropy," in *Proc. Inst. Elec. Eng.*, vol. 136, 1989, pp. 284–295.
24. H. D. Cheng, C. H. Chen, H. H. Chiu, and H. Xu, "Fuzzy homogeneity approach to multilevel thresholding," *IEEE Trans. Image Processing*, vol. 7, pp. 1084–1086, July 1998.
25. J. C. Bezdek, E. C. K. Tsao, and N. R. Pal, "Fuzzy Kohonen clustering networks," in *Proc. Int. Conf. Fuzzy Systems*, San Diego, CA, 1992, pp. 1035–1043.

26. J. P. Davis, T. M. Warms, and W. R. Winters, "A neural net implementation of the fuzzy c-means clustering algorithm," *Proc. IEEE Int. Joint Conf. Neural Networks*, p. A-953, 1991.
27. J. S. Lin, K. S. Cheng, and C. W. Mao, "Multispectral magnetic resonance images segmentation using fuzzy Hopfield neural network," *Int. J. Bio. Med. Comp.*, vol. 42, pp. 205–214, 1996.
28. J. C. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*.New York: Plenum Press, 1981.
29. A. De Luca and S. Termini, "A definition of non probabilistic entropy in the setting of fuzzy set theory," *Inform. Control*, vol. 20, pp. 301–312,1972.
30. B. Kosko, "Fuzzy entropy and conditioning," *Inform. Sci.*, vol. 40, pp.165–174, 1986.
31. R. R. Yager, "On the measure of fuzziness and negation part I: Membership in the unit interval," *Int. J. Gen. Syst.*, vol. 5, pp. 221–229, 1979.
32. R. P. Lippmann, "An introduction to computing with neural nets," *IEEE ASSP Mag.*, pp. 4–22, 1987.
33. J. Shen and S. Castan, "Edge detection based on multi-edge models," in *Proc. SPIE*, Cannes, France, 1987, pp. 46–53.
34. V. Boskovitz and H. Guterman, "Neuro-fuzzy system for adaptive multilevel image segmentation," *Proc. 19th IEEE Conf. EE Engineers Israel*, pp. 208–211, 1996.
35. "Self-tuning object extraction for online monitoring of biotechnological processes," in *Proc. Int. Conf. Fuzzy Logic Application*, Zichron- Yaakov, Israel, 1997, pp. 174–182.
36. "Edge detection and image segmentation: Two sides of the same coin," *Proc. 6th IEEE Int. Conf. Fuzzy Systems*, pp. 1063–1068, 1997.