

# A Hybrid Edge Detection Algorithm for Salt-and-Pepper Noise

H. Farahanirad, J. Shanbehzadeh, M. M. Pedram, A.Sarrafzadeh

**Abstract**—This paper presents a hybrid edge detection algorithm in situations where the image is corrupted by Salt-and-Pepper noise. Edge detection is an important pre-processing step in image analysis. Successful results of image analysis extremely depend on edge detection. Up to now several edge detection methods have been developed such as Roberts, Prewitt, Sobel, Zero-crossing, Canny, etc. But, they are sensitive to noise. The structure of our proposed edge detector, to make the process robust against noise, is a combination of neural networks, neuro-fuzzy network and adaptive median filter. The internal parameters of these networks are adaptively optimized by training using very simple artificial images that can be generated by computer. The proposed method is tested under noisy conditions on several images and also compared with conventional edge detectors such as Sobel and Canny and a neuro-fuzzy edge detector. Experimental results reveal that the proposed method exhibits better performance and may efficiently be used for the detection of edges in images corrupted by Salt-and-Pepper noise.

**Index Terms**— edge detection, image processing, neural network, neuro-fuzzy network, salt and pepper noise.

## I. INTRODUCTION

THE edges are place of abrupt changes in the intensity of image and are most essential feature of an image. They contain a richness of internal information of image [1],[2]. Edge always inhabits in two neighboring areas having different gray level and it mostly exists between objects or objects and background [3],[4]. Edge detection is one of the most important tasks in pattern recognition and image processing [5]. It plays an important role in the multimedia and computer vision, Image Understanding, Image enhancement and image compression, etc [6]. It is usually the first operation that is performed before tasks such as boundary detection, segmentation, classification, registration, understanding and recognition in image processing scope [7],[14]. Thus, success in such operations extremely depends on edge detection. In recent years several methods have

been developed for edge detection, These methods include mathematical morphology, wavelet transformation, Roberts, Prewitt, Sobel, Zero-crossing, Canny, Laplacian of Gaussian (LOG) etc [5],[8]. The most important factor in decreasing the quality of edge detection is the noise in images. One of the important noises is Salt-and-Pepper. This is described by random distribution of white or black spots. This noise is often happens during image acquisition and/or transmission due to environment condition, quality of sensing elements and communication channels [9]. In most image processing applications, it is of crucial importance to remove the noise from the image because the performances of image processing tasks (such as segmentation, feature extraction, recognition, etc.) are severely degraded by noise [3]. Most edge detectors first try to reduce noise. However, up to now, it is still a major challenging issue to develop robust and efficient edge detection algorithm because a major drawback of edge detectors is their high sensitivity against noise.

This paper presents a novel hybrid method for edge detection while images are corrupted by salt and pepper noise. The proposed method is very simple and includes a neuro-fuzzy network (nf), an adaptive median filter and four identical neural networks (nn). The internal parameters of networks are adaptively adjusted by training. Training of the networks is performed by using a computer generated artificial images. The essential advantage of the proposed method over other operators is that it offers higher edge detection performance under noisy conditions. Experiments show that the proposed operator may be used for efficient edge detector when an image is corrupted by Salt-and-Pepper noise and high accuracy is needed.

The rest of this paper is organized as follows. Section II explains the proposed method and its building blocks. Section III discusses the experimental results of the proposed method and classical methods such as Canny, Sobel and one only neuro-fuzzy method that discussed in [14]. Section IV, which is the final section, presents the conclusions and further research.

## II. PROPOSED METHOD

Fig. 1 shows the structure of the proposed method. It is a hybrid method by combining four similar neural networks (vertical, horizontal, left diagonal and right diagonal), a neuro-fuzzy (NF) network and an adaptive median filter. The NF network utilizes the information from neural networks to compute the output of the system, which is the edge points. A 3-by-3 window moves on the input image, this window moves from the upper left corner of image horizontally and moves to the next row when it arrives to the end of the row. This work repeats for every row and it is

Hossein farahanirad: Department of Computer Engineering, Science and Research-Islamic Azad University, Tehran, Iran.  
(e-mail: h.farahanirad@yahoo.com)

Dr. J. Shanbehzadeh is an associate professor with the Department of Computer Engineering at Tarbiat Moallem University-Tehran, I. R. Iran..(e-mail: jamshid@tmu.ac.ir)

Dr. M. M. Pedram is an Assistant Professor with the Department of Computer Engineering at Tarbiat Moallem University-Tehran, I. R. Iran..(e-mail: pedram@tmu.ac.ir )

Dr. Abdolhossein Sarrafzadeh is an Associate Professor and Head of Department of Computing at Unitech, New Zealand. (e-mail: hsarrafzadeh@unitec.ac.nz)

finished in the lower right side of the image. Input data which are applied to the inputs of the neural networks as show in fig. 1 include:

**X : pixels of image that are corrupted by Salt & Pepper noise.**

**Y : pixels of image that are created by Adaptive median filter from noisy image.**

The output of any network is compared with the ideal output, then it is cleared whether the output is edge or not. Any output of the neural network is used as the input for NF network and the neuro-fuzzy network produces the final output.

### A. Neural network

The neural networks that used in proposed method includes 4-input and 1-output and also contains 2-hidden layer and 1-output layer. It's hidden layer have 15 and 10 neurons respectively and output layer have one neuron. The internal parameters determine by training .when output of neural network converge to ideal output training is end. Activation functions in whole neurons of network are defined as following :

$$F(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

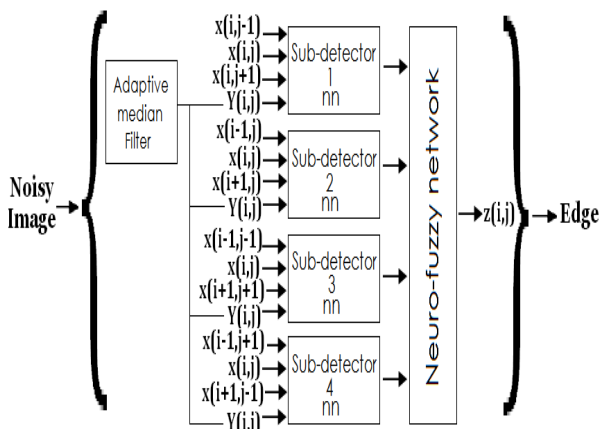


Fig. 1. The proposed method

### B. Neuro-Fuzzy Network

The NF network used in our method finds an edge by combining the information obtained from four neural networks. The rules of the fusion are represented by the rules in the rule base of the NF network and the fusion process is implemented by the fuzzy inference mechanism of NF network. These are explained in detail later in this subsection. The NF network is a first order *Sugeno* type fuzzy system with 4-inputs and 1-output [17]. Each input has 3 *generalized bell* type membership functions and the output has a *linear* membership function. *Sugeno*-type fuzzy systems are well-liked general nonlinear modeling apparatuses because they are very appropriate for tuning by optimization and they employ polynomial type output membership functions, which extremely make less complex

of defuzzification process. The relationship of input-output in the NF network is as follows: Let  $k_1, k_2, k_3$  and  $k_4$  denote the inputs of the NF network and  $Z$  as its output. Hence, in the structure of the proposed model  $k_1, k_2, k_3$  and  $k_4$  represents the output of neural networks for the noisy input pixels and the pixels that created by adaptive median filter. Since the NF network has four inputs and each input has three membership functions, the rule base contains a total of 81 ( $3^4$ ) rules, which are as follows:

- 1) if ( $k_1$  is  $M_{11}$ ) and ( $k_2$  is  $M_{21}$ ) and ( $k_3$  is  $M_{31}$ ) and ( $k_4$  is  $M_{41}$ ), then  $R_1 = F_1(k_1, k_2, k_3, k_4)$ .
- 2) if ( $k_1$  is  $M_{11}$ ) and ( $k_2$  is  $M_{21}$ ) and ( $k_3$  is  $M_{31}$ ) and ( $k_4$  is  $M_{42}$ ), then  $R_2 = F_2(k_1, k_2, k_3, k_4)$ .
- 3) if ( $k_1$  is  $M_{11}$ ) and ( $k_2$  is  $M_{21}$ ) and ( $k_3$  is  $M_{31}$ ) and ( $k_4$  is  $M_{43}$ ), then  $R_3 = F_3(k_1, k_2, k_3, k_4)$ .
- 4) if ( $k_1$  is  $M_{11}$ ) and ( $k_2$  is  $M_{21}$ ) and ( $k_3$  is  $M_{32}$ ) and ( $k_4$  is  $M_{41}$ ), then  $R_4 = F_4(k_1, k_2, k_3, k_4)$ .
- 5) if ( $k_1$  is  $M_{11}$ ) and ( $k_2$  is  $M_{21}$ ) and ( $k_3$  is  $M_{32}$ ) and ( $k_4$  is  $M_{42}$ ), then  $R_5 = F_5(k_1, k_2, k_3, k_4)$ .
- ...
- 81) if ( $k_1$  is  $M_{13}$ ) and ( $k_2$  is  $M_{23}$ ) and ( $k_3$  is  $M_{33}$ ) and ( $k_4$  is  $M_{43}$ ), then  $R_{81} = F_{81}(k_1, k_2, k_3, k_4)$ .

Where  $M_{ij}$  denotes the  $j$ th membership function for the  $i$ th input,  $R_k$  denotes the output of the  $k$ th rule, and  $F_k$  denotes the  $k$ th output membership function, with  $i=1,2,3,4$  and  $j=1,2,3$  and  $k=1,2,3,...,81$ . The input membership functions are generalized bell type:

$$M_{ij}(u) = \frac{1}{1 + \left| \frac{u - a_{ij}}{b_{ij}} \right|^{2c_{ij}}} \quad (2)$$

and the output membership functions are linear

$$F_k(u_1, u_2, u_3, u_4) = d_{k1}u_1 + d_{k2}u_2 + d_{k3}u_3 + d_{k4}u_4 + d_{k5} \quad (3)$$

Where  $u_1, u_2, u_3$  and  $u_4$  are formal parameters and  $a, b, c$  and  $d$  are constants that characterize the shape of the membership functions. The optimal values of these parameters are determined by training. As show in Fig. 2, this Sugeno-type NF network include five layers.

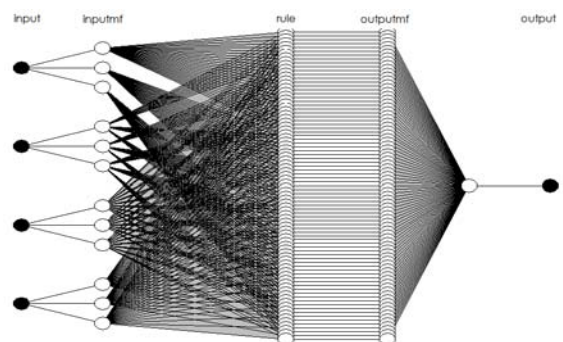


Fig. 2. structure of Sugeno-type neuro-fuzzy network

There is no method to determine the optimum number of the membership functions and it is usually determined heuristically and verified by experiments. It has been experimentally determined that three membership functions offer a very good cost balance. The output of the networks

is the weighted average of the individual rule outputs. The weighting factor of each rule is calculated by appraising the membership expressions in the previous rule. This is performed by first converting the input values to fuzzy membership values by utilizing the input membership functions and then applying the *and* operator to these membership values. The *and* operator corresponds to the multiplication of input membership values. Hence, the weighting factors of the rules are calculated as follows:

$$W_1 = M_{11}(x_1) \cdot M_{21}(x_2) \cdot M_{31}(x_3) \cdot M_{41}(x_4) \quad (4)$$

$$W_2 = M_{11}(x_1) \cdot M_{21}(x_2) \cdot M_{31}(x_3) \cdot M_{42}(x_4)$$

$$W_3 = M_{11}(x_1) \cdot M_{21}(x_2) \cdot M_{31}(x_3) \cdot M_{43}(x_4)$$

$$W_4 = M_{11}(x_1) \cdot M_{21}(x_2) \cdot M_{32}(x_3) \cdot M_{41}(x_4)$$

$$W_5 = M_{11}(x_1) \cdot M_{21}(x_2) \cdot M_{32}(x_3) \cdot M_{42}(x_4)$$

·  
·  
·

$$W_{81} = M_{13}(x_1) \cdot M_{23}(x_2) \cdot M_{33}(x_3) \cdot M_{43}(x_4)$$

When the weighting factors are obtained, the output of the NF network can be found by calculating the weighted average of the individual rule outputs

$$z = \frac{\sum_{k=1}^{81} w_k R_k}{\sum_{k=1}^{81} w_k} \quad (5)$$

### C. Adaptive median filter

Adaptive median filter can handle salt-and-pepper noise with preserving the details [9]. This filter works in a rectangular window area  $S_{xy}$ . Depending on certain conditions, adaptive median filter changes the size of  $S_{xy}$  during filtering operation. Consider the following notations:

- $z_{min}$  = Minimum intensity value in  $S_{xy}$
- $z_{max}$  = Maximum intensity value in  $S_{xy}$
- $z_{med}$  = Median of intensity values in  $S_{xy}$
- $z_{xy}$  = intensity value at coordinates (x,y)
- $S_{max}$  = Maximum allowed size of  $S_{xy}$

The adaptive median filter algorithm works in two stages, denoted stage A and B, as follows:

- Stage A:  $A1 = z_{med} - z_{min}$   
 $A2 = z_{med} - z_{max}$   
 If  $A1 > 0$  and  $A2 < 0$ , go to stage B  
 Else increase window size  
 If window size  $\leq S_{max}$  repeat stage A  
 Else return  $z_{xy}$
- Stage B:  $B1 = z_{xy} - z_{min}$   
 $B2 = z_{xy} - z_{max}$   
 If  $B1 > 0$  and  $B2 < 0$ , return  $z_{xy}$   
 Else return  $z_{median}$

### D. Training the networks

The fig. 3 shown that the internal parameters of the networks are repeatedly optimized by training in order that its output converges to the output of the *ideal edge detector*

(which generates *real edge image*). The ideal edge detector does not exist in reality and, it is only necessary for training and, this is represented by the original training image. The parameters of the neural networks and NF network are tuned to minimize the learning error by using the *Levenberg-Marquardt* and *back propagation* optimization algorithm respectively [17].

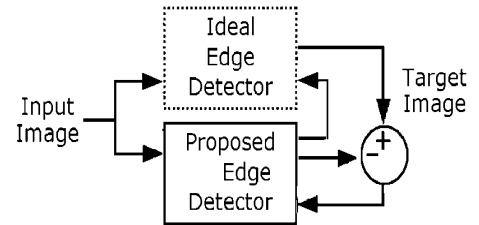


Fig. 3. Training the networks

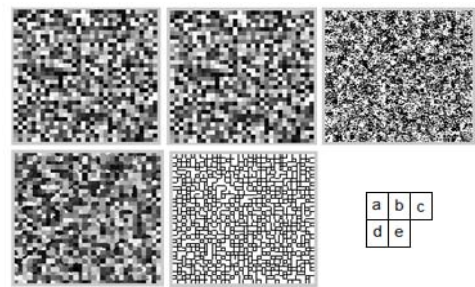


Fig. 4. Training images (128\*128 pixel) : (a)(b) Base training image (c)(d) Input training image, (e) Target training image.

Fig. 4 shows the images employed in the training phase. These images are artificial and can easily be generated by computer. The Fig. 4(a) shows the *base training image*. Each square box in this image has a size of 4-by-4 pixels (16 pixels) and each box have the same brightness value. The image in Fig. 4(b) is generated from Fig. 4(a) by considering the 4 most significant bits of each pixel and the rest 4 bits equal to zero. The image in Fig. 4(c) is the *input training image* and obtained from the image in Fig. 4(b) that is corrupted by Salt-and-Pepper noise. The image in Fig. 4(d) is the *other input training image* and obtained by applying adaptive median filter on the image in Fig. 4(c). The image in Fig. 4(e) is the *target training image*. It is a binary image where its black pixels indicate the edges of the input image. The images in Fig. 4(c) and 4(d) are applied as the input of NN and NF networks and the image in Fig. 4(e) is consider as the target output of neural network and NF network during training.

## III. SIMULATION RESULTS

The proposed method discussed in the previous section is simulated by computer programs. The performance of the method is tested under noisy condition (Salt & Pepper noise) on two test images including *Boats* and *Cameraman*. The images are corrupted by Salt & Pepper noise with 20%, 30%, 40% and 50% noise density before processing. Several experiments are performed to compare our proposed scheme with Sobel, Canny and other neuro-fuzzy scheme that discussed in [14]. Fig. 5 shows the results for images corrupted by 40% noise density. It can be seen that the

performance of the proposed method is much higher than other three schemes (Sobel, Canny and method in [14]). comparison of images in fig. 5 obviously indicated that the proposed method detects most of the edges better than [14] because In this detector after primary detecting final outputs is determined by post-processing but in proposed method final edges is determined by a neuro-fuzzy network using fuzzy techniques that the problems of pixel uncertainties on the edge areas are solved and the edges are recognized clearly. Unlike other three methods effect of noise on the performance of proposed method is very low. In addition, study of experimental results clearly exhibited where intensity variation is smooth the proposed method offers better performance over other three methods for all images. The essential purpose of our method is to increase performance of edge detection in highly noisy condition that it almost is obtained. The test images demonstrated that when images corrupted with high degree of salt and pepper noise Sobel, Canny and the method discussed in [14] no have appropriate performance, while in this condition the proposed method detected the most of edges whit high precision.

#### IV. CONCLUSION

A novel edge detection method for digital images corrupted by Salt & Pepper noise was proposed.

- 1) It has a simple structure, thus this extremely simplified implementation.
- 2) The training is performed without difficulty by using artificial images that can be generated in computer.
- 3) It can detect edge in image that corrupted by high degree of Salt-and-Pepper noise.
- 4) The proposed method could be had good performance for images without noise too.

The simulation result shows that this scheme outperforms conventional edge detectors under noisy conditions. However, the computational barrier of this scheme is higher than Sobel and Canny edge detectors.

#### REFERENCES

- [1] Wenshuo Gao, Lei Yang, Xiaoguang Zhang, Huizhong Liu. *An improved sobel edge detection* . 978-1-4244-5540-9/10/26.00 © IEEE 2010
- [2] Chunxi Ma, Lei Yang. *An Improved Sobel Algorithm Based on Median Filter*. International Conference on Mechanical and Electronics Engineering (ICMEE 2010).
- [3] Bing wand & shaosheng fan. *An improved CANNY edge detection algorithm*.second international workshop on computer science and engineering 2009.
- [4] Feng-ying Cui and Li-jun Zou, Bei Song. *Edge Feature Extraction Based on Digital Image Processing Techniques*.IEEE international conference on Automation and logistics Qingdo,china 2009.
- [5] Zhong-Ren wang Yang-Miung , *a novel approach for edge detection based on the theory of electrostatic fild*. Proceedings of International Symposium on Intelligent Signal Processing and Communication Systems Nov.28-Dec.1, Xiamen, China,2007.
- [6] Xiangjian He, Wenjing Jia and Qiang Wu. *An Approach of Canny Edge Detection with Virtual Hexagonal Image Structure*. Intl. Conf. on Control, Automation, Robotics and Vision Hanoi, Vietnam, 2008.
- [7] Lei Zhang, Mei Xiao, Jian Ma, Hongxun Song. *Edge Detection by Adaptive Neuro-Fuzzy Inference System*. 978-1-4244-4131-0/09/25.00 © IEEE,2009.

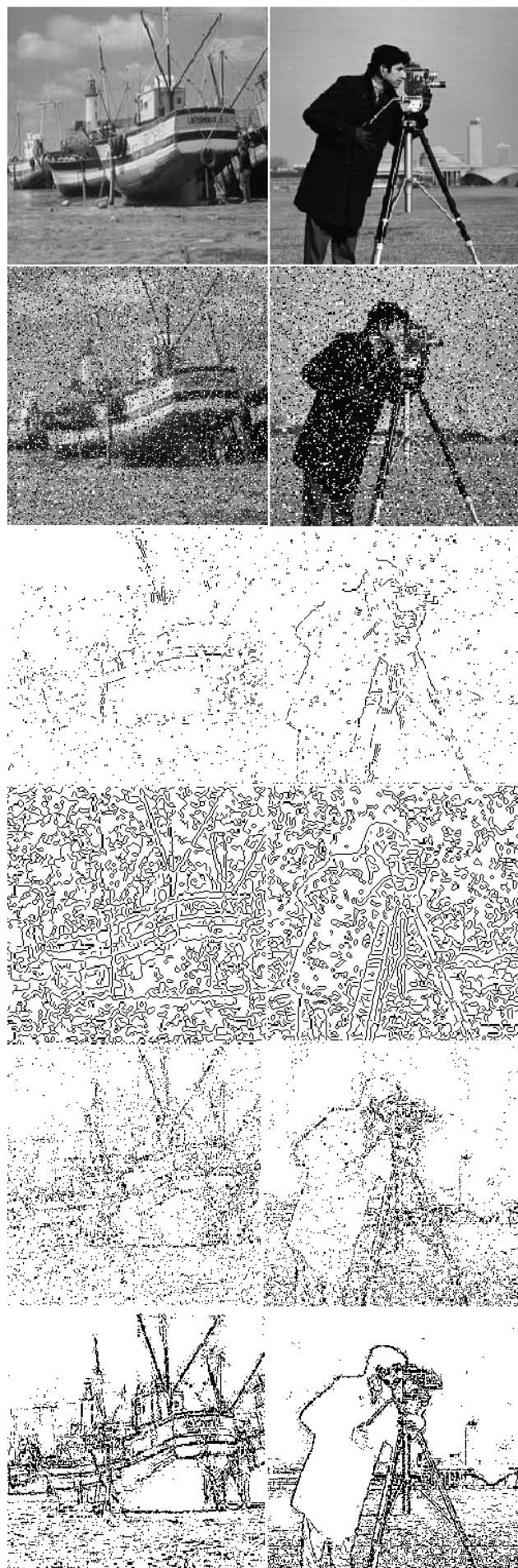


Fig. 5. From up to down: Original image, Noisy image, Sobel method, Canny method, another neuro-fuzzy method[14] and the proposed method.

- on Intelligent Information Technology Application*. 978-0-7695-3497-8/08.00 © 2008 IEEE DOI 10.1109/IITA.2008.
- [8] Xiangjian He, Jianmin Li, Darning wer, Wenjing Jia and Qiang Wu. *Canny Edge Detection on a Virtual Hexagonal Image Structure*. 2009 IEEE .
- [9] Chin-Chen Chang, Ju-Yuan Hsiao, Chih-Ping Hsieh. *An Adaptive Median Filter for Image Denoising*. *Second International Symposium*
- [10] Umbaugh, S. E.: *Computer vision and image processing* . Prentice-Hall International Inc., 1998.
- [11] Albovik, R.: *Handbook of image and video processing*. Academic press, 2000.
- [12] R. Owens, "Lecture 6", *Computer Vision IT412*, 10/29/1997. [http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL\\_COPIES/OWENS/LECT6/node2.html](http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT6/node2.html).
- [13] Linares, I.; Maersereau, R. M.; Smith, M. J. T.: *JPEG estimated spectrum adaptive post filter using image-adaptive Q-tables and Canny edge detectors*. ISCAS-96, IEEE International Symposium on Circuits and Systems, Atlanta, Georgia, USA, 1996.
- [14] M. Emin Yüksel and M. Tülin Yıldırım. *A Simple Neuro-Fuzzy Edge Detector for Digital Images Corrupted by Impulse Noise*. *international journal of Electronics and communication*. (2004) .
- [15] F. Russo and G. Ramponi, "A fuzzy filter for images corrupted by impulse noise" *IEEE Signal Process. Lett* vol. 3, no. 6, pp. 168–170, Jun. 1996.
- [16] S.-J. Lee, C.-S. Ouyang, and S.-H. Du, "A neuro-fuzzy approach for segmentation of human objects in image sequences," *IEEE Trans. Systems, Man, Cybern. B, Cybern.*, vol. 33, no. 3, pp. 420–437, Jun. 2003.
- [17] J.-S. R. Jang, C.-T. Sun , and E. Mizutani , *Neuro-fuzzy and soft computing*. Englewood Cliffs, NJ: Prentice-Hall, 1997.
- [18] M. Emin Yüksel, *A Hybrid Neuro-Fuzzy Filter for Edge Preserving Restoration of Images Corrupted by Impulse Noise*. IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 15, NO. 4, APRIL 2006.
- [19] Ali, M.; Clausi, D. *Using the Canny edge detector for feature extraction and enhancement of remote sensing images*. IGARSS-01, IEEE International Symposium on Geoscience and Remote Sensing, Sydney, Australia, 2001.
- [20] Jang, J. S. R.; Sun, C. T.; Mizutani, E.: *Neuro-fuzzy and soft computing*. Prentice-Hall International Inc., 1997.
- [21] Jinbo Wu, Zhouping Yin, and Youlun Xiong. *The Fast Multilevel Fuzzy Edge Detection of Blurry Images*. IEEE SIGNAL PROCESSING LETTERS, VOL. 14, NO. 5, MAY 2007.
- [22] Wan Yue-Liang, Cao Yuan-Da, Li Dun. *A Single Threshold Edge Detection Algorithm Based on Multiscale Fusion*, 1-4244-0605-6/06/\$20.00 C2006 IEEE.
- [23] Weiqing Li, Chengbiao Wang, Qun Wang, Guangshe Chen. *An Edge Detection Method Based on Optimized BP Neural Network*. © 2008 IEEE.
- [24] Shun-feng Ma, Geng-feng Zheng, Long-xu Jin, Shun-feng Ma, Geng-feng Zheng, *Directional Multiscale Edge Detection Using the Contourlet Transform*, ©2010 IEEE.
- [25] Chunhua Li, Kun He, Jiliu Zhou. *Edge Detection of Image on the Local Feature*. International Symposium on Intelligent Information Technology Application © 2008 IEEE.
- [26] Pal, Sankar K.; Ghosh, Ashish; Kundu, Malay K. *Soft Computing for Image Processing*. 2010.