Using Hierarchical Adaptive Neuro Fuzzy Systems And Design Two New Edge Detectors In Noisy Images

M. H. Olyaee¹, H. Abasi², M. Yaghoobi³

(1) Department of Electrical Engineering, Gonabad Branch, Islamic Azad University, Gonabad, Iran
(2) Department of Engineering, Damghan Branch, Islamic Azad University, Damghan, Iran
(3) Department of Artificial intelligence, Mashhad Branch, Islamic Azad University, Mashhad, Iran

Copyright 2013 © M. H. Olyaee, H. Abasi and M. Yaghoobi. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract
One of the most important topics in image processing is edge detection. Many methods have been proposed for this end but most of them have weak performance in noisy images because noise pixels are determined as edge. In this paper, two new methods are represented based on Hierarchical Adaptive Neuro Fuzzy Systems (HANFIS). Each method consists of desired number of HANFIS operators that receive the value of some neighbouring pixels and decide central pixel is edge or not. Simple train images are used in order to set internal parameters of each HANFIS operator. The presented methods are evaluated by some test images and compared with several popular edge detectors. The experimental results show that these methods are robust against impulse noise and extract edge pixels exactly.

Keywords: Edge detector, Hierarchical Adaptive Neuro fuzzy system, Impulse noise, Image processing.

1 Introduction
The edge detection is considered as one of the most significant topics in the field of image processing and computer vision. Edge detection is used in several problems such as image classification, image segmentation, image recognition and etc [1, 13, 15, 5, 2]. The edge detection is usually used as the first step. So, the results of next steps are related to the accuracy of edge detection phase. Edges are located at the boundary between two dissimilar regions. Edge pixels can be detected easily because their values are so different with their adjacent pixels. But in the presence of noise, this simple definition of edge pixels isn’t good because the values of noise pixels are very different with adjacent too [16]. Most of edge detectors include three basic steps: Smoothing, Detection and localization. But if image is corrupted by noise, noise removal operator will be required. There are many attempts have been made to remove noise exactly. Lou [9] proposed a new method that has two steps: fuzzy impulse detection and impulse noise cancellation. The histogram of image is used at first step that the two peaks at the ends of it give the
intensity values of noise. At second step, the value of noisy pixel is replaced by a linear combination of its original value and the median of noise free pixels. Lou’s method has been adopted by Toh and et al [14]. Their method has two steps similar to the previous one. However, maximum value of gradient in neighbours of noisy pixels is used to decrease the noise. However, these approaches have weak performance in dealing with images without impulse noise.

Another method, based on bio-inspired bacterial foraging (BF) algorithm, have been proposed by Verma and et al [16]. Edge pixels can be extracted by their approach directly in images that are corrupted by impulse noise. Verma’s method uses BF algorithm in order to detect edge pixels and noisy pixels are corrected by fuzzy median filter. But using BF algorithm has increased the computational complexity.

Adaptive Neuro Fuzzy systems is utilized by Yuksel [17] in order to edge detection in noisy images. He suggested an operator that contains a desired number of adaptive Neuro Fuzzy sub detector and a postprocessor. A 3*3 window is moved on the noisy image and each sub detector receives two adjacent with central pixel and determines central pixel is edge or not. The advantage of this method is that edge pixel is determined directly and doesn’t need removal noise operator. But if the window size is increased and the number of neighboring pixels are involved more in decision-making, then it can lead to better results, especially in images with great deal of noise [11]. Nevertheless, by increasing the number of input neuro-fuzzy systems, the training time will be increased exponentially [6, 10].

In this paper two novel methods are presented based on Hierarchical Adaptive Neuro Fuzzy inference systems (HANFIS). Using more pixels and decline the training time are the advantages of these new methods. The experimental results show that these proposed methods perform significantly better than many other well-known techniques and robust against salt and pepper noise and can preserve more details of image.

The remainder of this paper is organized as follows. In Section 2, Adaptive neuro fuzzy inference system is briefly explained. The details of our methods are proposed in Section 3. Experimental results are presented in Section 4. Finally, Section 5 concludes the paper.

2 Adaptive Neuro-Fuzzy Inference System

Neuro fuzzy systems have been used in many areas such as classifications, Rule-based expert systems, Time series predictions and etc [6]. An adaptive neuro fuzzy system is actually a reasoning fuzzy system that can be trained by some train data and its internal parameters be tuned. Using neural networks provide supervised learning to create nonlinear relation between input and output data. For a better understanding, a system with two inputs x and y and an output f is considered based on Fig. 1.

![Figure 1: Structure of Adaptive Neuro Fuzzy Inference System (ANFIS) with two inputs](image)

Action of nodes in each layer is similar to each other. fuzzification is done at first layer. $A_l$ and $B_l$ are linguistics variables that have their particular membership functions. The second layer executes the T-norm of the antecedent part of the fuzzy rules, at third layer, the membership functions is normalized by:

$$\bar{w}_l = \frac{w_l}{w_1+w_2}$$ (1)
At the fourth layer, consequent parameters are calculated based on:
\[ \bar{w}_i f_i = \bar{w}_i (p_i x + q_i y + r_i) \]  (2)

The values of \( r, p \) and \( q \) parameters are determined by training phase. Finally, the last layer computes the overall output as the summation of all incoming signals [6].
\[ \sum_i \bar{w}_i f_i = \sum_i \bar{w}_i \]  (3)

There are various methods in order to train ANFIS that can be referred to [6]. But as previously mentioned, by increasing the number of inputs and membership functions, the number of fuzzy rules exponentially will be increased and too much time will be spent in order to training [10]. To solve this problem, many solutions have been proposed that one of the most effective method is Hierarchical Adaptive Neuro Fuzzy Inference System (HANFIS) [12]. HANFIS includes multiple neuro fuzzy systems that their structures were mentioned previously and in different levels are connected to each other. In HANFIS model, the inputs are put into a collection of low-dimensional fuzzy logic units, instead of a single high-dimensional fuzzy logic system. HANFIS dramatically reduces the total number of involved fuzzy rules. In the conventional HANFISs, the outputs of ANFISs in the previous layer are used as the input linguistic variables of the next layer [8, 7, 12].

3 Proposed methods

3.1. The first method

In order to increase the accuracy of the edge pixels, a 5 x 5 window is moved over the image from beginning to end, and each time from the neighboring pixels, some of them are selected. Structure of operator is shown in Fig. 2.

![Figure 2: Structure of edge detector. The pixels applied to the inputs of each part in the structure are chosen so as to utilize the information from a different pixel neighborhood](image)

Each part in Fig. 2 is an adaptive neuro fuzzy inference system. The outputs of all parts are put to postprocessor. The postprocessor determines the final output based on averaging. Details of each part are seen in Fig. 3. Although all parts have the same structure but their inputs are different. Selected pixels of each part are displayed in Fig. 4.

![Figure 3: Structure of each edge detector part](image)
In Fig. 4, the selected pixels for each part are seen. This choice is done completely optional but usually edge pixels in local area have an order similar a, b, c or d. Each part receives values of central pixels and its neighbors as inputs. The values of neighbors are given to ANFIS at first layer and their outputs with value of central pixel are sent to ANFIS at second layer.

### 3.2 Second method

The total structure of operator is the same as Fig. 2; however, implementation of sub-detectors is based on Fig. 5. In this method, central pixel is ignored and four inputs are used to determination. In this method, neighboring pixels are divided into two groups according to their proximity. Two farther neighboring pixels that are symmetrical to each other will be sent to ANFIS at first layer. The output of first layer along with the two closer neighbors and symmetrical to each other will be sent to ANFIS at second layer. The reason of this decision is that closer neighbors have more decisive role than distant neighbors. Selected pixels of each part are displayed in Fig. 6.

### 3.3 Structure of Sub Adaptive Neuro Fuzzy Inference Systems

Each ANFIS is a first-order Sugeno type fuzzy inference system with 2 or 3 inputs and 1 output. The internal structures of the ANFIS operators are identical to each other. Each input has 3 generalized bell type membership functions and the output has a linear membership function. Let $X_1, X_2$ denote the inputs of the NF subdetector and $Y$ denote its output. Each possible combination between inputs and their membership functions is represented by a rule in the rule base of the ANFIS operator. So, the rule base contains a total of $9 (3^2)$ rules, which are as follows:
1. If \((x_1, x_2)\) and \((y_1, y_2)\) then \(R_1 = F_1(x_1, x_2)\)
2. If \((z_1)\) and \((z_2)\) then \(R_2 = F_2(z_1, z_2)\)
3. If \((u_1, u_2)\) and \((u_3, u_4)\) then \(R_3 = F_3(u_1, u_2)\)

In the above rules, \(M_{ij}\) represents the jth membership function of ith input, \(R_k\) represents Kth rule and \(F_k\) represents kth output of membership function. \(M_{ij}\) is calculated by:

\[
M_{ij}(u) = \frac{1}{1 + \frac{(u-a_{ij})^2}{b_{ij}}} \\
\]

where \(a, b\) and \(c\) are coefficients that determine the range of the membership function and its value will be determined based on the training data. According to the number of inputs at second layer and the number of associated membership functions, the rule base contains 27 rules. Thus, in the first method, each part has totally 45 rules that should be decided. If this operator was not designed hierarchical then the number of rules would be 243 and training time would be increased exponentially.

In the second method, the numbers of rules are 9 for first and 27 for second layer and totally are 36 that are so less than the number of rules in ordinary neuro fuzzy system with 81 rules.

3.4. The postprocessor
The outputs of all Hierarchical Neuro Fuzzy Systems are sent to the postprocessor and this part represents final decision based on average of them. The outputs of each part are in the interval \([0, 1]\). If the average is greater than the threshold 0.5, the output of postprocessor will be one and zero otherwise that respectively it is interpreted pixel x is edge or not.

3.5. Training Adaptive Neuro Fuzzy Inference System
The internal parameters of each sub neuro-fuzzy inference system must be tuned by train phase. Each sub system is trained individually. An arbitrary image such as Fig. 7a is selected and edge pixels as desired output will be extracted by canny edge detector. The desired output is actually Fig. 7b. Fig. 7c shows the selected image that is corrupted by impulse noise. The noisy image is used as train image in order to train sub neuro fuzzy inference systems. Thus, Fig. 7b and Fig. 7c are considered as desired output and input image, respectively. Each time the output of sub system is compared with desired output and required changes are applied on internal parameters based on Levenberg–Marquardt optimization algorithm [6]. Since sub neuro fuzzy inference systems are hierarchical, neuro fuzzy system at first layer should be trained and their results are used in order to train neuro fuzzy system at second layer [8].
Experimental results

The proposed methods are implemented and used to a number of popular test images such as Pepper, Pirate, airplane and Baboon which are shown in Fig. 8. All of test image are gray level and with dimension 256*256.

First row of Table 1 all includes test images that are corrupted by 10% impulse noise. Next row of table are the results of Sobel edge operator [4] and Third row of table are the results of Canny edge detector [3], both are the default parameters of the MATLAB. The results of Yuksel [17] edge detector with 8 sub neuro fuzzy system are shown in fourth row of table. The results of our proposed methods are shown in the next rows.

As seen in the Table 1, the results of Sobel and Canny edge detector are very weak and the great deal of noise is shown as edge. Yuksel method [17] has good performance and extract edges successfully. The results show that this method robust against the noise. But its efficiency in Baboon image isn’t suitable.
Table 1: Results of our proposed methods, Yuksel method, Sobel and Canny operator

<table>
<thead>
<tr>
<th>Test images with noise 10%</th>
<th>Results of Sobel operator</th>
<th>Results of Canny operator</th>
<th>Results of Yuksel[17] with 8 NF</th>
<th>Results of first method</th>
<th>Results of second method</th>
</tr>
</thead>
</table>


Table 2: Results of our propose methods, Yuksel method, Sobel and Canny operator

<table>
<thead>
<tr>
<th>Test images with noise 20%</th>
<th>Results of Sobel operator</th>
<th>Results of Canny operator</th>
<th>Results of Yuksel [17] with 8 NF</th>
<th>Results of first method</th>
<th>Results of second method</th>
</tr>
</thead>
</table>
In Table 2 the same test images are corrupted by 20% impulse noise. By increasing the measure of noise, the results of Yuksel’s method are weaker than previous. The results of our first proposed method in Table 1 and 2 show that this method has suitable performance and removes noise more accurate than Yuksel’s method but some edge pixels are ignored in results by this method and the results of Yuksel’s method are better than our first method totally. At our first method, farther neighboring pixels and closer neighboring pixels of central pixel have same priority and they are applied together by ANFIS at first layer.

The results of our second method are shown in row of Table 1 and 2. The results show that this method in addition to robust against the noise can well preserve details of image, too. The reason of improving results is that using farther neighboring pixels in ANFIS at first layer is caused impulse noise is removed truly and applying outputs of first layer with closer neighboring pixels helps to detect edge pixels accurately.

5 Conclusion

In this paper, two new methods are represented based on Hierarchical Adaptive Neuro Fuzzy Inference Systems (HANFIS) to create edge detection in noisy images. The edges pixels are extracted by proposed methods directly without requiring the filtering of the noise. Hierarchical structure of proposed operators caused our methods to have more flexibility and more performance than similar methods and provides the possibility of using more neighboring pixels. Another advantage of these methods is that they don’t require parameter setting and can be taught by each arbitrary image. The results show that these operators remove noise with high accuracy and are well able to extract edges. The results of second method is better than those of first, because the neighboring pixels are divided into two groups farther neighboring and closer neighboring pixels and the second has more effect on determination of edge pixels.

References

   http://dx.doi.org/10.1109/TPAMI.2010.161

   http://dx.doi.org/10.1109/TSMCC.2002.804448

   http://dx.doi.org/10.1109/TPAMI.1986.4767851


   http://dx.doi.org/10.1016/j.fss.2008.02.022


http://dx.doi.org/10.1016/S0165-0114(02)00517-1

http://dx.doi.org/10.1109/TCE.2006.1649674

http://dx.doi.org/10.1007/s10916-011-9747-5


http://dx.doi.org/10.1080/00207179108934205

http://dx.doi.org/10.1007/s11042-012-1132-y


http://dx.doi.org/10.1007/978-3-642-24728-6_51

http://dx.doi.org/10.1007/s11045-011-0164-1