A FUZZY DIFFERENCE BASED EDGE DETECTOR

M. A. NIKOUEI MAHANI, M. KOOHI MOGHADAM AND H. NEZAMABADI-POUR

Abstract. In this paper, a new algorithm for edge detection based on fuzzy concept is suggested. The proposed approach defines dynamic membership functions for different groups of pixels in a 3 by 3 neighborhood of the central pixel. Then, fuzzy distance and α-cut theory are applied to detect the edge map by following a simple heuristic thresholding rule to produce a thin edge image. A large number of experiments are employed to confirm the robustness of the proposed algorithm. In the experiments different cases such as normal images, images corrupted by Gaussian noise, and uneven lightening images are involved. The results obtained are compared with some famous algorithms such as Canny and Sobel operators, a competitive fuzzy edge detector, and a statistical based edge detector. The visual and quantitative comparisons show the effectiveness of the proposed algorithm even for those images that were corrupted by strong noise.

1. Introduction

Nowadays, with the progress of science and technology, special machines are designed to perform some advanced actions such as seeing, hearing, and so on. Although the quality of tasks done by machines is much lower than the quality of human abilities; researchers hope to make an advanced machine with the abilities of human. Concerning this issue in recent years, machine learning and machine vision became the hottest research topics in computer science. Machine vision has a wide range of applications such as robot navigation, intelligence transport systems, tracking, image inspection, face detection, gesture, action recognition.[6, 10, 12, 13, 25]. One of the fundamental features in image processing is Edge, because it carries important information and characterizes boundaries of objects. Hence, edge detection is a primary step in many applications of machine vision. It is notable that also for human; edges are important features in detecting and recognizing the objects. Thus, edge detection is one of the main preprocessing steps in human visual system (HVS)[41]. The complication of designing an edge detector is because of the difficulty of defining an edge precisely [28]. Simply, all sharp changes or fast variations in the brightness of images are called edges. The output of edge detection process is an image that reduces the amount of data and eliminates useless information, therefore working with this image is very facilitative and widely applied in feature extraction, image segmentation, texture analysis and so on. There are

Received: March 2010; Revised: November 2010 and April 2011; Accepted: November 2011

Key words and phrases: Edge detection, Fuzzy edge detection, Dynamic membership function, Fuzzy difference, Noisy images.

several different methods for edge detection. Three types of these famous methods are: -Spatial domain based algorithms -Frequency domain based algorithms -Soft computing based algorithms Sobel, Prewitt and Robert operators are some of the first type methods that employ the property of gradient of the image brightness. Gradient based edge detection (GBED) methods are widely used in this area. M. Basus research is one of the most significant contributions in edge detection[3]. The GBED methods have been analyzed by V. Torre and T. Poggio in another old research [37].Canny approach is another famous spatial domain based algorithm which tries to discover the optimal edge with the first order derivatives of Gaussian filter [30]. Unfortunately, all of the above mentioned operators are somewhat sensitive to noise and may provide unsatisfactory results when the image comprises a complicated texture. Edge detection methods in the frequency domain are able to suppress the noise and detect the more complete edges. In this type, the image is transformed into another domain like Laplacian or wavelet, then the information of the transformed domain is used by the algorithms to extract the edges [19, 34, 27, 9]. Hildreth which has used Laplacian transformation is one of these methods [19]. This algorithm has a low sensitivity to noise because it uses Gaussian filter in the earlier steps to reduce the noise [34]. Wavelet transform based edge detection algorithms are also used more widely, because they can provide good localization in both spatial and spectral domains [19, 43]. For example, Ducottet et al. devised a new method for multi-scale detection and characterization of edges based on wavelet transform [9] and Yi et al. proposed a new method based on shearlet framework which has a great capacity in devising alternative improved edge analysis and detection schemes [42]. In the last few years, soft computing based edge detection algorithms have been used more prevalent. Neural network based edge detector[36], fuzzy neural network edge detector [8], ant colony edge detectors [28], neuro-fuzzy edge detectors[5, 18], and fuzzy edge detectors[14, 15, 39] are the famous methods of this type. Actually, fuzzy edge detection methods have attracted the attention of many researchers because they can undertake the noisy, blurred, and low illumination images [39, 38]. Bezdek edge detector [4] is known as one of the most important algorithms in the field of fuzzy logic, which is a combination of the GBED method with a Takagi-Sugeno fuzzy system. Morillas proposed a fuzzy method to reduce the mixed Gaussian impulse noise from images [26]. Edge extraction based on Fuzzy Inference Ruled by Else-action (FIRE) operators is another fuzzy approach which was proposed by Russo in 1994 [33]. His approach can be applied in both noisy and noiseless images. Before applying fuzzy processing to an image, it is necessary to map the original image into the fuzzy domain. Most of these methods have used fuzzy rule base to detect the edges. To fuzzify the intensity values of image pixels, a 33 mask has been applied to scan all pixels in [15]. In this method, the neighborhood pixels of the current pixel (central pixel) are divided to some groups and for each group a membership function is defined. Then, the fuzzy rule base and fuzzy reasoning have been used to process the central pixel based on these membership functions verifying whether the pixel is an edge point or not. Edge detection of noisy images is one of the challenges which has been addressed by many researchers. For instance, a statistical method is proposed

by Lim [16] which has a good performance in images corrupted by Gaussian noise. Unfortunately, this method suffers from a high computational complexity; also it has two parameters which are problem dependent. This paper presents a new simple method based on fuzzy concept which has a low computational complexity. Besides, the proposed method has no training or tuning steps which makes it easy to use in practice. Moreover, it has a good performance in noisy images compared with other methods. Similar to the method proposed by Liang and Looney in [15], our method uses a 33 mask but the difference is that in our proposed algorithm any rule is not used in inference stage. Since the use of rule base gives rise to a high complexity computation in inference step, our proposed approach suggests dynamic membership functions for different groups of pixels. The proposed method applies fuzzy distance and -cut theory to detect the edges as an alternative to rule base approaches. This method is robust against noise and also it has lower complexity in contrast with other fuzzy or statistical methods. The rest of the paper is organized as follows. The fuzzy difference is described shortly in section 2. The proposed method is explained in details in section 3. Section 4 discusses experiments and results obtained by the proposed method and compares it with other methods. Finally the paper is concluded in section 5.

2. **Fuzzy Numbers and Their Distance**

This section focuses on a general method for calculating the distance between two fuzzy numbers. If X is a collection of objects denoted generically by x, then a fuzzy set A in X is define as a set of ordered pairs: $A = \{(x, \mu_A(x))|x \in X\}$ where $\mu_A(x)$ is called the membership function for fuzzy set A and a fuzzy number A is a set in the real line that satisfies the condition for normality and convexity [11]. In other words, A fuzzy set A on R is a fuzzy number if it satisfies some specific conditions as mentioned in [31, 1, 20, 29]. The membership functions which are introduced by bell, triangular and Gaussian functions are some examples of well-known fuzzy numbers. There are many similarity measures to calculate the distance between two fuzzy numbers, for example we can refer to the measures which have been proposed by [7, 35]. A similarity measure for calculating distance between two fuzzy numbers based on-cut theory is shown by (1). This measure has been used frequently for fuzzy based regression application [32, 40].

$$d_{total} = \left[\int_0^1 g(\alpha) d^2(A_\alpha, B_\alpha) d\alpha\right]^{1/2} \tag{1}$$

where

$$d^2(A_\alpha, B_\alpha) = (a_2(\alpha) - a_1(\alpha))^2 + (b_2(\alpha) - b_1(\alpha))^2$$

$$A_\alpha = [a_1(\alpha), b_1(\alpha)], B_\alpha = [a_2(\alpha), b_2(\alpha)]$$

where A and B are two fuzzy numbers and d_{total} shows the distance of the α level between them [40]. $g(\alpha)$ is an increasing function over $[0, 1]$ with $g(0) = 0$, where can be chosen according to the application. $d_{total}(A, B)$ is the distance of

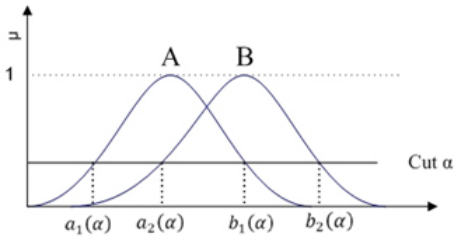


FIGURE 1. Two Typical Fuzzy Numbers A and B. Crossing Cut α and Fuzzy Numbers A and B Produce the Values $a_1(\alpha), b_1(\alpha)$, and $a_2(\alpha), b_2(\alpha)$, Respectively

the α level set of fuzzy numbers A and B [40]. It reflects the near and overlap degree between A and B. $a_1(\alpha)$ and $b_1(\alpha)$ are the cross points between A and α-level set, also $a_2(\alpha)$ and $b_2(\alpha)$ are cross points between α-level set and B (Figure 1). Proper selection of $g(\alpha)$ is very important because it can affect α-cut based distance. It is an increasing function and can be considered as a tuning function in calculating the distance between two fuzzy numbers. The flexibility of $g(\alpha)$ can be used for handling the noisy images as well as tackling the problem of non-uniform illumination in the images. It is also notable that α-cut based distance can be applied on non-concurrent membership functions as well as on two different types of membership functions. In the proposed method according to the experiments, $g(\alpha) = \alpha$ is considered as the increasing function. One can use the similarity measure presented in (1) to calculate the distance between two fuzzy numbers. Here, the aim is to apply the above measure for those fuzzy numbers that are described by bell, triangular and Gaussian membership functions. We follow the computation d^2 for these fuzzy numbers in the appendix A.

3. The Proposed Fuzzy Difference Based Edge Detector (FDBED)

3.1. **Basic Idea.** In conventional gradient based edge detection method to evaluate whether a pixel is an edge point or not, the gray-level transitions between its neighbor pixels are computed according to a criterion simulating the discrete gradient. If the transition (distance) value is greater than a threshold, it means that the pixel is an edge pixel, otherwise it does not. The gradient based edge detection methods are very attractive concerning low computational and implementation complexities while they are sensitive to noise. Here, attempt has been made to capture the basic idea of gradient based methods where they measure the difference between the intensity values of the pixels, and we extend it using fuzzy logic concept to achieve a more precise edge detector method. Therefore, the proposed method is a "fuzzy difference based edge detector" which is abbreviated as FDBED through the rest of the article. In FDBED, fuzzy concept is used to determine resemblance and dissemblance between pixels, but the form of using fuzzy concept in this paper is different from the reported methods so far. Unlike other methods such as competitive fuzzy edge detector (CFED) [15] that uses rule-base and predefined

P_1	P_2	P_3
P_4	P_0	P_5
P_6	P_7	P_8

FIGURE 2. The 8-neighbors of the Central Pixel P_0 in a 3*3 Mask

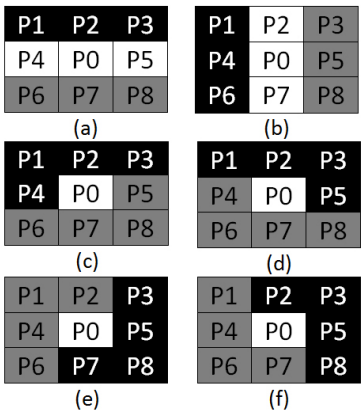


FIGURE 3. The Proposed Six T_is, Black Stands for Class A_i and Gray for Class B_i
a) $T_1 = Partition1 = \{A_1 = (P_1, P_2, P_3), B_1 = (P_6, P_7, P_8)\}$
b) $T_2 = Partition2 = \{A_2 = (P_1, P_4, P_6), B_2 = (P_3, P_5, P_8)\}$
c) $T_3 = Partition3 = \{A_3 = (P_1, P_2, P_3, P_4), B_3 = (P_5, P_6, P_7, P_8)\}$
d) $T_4 = Partition4 = \{A_4 = (P_1, P_2, P_3, P_5), B_4 = (P_4, P_6, P_7, P_8)\}$
e) $T_5 = Partition5 = \{A_5 = (P_1, P_2, P_4, P_6), B_5 = (P_3, P_5, P_7, P_8)\}$
f) $T_6 = Partition6 = \{A_6 = (P_1, P_4, P_6, P_7), B_6 = (P_2, P_3, P_5, P_8)\}$

membership functions, the FDBED uses dynamic membership functions and the α-cut based fuzzy distance (introduced in section 2) to detect the edge pixels.

3.2. **Neighboring Definition.** Let $0 \leq I(i,j) \leq I_{Max}$, $i = 1, 2, 3, \ldots, M; j = 1, 2, 3, \ldots, N$ is the input image. For each pixel, the FDBED considers its eight-neighborhood pixels to compute a value according to the proposed fuzzy criterion that indicates whether the pixel is an edge point or not. For simplicity, central pixel and its 8-neighbors are called $P_0, P_1, , P_8$ respectively as shown in Figure 2. In FDBED to determine the edges, the 8-neighbors of each pixel are partitioned into two classes (A and B) in six different ways. Figure 3 shows the proposed six partitions which are used in the FDBED. The pixels of class A and class B are shown with two different colors (black and gray). For simplicity, the i-th partition is shown as T_i which contains A_i and B_i classes.

3.3. **Construction of Membership Functions.** For partition T_i, two membership functions are defined: one for class A_i and another one for class B_i which

are called as μ_{A_i} and μ_{B_i}, respectively. μ_{A_i} is defined for class A_i such that all pixels of A_i belong to μ_{A_i} with high value; μ_{B_i} will be defined likewise for class B_i. Inspired by gradient-based edge detection methods, the gradient is defined as the difference of two membership functions μ_{A_i} and μ_{B_i} for each T_i. We use the -cut based distance measure introduced in section 2. For each pixel due to considering six T_i, we have six distance values. The maximum value is considered as a measure to present the intensity gradient for under processed pixel and will be used for deciding whether the pixel is an edge point or not. To create membership functions, three types of functions consisting bell, triangular, and Gaussian have been examined. The main issue is how the parameters of the functions should be defined according to the intensity values of the classes A_i and B_i which is explained in the next sub-sections.

3.3.1. *Construction of Bell Function.* Suppose we aim to find the parameters of a bell function for the assumed class A. The bell function has three parameters which should be obtained from pixel intensity of class A. (2) indicates the proposed method for selecting the parameters of the bell function based on its pixels' intensity. Referring to the appendix-A, a, b and c are the parameters of a bell function which are calculated according to class A pixels' intensity (2).

$$\begin{aligned} MaxM &= max\{I(P_x)|P_x \in A\} \\ MinM &= min\{I(P_x)|P_x \in A\} \\ a &= MaxM - MinM \\ c &= mean\{I(P_x)|P_x \in A\} \\ b &= Const \end{aligned} \tag{2}$$

3.3.2. *Construction of Triangular Function.* The proposed method for tuning the parameters of triangular functions based on the pixels' intensity of a certain class A is presented in (3).

$$\begin{aligned} c &= max\{I(p_x)|p_x \in A\} \\ a &= min\{I(p_x)|p_x \in A\} \\ c &= mean\{I(p_x)|p_x \in A\} \end{aligned} \tag{3}$$

where a, b and c are the parameters of a triangular membership function (appendix-A).

3.3.3. *Construction of Gaussian function.* In a similar way, the parameters of a Gaussian membership function are calculated according to (4).

$$\begin{aligned} c &= meanI(p_x)p_x \in A \\ \delta &= \frac{1}{|A|}\left[\sum_{p_x \in A}(I(p_x)-c)^2\right]^{\frac{1}{2}} \end{aligned} \tag{4}$$

3.4. **Edge Map Calculation.** Using the proposed equations (2-4), one can define two dynamic membership functions and fit them on the classes A_i and B_i of partition T_i. These membership functions express how much the intensity of the class members vary through the class. This fitting will be done for all six classes introduced in Figure 3. According to the edge definition, a central pixel of a 3 by 3 mask is an edge point if a big change in the intensity of its neighbors is observed.

It means that at least in one of the six partitions, the intensity of classes A_i and B_i must have a high difference. Here, this idea is employed to determine the edge points. Assume μ_{A_i} and μ_{B_i} are the membership functions created for classes A_i and B_i, respectively. The difference of these membership functions is calculated using the -cut based distance measure described in section 2. The difference between the two membership functions of partition T_i is denoted by d_{T_i}. The difference is calculated for all six partitions and the maximum difference value is considered as the final difference for central pixel. This final difference is denoted by d_f (5). Therefore, the image is scanned by a 3 by 3 mask and d_f is calculated for all pixels to give the edge map image.

$$d_f = max\{d_{T1}, d_{T2}, ..., d_{T6}\} \tag{5}$$

Therefore, at the end of this step we have an edge map image EM(i,j) ,i=1,2,3, . . . ,N ; j=1,2,3, . . . ,M where each entry of this image contains the d_f value regarding the central point.

3.5. **Thresholding and Thinning.** The final goal of edge detection is a binary image indicating the presence of edge. Therefore, we have included a global threshold at the output of our algorithm to discard irrelevant edges. Tuning the parameters is one the difficult problems for unprofessional users that aim to use a method. This problem has been solved in the FDBED by considering a constant value as global threshold. The FDBED constructs the membership functions dynamically and locally according to statistical properties of the illumination of the pixels; it allows FDBED to find the edges for a wide range of images only with a constant threshold. Rely on the experiments to achieve the output binary edge image, in this method a global threshold is suggested for all situations to compare FDBED with the other methods. Albeit the output binary image should be thinned, often the thinning process is done by a morphological thinning algorithm [30] or something else like that. To avoid any post thinning process, a heuristic thinning approach is suggested. It applies on edge map image as a rule integrated with thresholding rule. This heuristic method selects the edge points which has more edgy value than the threshold and also have maximum edgy value comparing with their neighbors at least in one direction. Therefore, the simple rule of the edge detection on edge map redefined as Eq.6. To realize this rule, the edge map image, EM(.) is scanned by a 3 by 3 mask and in each position the central pixels are classified to edge or non-edge classes. The positions of pixels within the mask are considered as the mask shown by Figure 2.

$$\begin{gathered} IF[EM(P_0)]AND[\{(EM(P_0)EM(P_2))AND(EM(P_0)EM(P_7))\} \\ OR(EM(P_0)EM(P_4))AND(EM(P_0)EM(P_5))OR(EM(P_0)EM(P_3)) \\ AND(EM(P_0)EM(P_6))OR(EM(P_0)EM(P_1))AND(EM(P_0)EM(P_8))] \\ THENP_0 \text{ is an edge point } (EI(P_0) = 1) \text{ otherwise } EI(P_0) = 0 \end{gathered} \tag{6}$$

where EI presents the final binary edge image and thr is the threshold value. This heuristic rule allows the points with edge map value to be selected as edge points by

For each pixel i of the input image,
  Considering a 3*3 mask around the pixel i,
  Assume six partitions (T_1 ...T_6) in each mask,
  For each partition (T_i)
    Suppose Two classes of neighboring pixels, classes A_i and B_i,
    Construct two membership functions, μ_{A_i} and μ_{B_i}, according to statistical information of classes A_i and B_i.
    Calculate $d_{total}(A_i, B_i)$, d_{Ti}, Using Eq.1.
  Calculate for each pixel i, $d_f = \max\{d_{T1}, d_{T2}, \ldots, d_{T6}\}$, d_f is considered as the edgy value of the pixel i in the edge map image to provide image EM(i).
For each pixel i of the EM image,
Classify the pixel as edge pixel or non-edge pixel, according to rule given by Eq.6.

FIGURE 4. The Pseudo Code of the Proposed FDBED Algorithm

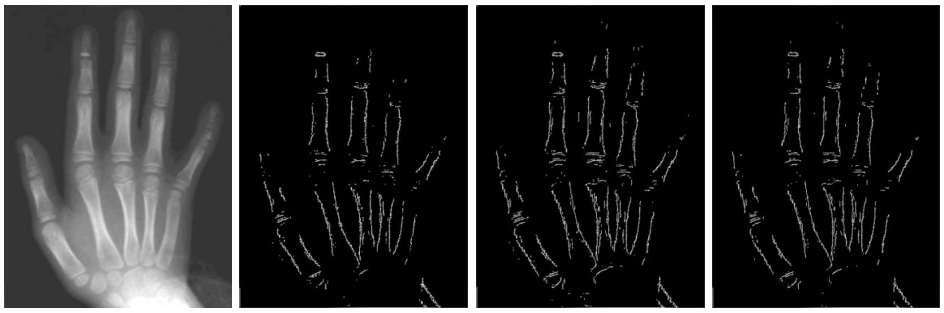
choosing a low threshold. It also selects only the highest edgy point in continuous edgy region and due to this property a thin edge image is detected. The pseudo code of the FDBED is given by Figure 4.

4. **Experimental Results**

In this section, we experimentally study the effectiveness of the proposed FDBED. The proposed edge detector implemented in C++ language and its performance is tested in three cases including normal, noisy Gaussian corrupted and uneven lightening images. The FDBED is only part of the problem in edge detection. Most of edge detection algorithms use a smoothing method to improve edge detection performance in noisy images. Hence, in FDBED a simple averaging filter is used as a preprocessing stage before main edge detector process in noisy images. The rest of this section is organized in two subsections: visual experiments and quantitative comparative study. In the visual experiments, the result images of the FDBED in different situation are illustrated. Comparative study subsection illustrates the numeric comparison between FDBED and other methods using Baddeleys Delta Metric (BDM) [2].

4.1. **Visual Experiments.** In spite of the fact that each type of the membership function can be effective for a specific situation, the experimental studies reveal that Gaussian function typically performs better than bell and triangular functions. Figure 5 illustrates the result of FDBED for triangular, bell and Gaussian membership functions in which the threshold value is set to 0.3. The next three experiments illustrate the FDBED performance on obtaining the edge map in normal, noisy and uneven lightening image respectively, and in the next step, the heuristic rule is used to convert the edge map to a binary image.

Figure 6 illustrates the result of FDBED in normal situation on Lena image. It is apparent that the edge map is a worth demonstration of the original image which highlights the real edge points. Figure 7 shows the results of FDBED for a noisy image which is corrupted by a Gaussian noise with zero mean and variance 25. It is obvious from Figure 6, FDBED can obtain nice edge map for noisy images. Figure 8 illustrates FDBED performance on varied illumination (uneven lightening) images. Results illustrate that FDBED has an adequate performance for this type of images, since in FDBED membership functions are constructed locally and dynamic.



(a) Original image (b) Triangular (c) Bell (d) Gaussian

FIGURE 5. The Edge Image Provided by the Proposed FDBED Using Different Type of Membership Functions with Image 430*580 Pixels Images

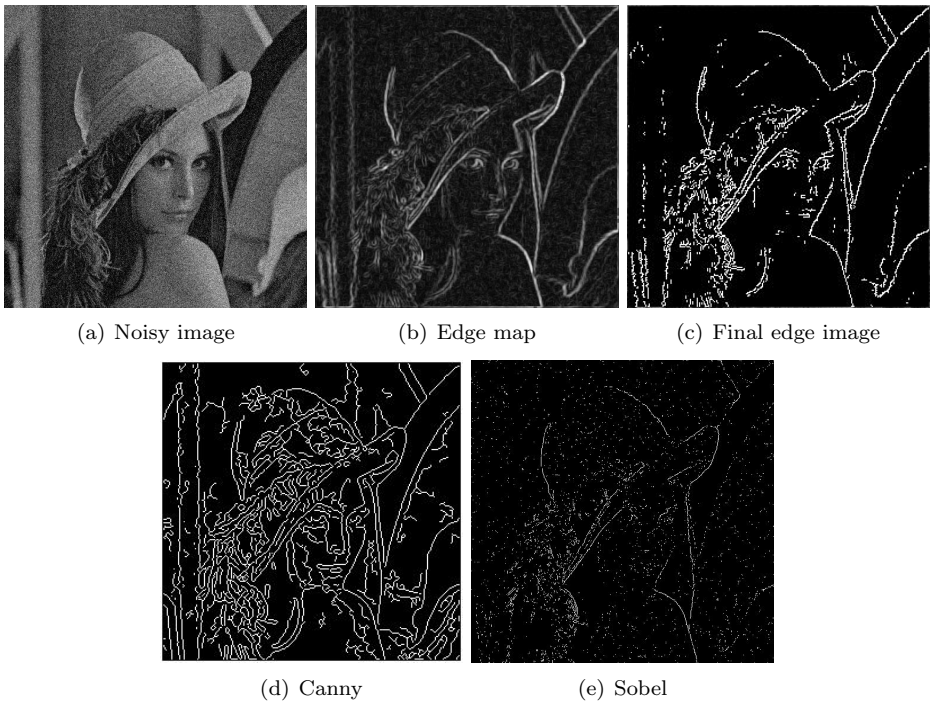


(a) Original image (Lena) (b) Edge map (c) Final edge image

FIGURE 6. a) Original Image (Lena) with 512*512 Pixels, b) The Edge Map of (a) Obtained by FBDED Using Gaussian Membership Function, c) The Final Edge Image of (a) Obtained by the Proposed Heuristic Rule, and the Threshold is set to 0.3.

4.2. **Comparative Study.** In this subsection, the FDBED is experimentally compared with other methods. To compare FDBED with the other methods, Baddeleys Delta Metric (BDM)[38] is used as a numeric metric. Due to the fact that BDM calculates the distance between two binary images, it is applied to binary edge image. The experiments were done on various types of Berkeley Segmentation Data Set (BSDS) images. Up to 11 images with various scenes and texture including face, human, tree, airplane, sea-sky, zebra, garden, building, island and Panda were used in our experiments.

4.2.1. *Baddeley's Delta Metric.* Baddeleys Delta Metric (BDM) [2] is a measure which intends to estimate the dissimilarity (distance) between two binary images. Therefore it can be applied after the last step where the images are binarized and



(a) Noisy image (b) Edge map (c) Final edge image

(d) Canny (e) Sobel

FIGURE 7. a) The Noisy Image Corrupted by Gaussian Noise with Zero Mean and Variance 25, b) The Edge Map Obtained by FBDED Using Gaussian Membership Function After Smoothing. c) The Final Edge Image of (a) Obtained by the Proposed Heuristic Rule, and the Threshold is Set to 0.3. d) The Edge Image Obtained by Canny. e) The Edge Image Obtained by Sobel

the image pixels get only two values zero or one which demonstrates the edge or non-edge pixels. Assume I_1 and I_2 are two binary images with the same dimension. $P = 1, ..., M1, , N$ is defined as the set of position where the image dimension is MN. Distance function, $d(p, I_i)$, is defined to illustrate the minimum distance between position $p \in P$ and edge pixel set of I_i. That is to say that Euclidean distance is used in our experiments like Lopez [17]. and many other researchers. In general case of k-BDM, k is a value between 0 and ∞ $(0 < k < \infty)$, BDM measure is defined by (7).

$$\Delta^k(I_1, I_2) = \frac{\frac{1}{|P|}\sum_{p\in P} |w(d(p, I_1)) - w(d(p, I_2))|^k}{k} \tag{7}$$

where w is a concave and increasing weighting function. For all experiments which are done in our experiments, we select k=2 and $w(x) = min(c, x)$ where $c = \sqrt{(M^2 + N^2)}$ similar to Lopez [17] and also references [22, 23, 21, 24]. We



(a) Uneven lightening image (b) Edge map (c) Final edge image



(d) Canny (e) Sobel

FIGURE 8. a) An Original Uneven Lightening Image with 512*512 pixels, b) Its Edge Map Obtained by FBDED Using Gaussian Membership Functions, c) Its Final Corresponding Result Obtained by the Proposed Heuristic Rule, and the Threshold is Set to 0.3. d) The Edge Image Obtained by Canny e) The Edge Image Obtained by Sobel.

used the program provided by Lopez [17] for calculating BDM in our experiments. The comparative study is done using BDM and the results are come in the next subsection.

4.2.2. *Results.* To numerically and quantitatively compare the FDBED with other method we applied our FDBED along with Canny and Sobel operators, a competitive fuzzy edge detector (CFED)[15] and a statistical based edge detector [16] that is based on the robust rank-order (RRO) test, to 11 images from BSDS and the results are reported with regard to the BDM. The experiments are expressed in three parts, normal images, uneven lightening images and noisy images. Table 1 illustrates the comparison of FDBED with the others in normal images. In the experiments we carried out the results of Canny approach for four threshold values: 2.25, 1.5, 0.75 and auto selection. Also, CFED and RRO are implemented as reported in [15, 16]. The results of FDBED are provided by five threshold values: 0.3, 0.4, 0.5, 0.6 and 0.7.

Image No in BSDS	CFED	Canny 1.5	Canny 2.25	Canny 0.75	Canny Auto	Sobel Auto	FDBED					RRO
							0.3	0.4	0.5	0.6	0.7	
69015	2.32	1.32	1.35	1.39	2.17	1.51	1.741	1.436	1.279	1.188	1.141	1.35
37073	1.58	1.03	1.02	1.06	1.83	1.18	1.291	1.159	1.044	0.964	0.909	1
46076	1.43	0.95	0.94	0.96	1.63	1.11	1.289	1.105	0.988	0.922	0.871	0.94
54082	2.1	1.03	1.03	1	1.76	1.3	1.554	1.275	1.086	0.947	0.864	0.99
102061	2.17	0.92	0.89	1	1.69	1.22	1.493	1.324	1.208	1.111	1.024	0.9
143090	1.45	0.89	0.9	0.89	1.53	1.1	1.060	0.918	0.839	0.788	0.736	0.89
147091	2.01	1	0.99	1.07	1.63	1.2	1.589	1.428	1.304	1.193	1.095	0.96
181079	2.22	1.26	1.24	1.37	2.03	1.53	1.915	1.727	1.574	1.455	1.367	1.15
189080	2.13	1.25	1.25	1.26	2.04	1.44	1.333	1.194	1.121	1.080	1.054	1.32
253027	2.54	1.26	1.18	1.47	2.01	1.52	1.815	1.716	1.646	1.585	1.527	1.27
86016	3.65	1.39	1.32	1.49	2.61	1.73	3.460	2.767	2.111	1.618	1.306	1.17
Sum	23.6	12.3	12.11	12.96	20.93	14.84	18.54	16.049	14.2	12.851	11.894	11.94

TABLE 1. The Comparison of FDBED with Others Algorithms in Normal Images Using BDM

Image NO in BSDS	CFED	Canny 1.5	Canny 2.25	Canny 0.75	Canny Auto	Sobel Auto	FDBED					RRO
							0.3	0.4	0.5	0.6	0.7	
69015	3.17	1.33	1.32	1.38	2.24	1.55	1.433	1.297	1.216	1.172	1.144	1.44
37073	1.93	1.01	1	1.02	1.56	1.25	1.122	1.036	0.965	0.915	0.879	1.02
46076	2	1	0.99	1	1.71	1.18	0.809	0.737	0.696	0.673	0.654	0.98
54082	2.44	1.05	1.04	1.1	1.9	1.37	1.392	1.167	1.034	0.947	0.890	0.99
102061	2.24	0.96	0.93	1.08	1.6	1.2	1.369	1.187	1.040	0.923	0.824	0.93
143090	1.66	0.92	0.93	0.95	1.58	1.15	0.983	0.883	0.824	0.784	0.761	0.89
147091	1.66	0.92	0.93	0.95	1.58	1.15	1.466	1.280	1.158	1.070	0.993	0.89
181079	2.93	1.28	1.24	1.43	2.17	1.53	1.674	1.552	1.444	1.352	1.278	1.3
189080	2.53	1.3	1.31	1.27	2.05	1.42	1.235	1.145	1.092	1.066	1.041	1.37
253027	2.5	1.27	1.16	1.53	1.98	1.51	1.754	1.626	1.531	1.440	1.339	1.32
86016	5	1.23	1.21	1.72	2.87	1.59	2.520	1.893	1.463	1.211	1.096	1.2
Sum	28.06	12.27	12.06	13.43	21.24	14.9	15.75	13.80	12.46	11.55	10.89	12.33

TABLE 2. The Comparison of FDBED with Others Algorithms in Uneven Lightening Condition Using BDM

According to Table.1, it is obvious that FDBED performs better in contrast to the CFED, and Sobel, and comparable in contrast to outstanding Canny approach and RRO. The next experiment is done to evaluate the FDBED performance in uneven lightening images. In this case, we carry out the experiment with the same BSDS images which are corrupted by uneven lightening effect. Table. 2 shows the performance of the FDBED in uneven lightening images.

Table 2 shows that FDBED provides better performance than Sobel and CFED methods and comparable somewhat better than Canny and RRO with regard to the BDM criterion in uneven lightening condition, since it is able to offer the lower BDM in contrast to the other methods. Finally, FDBED is tested on noisy images. The eleven selected images of BSDS are corrupted by Gaussian noise with zero mean and three different variances. The images are smoothed using a simple average filter, and delivered to the comparative methods for edge detection. The results of BMD criterion for all comparative edge detector methods for noisy images are given by Table 3.The final experiments illustrate the appealing performance of the

FDBED in contrast to the other edge detectors, again. The results obtained in Table 3 show that FDBED is robust against different noise variances. Tables 1-3 shows that for test images of BSDS the best results are obtained for FDBED with threshold value of 0.7.

5. **Conclusion**

This paper proposes a fuzzy based edge detection algorithm, which uses dynamic membership functions for different groups of pixels in a 3 by 3 neighborhood of the central pixel and applies fuzzy distance and -cut theory to detect the edge map. Based on the experiments, the proposed algorithm in contrast to existing fuzzy based edge detection methods has a good performance for handling the noisy images as well as uneven lightening conditions. The main feature of the FDBED is its robustness in detection and extraction of edges in the noisy corrupted images. Based on our experiments, compared with Sobel and Canny operators, CFED and RRO, the proposed algorithm can suppress the noise and reduce the false edge points; therefore, it is proper to process edge detection with strong noise image in actual works.

6. **Appendix-A**

The Bell-shape Membership Function. A bell-shape membership function can be presented in (8) in which a, b, and cc are the function parameters. Therefore, the crossing points of a specific cut and the fuzzy number$\mu_{A_{Bell}}(x)$ are computed based on(9) in which, it is supposed that b=1.

$$\mu_{A_{Bell}}(x) = (x; a, b, c) = \left[1 + \left|\frac{x-c}{a}\right|^{2b}\right]^{-1} \tag{8}$$

$$a_1(\alpha), b_1(\alpha) = f^{-1}(\alpha) = c \pm a\sqrt{\left(\frac{1}{\alpha} - 1\right)} \tag{9}$$

The Triangular Membership Function. Another type of the membership function which is used is triangular function. It can be shown by (10) in which a, b, and c are the function parameters. We have two points for any α-cut. The crossing points of a specific cut α and the fuzzy number $\mu_{A_{Triangular}}(x)$ are computed based on (11).

$$\mu_{A_{Triangular}}(x) = f(x; a, b, c) = \begin{cases} \frac{x-a}{b-a}, & if a < x < b \\ \frac{c-x}{c-b}, & if b < x < c \\ 0, & otherwise \end{cases} \tag{10}$$

$$a_1(\alpha), b_1(\alpha) = f^{-1}(\alpha) = \begin{cases} \alpha b - \alpha a + a \\ -c\alpha + b\alpha + c \end{cases} \tag{11}$$

Noise	Method		69015	37073	46076	54082	102061	143090	147091	181079	189080	253027	86016	Sum
10	**CFED**		3.17	2.08	2.04	2.29	2.4	1.77	1.77	3.2	2.93	2.8	3.64	28.09
	Canny 1.5		1.32	1.04	1.02	1.05	0.96	0.95	0.95	1.32	1.39	1.25	1.34	12.59
	Canny 2.25		1.31	1.03	1	1.07	0.92	0.94	0.94	1.37	1.33	1.15	1.33	12.39
	Canny 0.75		1.37	1.08	1.08	1.14	1.08	0.97	0.97	1.4	1.45	1.44	1.62	13.6
	Canny Auto		2.38	2.17	2.15	2.19	2.03	2.2	2.2	2.38	2.7	2.15	2.81	25.36
	Sobel Auto		1.47	1.13	1.07	1.17	1.09	1	1	1.43	1.34	1.39	1.36	13.45
	FDBED	**0.3**	1.516	1.216	0.94	1.204	1.273	0.991	1.324	1.693	1.540	1.684	1.789	15.17
		0.4	1.260	0.971	0.813	0.964	1.059	0.869	1.144	1.388	1.435	1.487	1.364	12.754
		0.5	1.139	0.866	0.754	0.854	0.897	0.785	0.994	1.196	1.379	1.363	1.197	11.424
		0.6	1.090	0.832	0.701	0.816	0.763	0.715	0.891	1.074	1.317	1.270	1.067	10.536
		0.7	1.076	0.818	0.624	0.805	0.672	0.676	0.828	0.988	1.142	1.188	1.010	9.827
	RRO		1.41	1.04	1.05	0.99	1.06	0.91	0.91	1.4	1.49	1.48	1.24	12.98
20	**CFED**		3.19	2.06	2.04	2.3	2.37	1.77	1.77	3.16	2.89	2.79	3.58	27.92
	Canny 1.5		1.35	1.04	1.04	1.01	0.98	0.95	0.95	1.34	1.4	1.25	1.36	12.67
	canny 2.25		1.31	1.04	1.04	1.05	0.94	0.96	0.96	1.26	1.33	1.16	1.33	12.38
	Canny 0.75		1.41	1.06	1.09	1.12	1.08	0.99	0.99	1.46	1.45	1.46	1.6	13.71
	Canny Auto		2.39	2.26	2.03	2.2	2.05	2.26	2.26	2.44	2.66	2.12	2.81	25.48
	Sobel Auto		1.46	1.15	1.07	1.18	1.1	1	1	1.43	1.34	1.4	1.34	13.47
	FDBED	**0.3**	1.533	1.254	0.950	1.229	1.255	1.027	1.332	1.707	1.540	1.694	1.770	15.29
		0.4	1.268	0.999	0.817	0.984	1.059	0.883	1.155	1.400	1.433	1.494	1.376	12.86
		0.5	1.142	0.870	0.752	0.863	0.896	0.797	1.010	1.195	1.382	1.356	1.217	11.48
		0.6	1.092	0.834	0.709	0.819	0.754	0.730	0.896	1.064	1.332	1.265	1.088	10.58
		0.7	1.076	0.817	0.634	0.803	0.659	0.678	0.824	0.978	1.180	1.174	1.014	9.83
	RRO		1.42	1.04	1.04	1.01	1.05	0.92	0.92	1.38	1.48	1.47	1.24	12.97
30	**CFED**		3.24	2.07	2.03	2.3	2.4	1.77	1.77	3.19	2.87	2.81	3.49	27.94
	Canny 1.5		1.35	1.02	1.06	1.04	1	0.97	0.97	1.29	1.39	1.26	1.37	12.72
	canny 2.25		1.32	1.03	1.05	1.09	0.93	0.98	0.98	1.26	1.45	1.18	1.33	12.6
	Canny 0.75		1.45	1.07	1.06	1.13	1.07	1.01	1.01	1.51	1.78	1.46	1.45	14
	Canny Auto		2.43	2.28	1.97	2.24	2.09	2.26	2.26	2.39	2.65	2.15	2.78	25.5
	Sobel Auto		1.46	1.13	1.07	1.17	1.09	0.99	0.99	1.43	1.35	1.4	1.33	13.41
	FDBED	**0.3**	1.551	1.274	0.961	1.248	1.260	1.049	1.346	1.736	1.541	1.701	1.736	15.40
		0.4	1.283	1.022	0.819	1.007	1.056	0.895	1.170	1.419	1.432	1.503	1.370	12.97
		0.5	1.154	0.877	0.756	0.873	0.880	0.805	1.012	1.197	1.384	1.358	1.234	11.53
		0.6	1.096	0.834	0.710	0.826	0.734	0.734	0.894	1.057	1.345	1.251	1.124	10.60
		0.7	1.076	0.815	0.647	0.807	0.644	0.687	0.825	0.970	1.220	1.146	1.023	9.86
	RRO		1.42	1.04	1.03	1.01	1.05	0.93	0.93	1.38	1.48	1.46	1.24	12.97

TABLE 3. The Comparison of FDBED with Others Algorithms in Noisy Images with Different Variances Based on BDM

The Gaussian Membership Function. (12) shows the Gaussian membership function, which is another type of function used in the proposed method. In (12), c and σ stand for mean and standard deviation of the Gaussian function. The crossing points of a specific -cut and the fuzzy number$\mu_{A_{Gaussian}}(x)$ are computed based on (13).

$$\mu_{A_{Gaussian}}(x) = e^{\frac{-(x-c)^2}{2\sigma^2}} \tag{12}$$

$$a_1(\alpha), b_1(\alpha) = f^{-1}(\alpha) = c \pm \sigma\sqrt{-2ln(\alpha)} \tag{13}$$

Acknowledgements. The authors would like to thank the IJFS Editorial Board and the anonymous reviewers for their very helpful suggestions. In addition, the authors give kind respect and special thanks to Mr. Carlos Lopez-Molina for giving us his BDM programs.

References

[1] O. AbuAarqob, N. Shawagfeh and O. AbuGhneim, *Functions defined on fuzzy real numbers according to zadehs extension*, International Mathematical Forum, (2008), 763-776.

[2] A. J. Baddeley, *An error metric for binary images*, Robust Computer Vision-Quality of Vision Algorithms, (1992), 59-78.

[3] M. Basu, *Gaussian-based edge-detection methodsa survey*, IEEE Transactions on Systems Man, and CyberneticsPart C, **32(3)** (2002), 252-260.

[4] J. C. Bezdek, R. Chandrasekhar and Y. Attikiouzel, *A geometric approach to edge detection*, IEEE Transaction on Fuzzy Systems, **6(1)** (1998), 52-75.

[5] V. Boskovitz and H. Guterman, *An adaptive neuro-fuzzy system for automatic image segmentation and edge detection*, IEEE Transactions on Fuzzy Systems, **10** (2002), 247-262.

[6] D. Brzakovic and L. Hong, *Road edge detection for mobile robot navigation*, In Proceedings of IEEE International Conference on Robotics and Automation, Scottsdale, AZ , USA, **2** (1989), 1143-1147.

[7] S. M. Chen, *New methods for subjective mental workload assessment and fuzzy risk analysis*, Cybernetics and Systems, **27** (1996), 449-472.

[8] K. David A. and B. James C, *Edge detection using a fuzzy neural network*, Science of Artificial Neural Networks, **1710** (1992), 510-521.

[9] C. Ducottet, T. Fournel and C. Barat, *Scale-adaptive detection and local characterization of edges based on wavelet transform*, Signal Processing, **84(11)** (2004), 2115-2137.

[10] T. Hou and W. Kuo, *A new edge detection method for automatic visual inspection*, The International Journal of Advanced Manufacturing Technology, **13** (1997), 407-412.

[11] J. S. R. Jang, C. T. Sun and E. Mizutani, *Neuro-fuzzy and soft computing- a computational approach to learning and machine intelligence*, Prentice-Hall of India Pvt. Ltd., New Delhi, 2006.

[12] S. Karungaru, M. Fukumi, N. Akamatsu and T. Akashi, *A simple 3D edge template for pose invariant face detection*, Lecture Notes in Computer Science, **4253** (2006), 692-698.

[13] H. Kim, J. Lee, D. Kim, H. Yoon and S. Chi, *Motion and natural hand detection for gesture recognition*, SICE-ICASE,International Joint Conference, Busan,Korea, (2006), 313-316.

[14] T. Law, H. Itoh and H. Seki, *Image filtering, edge detection, and edge tracing using fuzzy reasoning*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **18** (1996), 481-491.

[15] L. R. Liang and C. G. Looney, *Competitive fuzzy edge detection*, Applied Soft Computing, **3** (2003), 123-137.

[16] D. H. Lim, *Robust edge detection in noisy images*, Computational Statistics and Data Analysis, **50** (2006), 803-812.
[17] C. Lopez-Molina, H. Bustince, J. Fernandez, P. Couto and B. De Baets, *A gravitational approach to edge detection based on triangular norms*, Pattern Recognition, **43(11)** (2010), 3730-3741.
[18] S. Lu, Z. Wang and J. Shen, *Neuro-fuzzy synergism to the intelligent system for edge detection and enhancement*, Pattern Recognition, **36** (2003), 2395-2409.
[19] D. Marr and E. Hildreth, *Theory of edge detection*, Proceedings Royal Soc.London, **207** (1980), 187-217.
[20] A. Maturo, *On some structures of fuzzy numbers*, Iranian Journal of Fuzzy Systems, **6** (2009), 49-59.
[21] R. Medina-Carnicer, A. Carmona-Poyato, R. Muoz-Salinas and F. J. Madrid-Cuevas, *Determining hysteresis thresholds for edge detection by combiningthe advantages and disadvantages of thresholding methods*, IEEE Transactionson Image Processing, **19(1)** (2010), 165-173.
[22] R. Medina-Carnicer, F. Madrid-Cuevas, A. Carmona-Poyato and R. M. noz Salinas, *On candidates selection for hysteresis thresholds in edge detection*, PatternRecognition, **42(7)** (2009), 1284-1296.
[23] R. Medina-Carnicer and F. Madrid-Cuevas, *Unimodal thresholding for edgeDetection*, Pattern Recognition, **41(7)** (2008), 2337-2346.
[24] R. Medina-Carnicer, F. Madrid-Cuevas, R. Muoz-Salinas and A. Carmona-Poyato, *Solving the process of hysteresis without determining the optimal thresholds*, Pattern Recognition, **43(4)** (2010), 1224-1232.
[25] H. Meng, M. Freeman, N. Pears and C. Bailey, *Real-time human action recognition on an embedded, reconfigurable video processing architecture*, Journal of Real-Time Image Processing, **3** (2008), 163-176.
[26] S. Morillas, V. Gregori and Antonio. Hervs, *Fuzzy peer groups for reducing mixed gaussian-impulse noise from color images*, IEEE Transaction on Image Processing, **18(7)** (2009), 1452-1466.
[27] J. Musevi-Niya and A. Aghagolzadeh, *Adaptive directional wavelet-based edge detection*, 2nd International Symposium on Telecommunications (IST2003), Isfahan, Iran, (2003), 191-195.
[28] H. Nezamabadi-pour, S. Saryazdi and E. Rashedi, *Edge detection using ant algorithms*, Soft Computing, **10** (2005), 623-628.
[29] E. Pasha, A. Saiedifar and B. Asady, *The percentiles of fuzzy numbers and their applications*, Iranian Journal of Fuzzy Systems, **6** (2009), 27-44.
[30] W. K. Pratt, *Digital image processing*, John Wiley and Sons, 2001.
[31] G. Roy Jun and W. Voxman, *Topological properties of fuzzy numbers*, Fuzzy Sets and Systems, **10(1-3)** (1983), 87-99.
[32] X. Ruoning, *A linear regression model in fuzzy environment*, Adv. Modelling Simulation, **27** (1991), 31-40.
[33] F. Russo and G. Ramponi, *Edge extraction by FIRE operators* , in IEEE World Congress on Computational Intelligence, **1** (1994), 249-253.
[34] J. I. Siddigue and K. E.Barner, *Wavelet-based multi-resolution edge detection utilizing gray level edge maps*, International Conference on Image Processing (ICIP 98), (1998), 550-554.
[35] B. Sridevi and R. Nadarajan, *Fuzzy similarity measure for generalized fuzzy numbers*, International Journal of Open Problems in Computer Science and Mathematics, **2** (2009), 240-253.
[36] P. Terry and D. Vu, *Edge detection using neural networks*, In IEEE Proceedings of 27th Asilomar Conference on Signals, Systems and Computers, (1993), 391-395.
[37] V. Torre and T. Poggio, *On edge detection*, Massachusetts Institute of Technology-Artificial Intelligence Laboratory, 1984.
[38] D. Van De Ville, M. Nachtegael, D. Van Der Weken, E. Kerre, W. Philips and I. Lemahieu, *Noise reduction by fuzzy image filtering*, IEEE Transactions on Fuzzy Systems, **11** (2003), 429-436.
[39] J. Wu, Z. Yin and Y. Xiong, *The fast multilevel fuzzy edge detection of blurry images*, IEEE Signal Processing Letters, **14** (2007), 344-347.

[40] R. Xu and C. Li, *Multidimensional least-squares fitting with a fuzzy model*, Fuzzy Sets and Systems, **119** (2001), 215-223.
[41] F. Yang, S. Wan and Y. Chang, *Improved method for gradient-threshold edge detector based on HVS*, Lecture Notes in Computer Science, **3801** (2005), 1051-1056.
[42] S. Yi, D. Labate, G. R. Easley and H. Krim, *A shearlet approach to edge analysis and detection*, Trans. Image Proc,**18(15)** (2009), 1057-7149.
[43] X. Zong and W. Liu, *Fuzzy edge detection based on wavelets transform*, Machine Learning and Cybernetics, International Conference, (2008), 2869-2873.

M. A. Nikouei Mahani, Electrical Engineering Department, Shahid Bahonar University of Kerman, Kerman, Iran
E-mail address: nikouei@mahani.info

Mohamad Koohi Moghadam, School of Computer Engineering, Iran University of Science and Technology, Tehran, Iran
E-mail address: m_koohi_m@comp.iust.ac.ir

Hosein Nezamabadi-pour*, Electrical Engineering Department, Shahid Bahonar University of Kerman, Kerman, Iran
E-mail address: nezam@mail.uk.ac.ir

*Corresponding author