

Research Article

Edge Detection via Edge-Strength Estimation Using Fuzzy Reasoning and Optimal Threshold Selection Using Particle Swarm Optimization

Ajay Khunteta and D. Ghosh

Department of Electronics and Communication Engineering, Indian Institute of Technology Roorkee, Roorkee, Uttarakhand 247 667, India

Correspondence should be addressed to Ajay Khunteta; ajaykhunteta@ymail.com

Received 2 July 2014; Revised 13 November 2014; Accepted 14 November 2014; Published 14 December 2014

Academic Editor: Katsuhiko Honda

Copyright © 2014 A. Khunteta and D. Ghosh. 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.

An edge is a set of connected pixels lying on the boundary between two regions in an image that differs in pixel intensity. Accordingly, several gradient-based edge detectors have been developed that are based on measuring local changes in gray value; a pixel is declared to be an edge pixel if the change is significant. However, the minimum value of intensity change that may be considered to be significant remains a question. Therefore, it makes sense to calculate the edge-strength at every pixel on the basis of the intensity gradient at that pixel point. This edge-strength gives a measure of the potentiality of a pixel to be an edge pixel. In this paper, we propose to use a set of fuzzy rules to estimate the edge-strength. This is followed by selecting a threshold; only pixels having edge-strength above the threshold are considered to be edge pixels. This threshold is selected such that the overall probability of error in identifying edge pixels, that is, the sum of the probability of misdetection and the probability of false alarm, is minimum. This minimization is achieved via particle swarm optimization (PSO). Experimental results demonstrate the effectiveness of our proposed edge detection method over some other standard gradient-based methods.

1. Introduction

Edge detection is an essential and important first step in object identification. An edge may be defined as a set of connected pixels lying at the boundary between the foreground and the background. Therefore, edge detection algorithms generally rely on detecting discontinuities within an image. Several gradient-based edge detectors are available in the literature which are based on measuring local changes in gray value; a pixel is declared to be an edge pixel if the change is significant. Accordingly, the underlying principle in most edge detection techniques is to compute the first- or second-order derivative of the intensity function within the image-detectors based on the first derivative looks for points where the derivative value is large while those using the second-order derivative find edges at zero-crossings of the image [1]. Several gradient operators, such as the Roberts, Prewitt,

Sobel, and the Laplacian masks, exist which are used to estimate the first- and the second-order derivatives [2]. However, these detectors are generally very sensitive to noise and hence perform poorly in case of noisy images. In [3], Canny proposed a method to counter this noise problem by convolving the image with the first-order derivatives of Gaussian filter prior to edge detection. Some other edge detectors that incorporate linear filtering, local orientation analysis, fitting of analytical models, and local energy are available in the literature [4–7].

However, the operators mentioned above are referred to as “noncontextual” or general edge detectors since they do not make any distinction between edges originating from textured regions and object boundaries. “Contextual” edge detectors, on the other hand, selectively detect contours (object boundaries) that are of interest in the context of a specific computer vision task by taking into account additional

information around an edge, such as local image statistics, image topology, perceptual differences in texture, and edge continuity. Contour detectors are mainly divided into local and global operators. Local detectors are mainly based on differential analysis, statistical approaches, phase congruency, rank-order filters, and their combinations. The latter class of detectors include computation of contour saliency, perceptual grouping, relaxation labeling, and active contours. The behavior of edges in the scale space has also been critically studied and evaluated [8]. Natural images generally contain edges with different levels of blurring. Further, the human visual system perceives different frequency ranges differently. Accordingly, multiresolution analysis finds importance in contour detection. Multiresolution contour detection may be classified into edge focusing and position-dependent blurring. Edge focusing can achieve both high noise rejection and good edge localization but at the cost of increased computational cost. Position-dependent blurring, on the other hand, is a better option for images that contain different degrees of blurring. Inspired by the surround suppression mechanism exhibited by the human visual system, contour detection methods using model for surround suppression via receptive field inhibition have also been developed [9–11]. The operator responds strongly to isolated lines and edges, region boundaries, and object contours, while suppressing texture edges. This yields better discrimination between object contours and texture edges, thereby improving contour detection performance. A comprehensive review on the various approaches to contour detection that have been developed in the past two decades is available in [12].

Gradient-based method for edge detection faces the challenge of edge localization in images that exhibit smooth transition in gray level. This is due to the ambiguous nature of the edge structures in such images. Also, since no absolute ground truth for the gradient threshold is available, clear demarcation between the pixels with high local intensity gradient (edge pixels) and those with low intensity gradient (non-edge pixels) does not exist. To deal with this ambiguity and vagueness in edge structures, some researchers used fuzzy logic theory in defining edges, as summarized below.

1.1. Fuzzy-Based Edge Detection. Fuzzy logic plays key role in situation of ambiguity. This motivated researchers to employ fuzzy reasoning for edge detection. The earliest and the most typical fuzzy-based edge detection method is due to Pal and King [13] which used fuzzy based logic as a contrast intensifier to detect edges in X-ray image. This was followed by a series of works in the last three decades, [14–27] to name a few.

The method presented by Tao et al. in [14] is based on a set of sixteen fuzzy IF-THEN rules. All these rules are combined to generate a set of potential edge pixels. This method avoids the difficulty of selecting parameter values present in most other edge detectors. Russo and Ramponi used FIRE operator (set of fuzzy inference rules) to detect edges from noiseless as well as noisy images [15]. In [16], Russo used 3×3 window-based filtering followed by fuzzy reasoning to detect edges in the presence of noise. A fuzzy classifier was used by Liang and

Looney [17] to classify image pixels corresponding to gray level variation in various directions by using a 3×3 mask. They used an extended Epanechnikov function as fuzzy set membership function for each class. The class assigned to each pixel is the one with the highest fuzzy membership. A fuzzy-based approach to edge detection in gray-level images is proposed in [18]. This fuzzy edge detector involves two phases: global contrast intensification and local fuzzy edge detection. In the first phase, a modified Gaussian membership function is chosen to represent each pixel in the fuzzy plane. This requires use of some parameters to enhance the image which are obtained by optimizing an entropy function. Rakesh et al. [19] proposed an edge detector in which thresholding is performed using statistical principles. Local standardization of threshold for each individual pixel, depending on the statistical variability of the gradient vector at that pixel, is taken into account for image binarization. In [20], Mendoza et al. applied Sobel operator on a digital gray-scale image and calculated the intensity gradients at each pixel position. This measure of intensity gradient gives the probability of occurrence of an edge. High pass and low pass filter masks are used to detect object boundaries in low contrast region. Finally, type-2 fuzzy inference is used to detect edges. Edge detection in blurry images was carried out by Wu et al. in [21]. For this, contrast of blurry image is first enhanced by means of fast multilevel fuzzy enhancement (FMFE) algorithm and then edges are extracted from the enhanced image by two-stage edge detection operator that identifies the edge candidates based on the local characteristics of the image. A novel edge detector based on fuzzy IF-THEN inference rules to model edge continuity criteria was proposed in [22]. Eight masks are used to get the gradient information. The maximum entropy principle is used for adjusting the parameter values and finally a set of fuzzy rules is used to decide edge pixels. Yang modified the classical Pal and King algorithm with a new fuzzy membership function [23]. To make the algorithm adaptive, membership function is defined automatically from the threshold of the image. In another paper [24], Madasu and Vasikarla proposed fuzzy edge detection in biometric systems. In this approach, edge detection is carried out by means of global (histogram of gray levels) and local (pixels within a window) information. The local information is fuzzified by employing a modified Gaussian membership function. Alshennawy and Aly proposed a method in [25] that is based on fuzzy logic reasoning for edge detection in digital images without determining the threshold value. The proposed approach begins by segmenting the images into regions using floating 3×3 binary matrix and finally the edge pixels are mapped to a range of values distinct from each other. An adaptive neurofuzzy inference system (ANFIS) for edge detection in digital images was used by Zhang et al. in [26]. The internal parameters of the proposed ANFIS edge detector are optimized by training with the help of very simple artificial images. The algorithm uses 81 rules and four 3×3 masks to detect edges in four directions. In [27], Melin et al. used morphological gradient and fuzzy logic to detect edges and an interval type-2 fuzzy inference system (IT2FIS) are used for improving the edge detection.

1.2. Edge Detection Using Evolutionary Algorithms. Nature-inspired evolutionary algorithms like PSO, bacterial foraging algorithm (BFA), ant colony optimization (ACO), genetic algorithm (GA), and so forth have recently been also applied in solving many complex problems including edge detection. A GA-based edge detection technique for texture image was proposed as early as in 1997 [28]. In this method, the edge detection problem is formulated as a combinatorial optimization problem and detection of the edge is executed according to the variance of texture feature in the local area. The candidate edge regions are selected first and then GA is applied in order to decide the optimum edge regions. A novel approach for edge detection based on the theory of universal gravity was presented in [29]. The algorithm assumes that each image pixel is a celestial body with a mass represented by its gray-scale intensity. Accordingly, each celestial body exerts force onto its neighboring pixels and in return receives force from the neighboring pixels. These forces can be calculated by the law of universal gravity. The vector sum of all gravitational forces along the horizontal and the vertical directions is used to compute the magnitude and the direction of signal variation. Edges are characterized by high magnitude of gravitational forces along a particular direction and can therefore be detected. Edge detection based on the fusion of fuzzy heuristic and PSO has been developed in [30]. Edge detection using ant colony optimization and adaptive thresholding was proposed in [31]. Ant colony is used to obtain well-connected edge-map. Ant movements are guided by local variation in intensity values. A novel edge detection technique that combines BFA with probabilistic derivative was proposed by Verma et al. in [32]. In this approach, the direction of movement of the bacteria decided by probability matrix is computed using derivatives along the possible edge directions. To deal with noisy images, this method is modified in [33] by using fuzzy derivative in place of probabilistic derivative. A reliable and accurate method of edge detection via tuning of parameters in BFA-based optimization has also been proposed recently in [34].

1.3. Objective of the Paper. In gradient-based image edge detection, thresholding is always a point of concern. Performance of classical edge detectors is very much dependent on the value of threshold chosen. This makes the use of these edge detectors relying on heuristic value manually selected by the user. However, proper choice of the threshold is necessary for selective detection of “contextual” edges that generally form object contours and region boundaries while leaving out “noncontextual” textured edges. Accordingly, in our work, we aim at developing user independent automatic method for determining the optimal threshold value such that only the object contours and region boundaries, which are relevant to object recognition, shape analysis, and image segmentation point of view, are detected.

Recently, we proposed a novel edge detection technique that uses a combination of fuzzy reasoning as well as threshold optimization for the purpose of edge detection [35]. The proposed method starts with intensity gradient calculation,

similar to other gradient-based edge detectors. Following this, unlike conventional gradient-based edge detectors, we use a set of fuzzy rules to measure the possibility of a pixel to be an edge pixel rather than taking a hard decision at this stage. We refer to this measure as “*edge-strength*.” Edge-strength at every pixel point is estimated using a fuzzy rule-based inference mechanism. Pixels are subsequently classified as edge or non-edge pixels on the basis of an optimal edge-strength threshold. This optimal threshold is calculated such that the overall probability of error in edge detection is minimized. We propose to employ particle swarm optimization (PSO) [36] technique for the purpose of minimization wherein the objective function is the sum of the probability of misdetection and the probability of false alarm. The purpose of using PSO for minimization is that it is relatively simple metaheuristic algorithm that is easy to implement and capable of dealing with complex search spaces where only minimum knowledge is available. PSO provides relatively fast convergence to the global minimization through a heuristic search within a search space consisting of infinite search points. In this paper, we describe our earlier proposed edge detection algorithm in more details. This is presented in Sections 2 and 3. Our proposed method for edge-strength calculation using fuzzy rules is described in Section 2 while our proposed optimal threshold selection procedure is given in Section 3. In Section 4, we compared our proposed method with some other previously reported methods of edge-detection. Finally, we draw our conclusion in Section 5.

2. Proposed Fuzzy Rule-Based Edge-Strength Calculation

The underlying principle behind gradient-based edge detection is to calculate the intensity gradient at every pixel point in the image and then labeling those pixels as edge pixels where the intensity gradient is *high*. Therefore, it is necessary to define a criterion for deciding *high* and *low* intensity gradients. In our proposed method, gradient at each image pixel is calculated by any available gradient operator following which a set of fuzzy rules is used to decide whether the edge-strength at a pixel point is low, medium, or high. This way an edge-strength map of the complete image is obtained.

We propose to use fuzzy theory for estimating the *edge-strength* at every pixel point in the input image. In our proposed approach, we first calculate the intensity gradients G_x and G_y along the horizontal and vertical directions, respectively, at every pixel (x, y) in the image. Next, we determine the edge-strength at that pixel point using several fuzzy rules. A fuzzy inference system that uses these fuzzy rules gives a measure of the edge-strength which is subsequently fuzzified to assess the extent to which the pixel relates to a true edge in the image.

In our proposed method for edge detection, we first determine the edge-strength so that edges can be detected appropriately. For this, we propose to use some “IF ... THEN ...” fuzzy rules and estimate the edge-strength using a fuzzy rule-based inference mechanism. A fuzzy rule typically includes a group of “antecedent clauses” which define

conditions and a “consequent clause” which defines the corresponding output action and/or conclusion. These fuzzy rules give directives much similar to human-like reasoning. Such a rule, which is expressed in plain linguistic form, is translated into the more formal structure of a fuzzy operator.

As said above, edge pixels are identified by measuring local change in intensity followed by thresholding. The change in intensity may be measured by taking the first derivative of the image function. The derivative quantifies the rate of change in the intensity and hence yields high values at points of rapid transition. In continuous domain, the rate of change of a 2D image intensity function $f(x, y)$ is given by the gradient vector, defined as

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} = \begin{bmatrix} G_x \\ G_y \end{bmatrix}, \quad (1)$$

where G_x and G_y denote the rate of intensity change along the horizontal (x -axis) and vertical (y -axis) directions, respectively. The magnitude of the gradient vector ∇f , referred to as *gradient*, gives the measure of the rate of change in intensity at the pixel location (x, y) . This is calculated as

$$G(x, y) = \sqrt{G_x^2 + G_y^2} = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}. \quad (2)$$

To reduce the computational burden, the gradient is sometimes approximated as

$$G(x, y) = |G_x| + |G_y|. \quad (3)$$

However, the degree of edginess or edge-strength at a pixel point is to some extent a subjective issue that may not be necessarily quantified in terms of G_x and G_y , as in (2) and (3) above. On the other hand, the edge-strength at a point is in every way related to the rate of change in intensity at that point. That means the degree of edge-strength at a pixel point may be estimated on the basis of combined G_x and G_y information. This may be accomplished using fuzzy reasoning and fuzzy inferencing, the way similar to the human reasoning.

In discrete domain, as in case of digital images, it is a common practice to use the first difference in place of the first-order partial derivative. Accordingly, the digital gradients are generally computed using gradient operators consisting of two masks: one mask for computing the horizontal gradient G_x and the other for computing the vertical gradient G_y . Several gradient operators, such as the Roberts, Prewitt, and Sobel masks, are available in the literature for that purpose [2]. In our method, we propose to use any such gradient operator for computing the gradients G_x and G_y at a point (x, y) in the image. These two gradient values are then combined to determine the edge-strength at that point using fuzzy reasoning. Thus, our proposed approach differs from the conventional gradient-based edge detectors. Our proposed method for edge-strength calculation uses deductive fuzzy inference system based on a set of four fuzzy rules, as stated below.

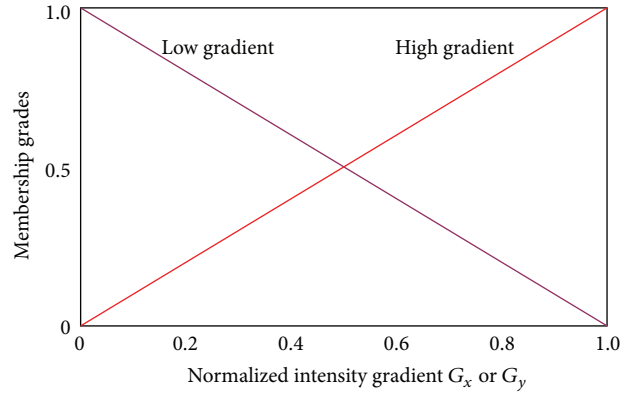


FIGURE 1: Membership function plots for fuzzy sets corresponding to low and high intensity gradient (normalized).

- (i) *Rule 1.* IF G_x is low AND G_y is low THEN edge-strength $E(x, y)$ is low.
- (ii) *Rule 2.* IF G_x is low AND G_y is high THEN edge-strength $E(x, y)$ is medium.
- (iii) *Rule 3.* IF G_x is high AND G_y is low THEN edge-strength $E(x, y)$ is medium.
- (iv) *Rule 4.* IF G_x is high AND G_y is high THEN edge-strength $E(x, y)$ is high.

Thus, our fuzzy system is a two-input and one-output system: two input variables (antecedent clauses) are the two gradients G_x and G_y , and the resultant output variable (consequent clause) is the normalized edge-strength $E(x, y)$, at pixel position (x, y) in the image. These four IF-THEN rules include two conditions (antecedent clauses) about the input variables and specify a consequent clause related to the output variable. Antecedent clauses are linked by fuzzy AND operator. Each clause is completely defined by the shape and position of a fuzzy set, which maps the corresponding variable to the real interval $[0, 1]$. Since the two input variables G_x and G_y are measures of the same quantity (intensity gradient) differing only in their directions of measure (horizontal and vertical directions), they may be defined by the same set of fuzzy sets. Two fuzzy sets for the antecedents and three fuzzy sets for the consequent are used, as represented in Figures 1 and 2. Fuzzy sets for the antecedents are labeled as “low gradient” and “high gradient.” Fuzzy sets for the consequent are “low edge-strength,” “medium edge-strength,” and “high edge-strength.”

The above stated fuzzy rules are combined using Mamdani’s max-min inference method followed by centroid-based defuzzification. A detailed description of the inference mechanism can be found in [37]. Thus, the “edge-strength” $E(x, y)$ is calculated on the basis of the extent of horizontal and vertical intensity gradients at pixel location (x, y) in the input image.

3. Edge-Strength Threshold Selection

After deriving the information regarding edge-strength of an image, as discussed above, now the question is how to

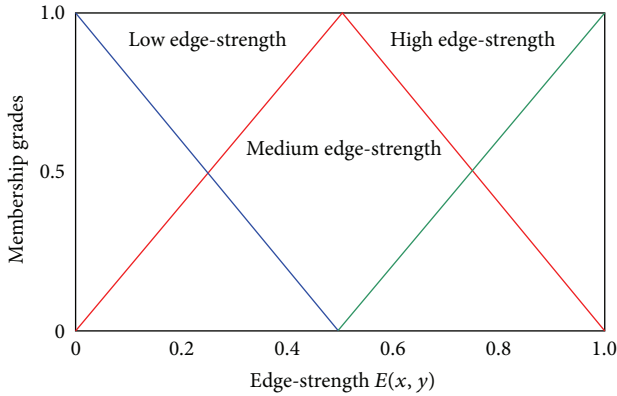


FIGURE 2: Membership function plots for fuzzy sets corresponding to low, medium, and high edge-strength.

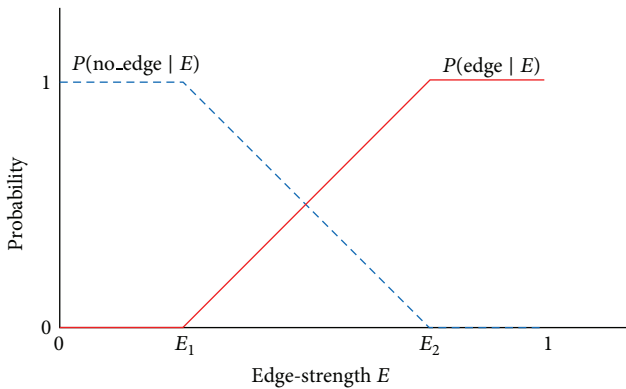


FIGURE 3: Trapezoidal functions representing the conditional probabilities for a pixel to be “edge” or “not edge,” given edge-strength E of the pixel.

decide the threshold value automatically to differentiate edge pixels from non-edge pixels in the image. This implies that the choice of threshold is very important in detecting edge pixels—higher value of threshold may result in misdetection while lower value of threshold may result in false detection. In other words, the performance of any gradient-based edge detector depends on the choice of threshold.

The edge-strength calculation step described above maps the input image to a gray-scale edge-map—the gray value at every pixel location corresponds to the edge-strength of that pixel, that is, how much likely is that pixel to be an edge, measured in the range from 0 to 1. However, a binary edge-map is generally desired which shows whether a pixel is an edge or not an edge. So, following the above edge-strength calculation step, a decision making step is necessary through which the pixels in the edge-map are classified into two categories, namely, “edge pixel” and “non-edge pixel.” Thus, the problem at hand essentially boils down to image binarization via 2-class pattern classification that involves selection of an appropriate decision boundary (threshold). Several methods of an optimal threshold for image binarization have been reported in the literature and discussed below.

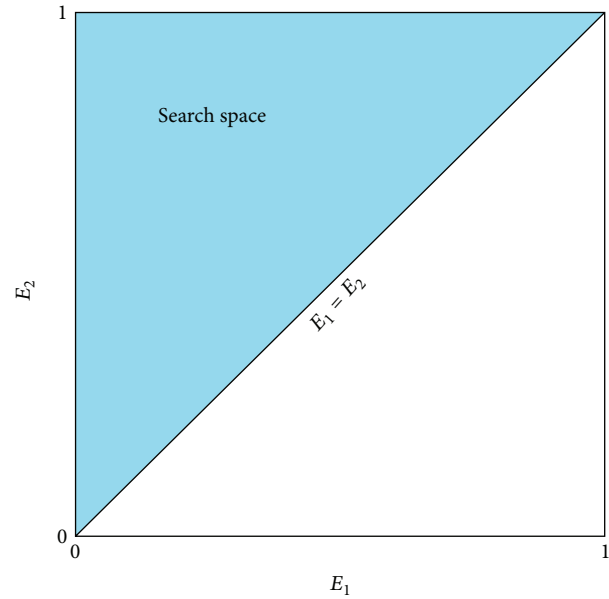


FIGURE 4: Search space for optimization to determine the values of E_1 and E_2 .

3.1. Methods for Threshold Selection for Image Binarization and/or Edge Detection. Thresholding is a commonly used technique in image segmentation and binarization. Selecting the correct threshold is a critical issue—lower value of threshold increases false alarm while higher value of threshold increases the probability of misdetection. Accordingly, many researchers attempted to address this issue by devising techniques for selection of optimal threshold. One such early attempt in this direction by entropic thresholding was proposed by Pun [38]. Kapur et al. [39] also attempted automatic thresholding using entropy of the histogram for gray level images. In [40], a method for automatically selecting threshold is presented which uses a model based on the weighted sum of two gamma density functions corresponding to edge and non-edge pixels. A robust automatic threshold selection algorithm was introduced in [41] which is based on the average of the gray levels of the pixels in an image weighted by the response of a specific edge detector at each pixel. A technique of three-level thresholding based on probability partition and fuzzy 3-partition has been presented in [42] in which a relationship between a probability partition and a fuzzy partition in thresholding is given. This relationship and the entropy approach are used to derive a thresholding technique to select the best fuzzy partition. In [43], El-Khamy et al. considered the problem of fuzzy edge detection as a two-level thresholding problem where the goal is to partition the image domain according to the gradient value into two fuzzy partitions (regions). These two regions relate to pixels that acquire low local gradient value (smooth region) and pixels that acquire high local gradient value (edge region). Using the relation of the probability partition and the fuzzy 2-partition of the image gradient, the best gradient threshold is selected via minimization of fuzzy entropy. An automatic threshold selection method was proposed in [44] which considers

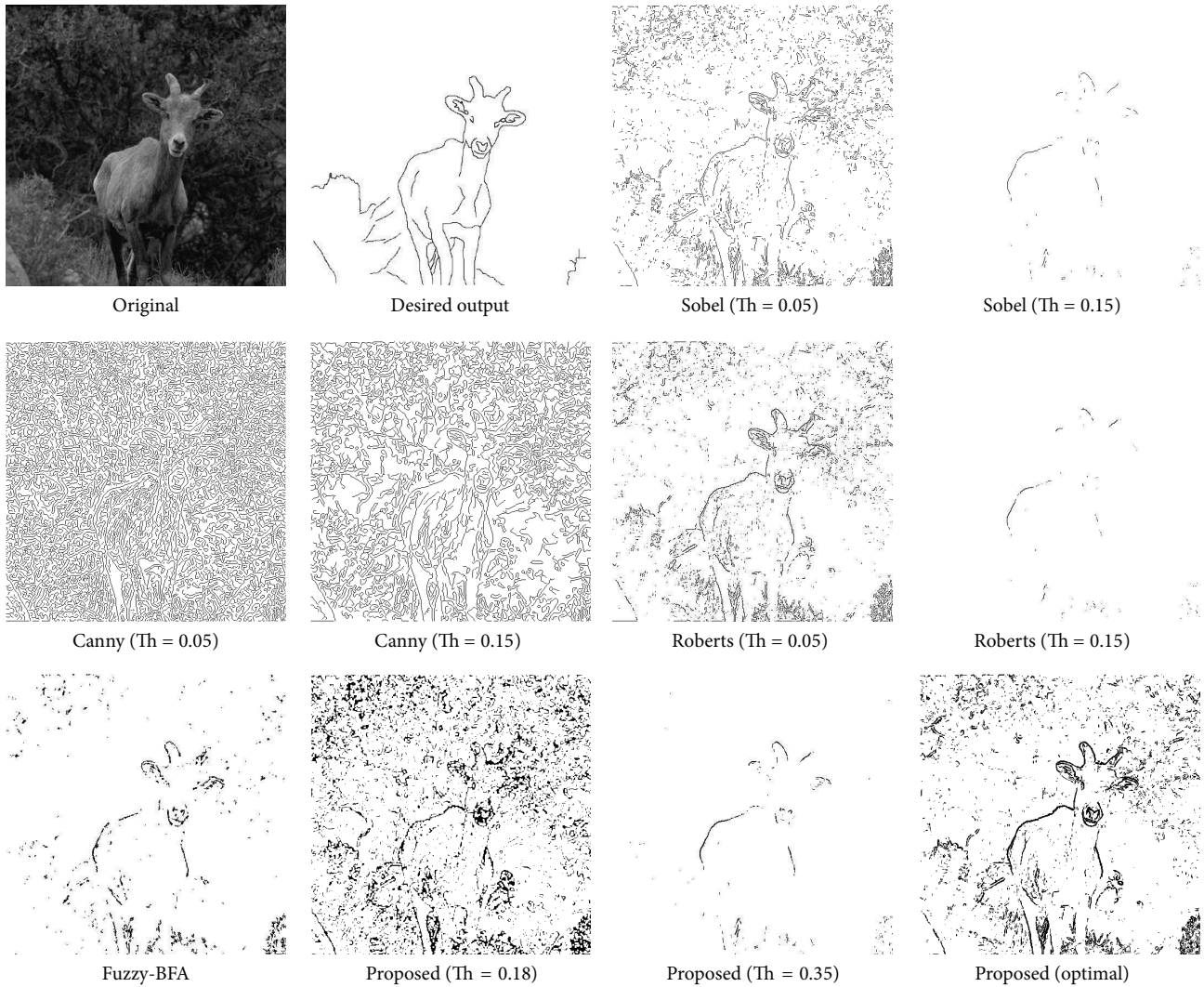


FIGURE 5: Edge detection results of “goat” image: desired edge output of “goat” image; Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

both statistical and spatial correspondence between detected edge points. Local thresholds are adaptively selected in [45] taking into account the activity masking characteristic of the human visual system. The selected local thresholds are then utilized for edge labeling in the gradient image. An adaptive thresholding-based edge detection method using morphological operators is presented in [46]. A double-threshold image binarization method based on the edge detector was proposed in [47] which is based on the edge and intensity information. Image binarization is done in two stages using high and low threshold values. Zhang et al. presented fuzzy edge detection based on Gauss function and automatic selection of threshold parameter by maximum variance [48]. Li and Gao [49] proposed an edge detection algorithm that is based on fuzzy 2-partition entropy approach. The authors addressed the problem of detecting the edge pixels in gray-level image by maximum fuzzy 2-partition entropy principle. Entropy function used to justify that the information of an

image is mostly retained after thresholding. Another entropy-based flexible thresholding algorithm was proposed in [50]. Three different methods for adaptive thresholding using fuzzy approach, statistical parameters, and fuzzy reasoning were proposed in [51].

3.2. Proposed Method for Optimal Edge-Strength Threshold Selection. In our work, we propose using Bayes decision theoretic approach for optimal threshold selection so as to minimize the overall error in categorizing the pixels into “edge pixels” and “non-edge pixels.” Error occurs when an actual edge pixel is categorized as “non-edge pixel” (misdetection) or vice versa (false alarm).

According to Bayes decision rule [52], a pixel at location (x, y) with edge-strength E is marked as an edge pixel if

$$P(\text{edge} | E) > P(\text{no_edge} | E), \quad (4)$$



FIGURE 6: Edge detection results of “hyena” image: desired edge output of “hyena” image; Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

where $P(\text{edge} | E)$ and $P(\text{no_edge} | E)$ denote the conditional probabilities that the pixel is an edge or not an edge, respectively. It follows that the decision boundary (threshold) between non-edge and edge pixels is given by the edge-strength value E for which

$$P(\text{edge} | E) = P(\text{no_edge} | E). \quad (5)$$

Therefore, the task now is to seek for suitable functions defining the conditional probabilities $P(\text{edge} | E)$ and $P(\text{no_edge} | E)$ such that the overall probability of classification error is minimized. Once these two probability functions are obtained, the required threshold is automatically established.

In connection to the above stated task of finding suitable probability functions, we note the following.

- (i) Since a pixel can only be either an edge pixel with some probability or not an edge pixel, these two events

are mutually exclusive and form the total probability space. Therefore, we will have

$$P(\text{edge} | E) + P(\text{no_edge} | E) = 1, \quad \text{or} \quad (6)$$

$$P(\text{edge} | E) = 1 - P(\text{no_edge} | E).$$

- (ii) Since the edge-strength is the measure of the possibility of a pixel to be an edge pixel, it is intuitively desired that the probability $P(\text{edge} | E)$ be a monotonically non-decreasing function of the edge-strength measure E . It follows that $P(\text{no_edge} | E)$ will be a monotonically non-increasing function of E .
- (iii) The demarcation between “non-edge” and “edge” is generally fuzzy. This may be well characterized by defining a range of edge-strength value E , say, (E_1, E_2) , corresponding to pixels of such ambiguous nature. In other words, pixels with edge-strength

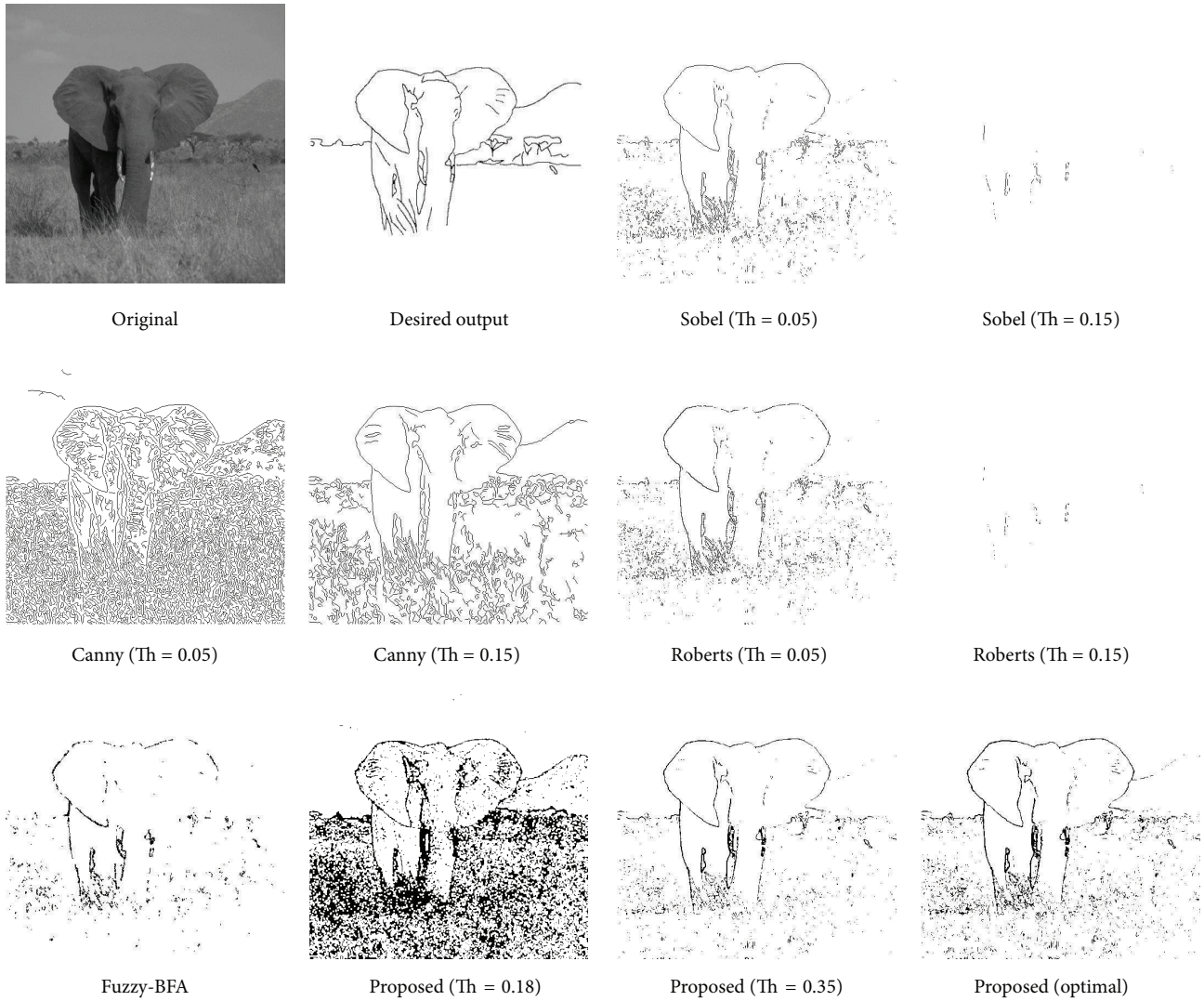


FIGURE 7: Edge detection results of “elephant” image: desired edge output of “elephant” image; Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

value in this range will have non-zero probability of being both an edge pixel and non-edge pixel; that is,

$$\begin{aligned} 0 < P(\text{no_edge} | E) < 1, \\ 0 < P(\text{edge} | E) < 1 \quad \text{for } E_1 < E < E_2. \end{aligned} \quad (7)$$

- (iv) Based on the preceding observation, a pixel is definitely an edge pixel if its edge-strength exceeds E_2 . On the other hand, pixels with edge-strength less than E_1 are definitely non-edge pixels. Accordingly, we will have

$$\begin{aligned} P(\text{no_edge} | E) &= 1 \quad \text{for } 0 \leq E \leq E_1, \\ P(\text{edge} | E) &= 1 \quad \text{for } E_2 \leq E \leq 1. \end{aligned} \quad (8)$$

3.2.1. Fuzzy Two-Partition of Normalized Image Edge-Strength. Following the above discussion, we propose two

trapezoidal functions defining $P(\text{no_edge} | E)$ and $P(\text{edge} | E)$, as depicted in Figure 3. The reason for this particular choice is its simplicity due to the linear nature of the functions in the intervals from E_1 to E_2 . Nevertheless, some other complex functions represented by higher-order curves in the range (E_1, E_2) may also be used, as per the will of the user.

The trapezoidal functions so chosen are completely defined when the values of E_1 and E_2 are known. The required decision threshold is then obtained as

$$E_{\text{th}} = \frac{1}{2}(E_1 + E_2). \quad (9)$$

Pixels with edge-strength value above E_{th} will be marked as edge pixels while those with edge-strength less than E_{th} will be classified as non-edges.

Thus, the problem of optimal threshold selection now reduces to the problem of finding a pair of suitable values for E_1 and E_2 such that the overall error in identifying edge pixels is minimized. Error occurs when an actual edge pixel

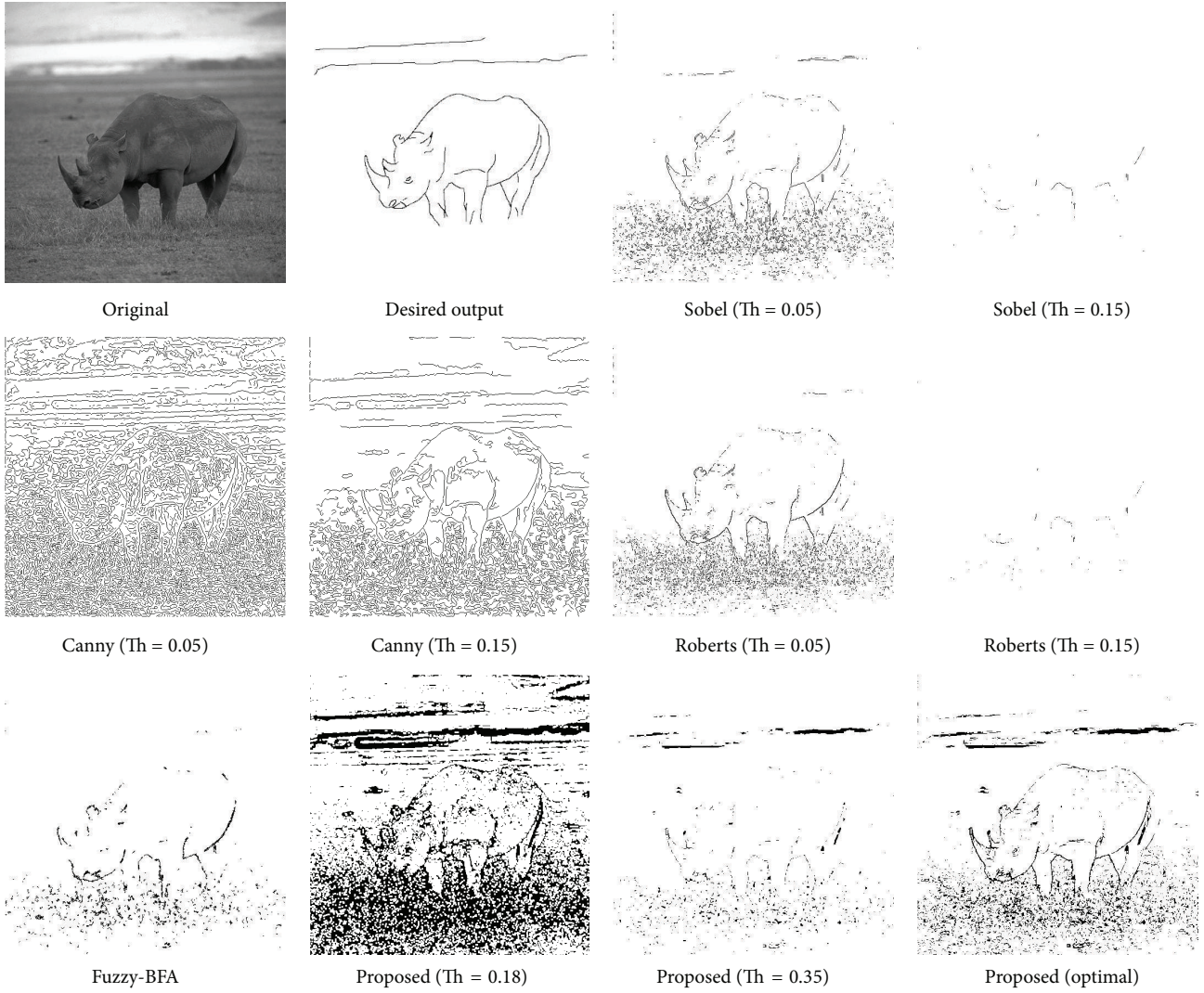


FIGURE 8: Edge detection results of “rhino” image: desired edge output of “rhino” image; Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

is not detected (mis-detection) or when an actual non-edge pixel is marked as an edge pixel (false alarm). The probability of mis-detection is given as

$$\begin{aligned}
 P_{MD} &= \int_0^{E_{th}} P(\text{edge} | E) p(E) dE, \\
 P_{MD} &= \int_0^{E_1} P(\text{edge} | E) p(E) dE \\
 &\quad + \int_{E_1}^{E_{th}} P(\text{edge} | E) p(E) dE, \quad (10) \\
 P_{MD} &= \int_{E_1}^{E_{th}} P(\text{edge} | E) p(E) dE \\
 &\quad (\because P(\text{edge} | E) = 0 \text{ for } 0 \leq E \leq E_1)
 \end{aligned}$$

and the probability of false alarm is calculated as

$$\begin{aligned}
 P_{FA} &= \int_{E_{th}}^1 P(\text{no_edge} | E) p(E) dE, \\
 P_{FA} &= \int_{E_{th}}^{E_2} P(\text{no_edge} | E) p(E) dE \\
 &\quad + \int_{E_2}^1 P(\text{no_edge} | E) p(E) dE, \quad (11) \\
 P_{FA} &= \int_{E_{th}}^{E_2} P(\text{no_edge} | E) p(E) dE \\
 &\quad (\because P(\text{no_edge} | E) = 0 \text{ for } E_2 \leq E \leq 1),
 \end{aligned}$$

where $p(E)$ is the probability density function of the edge-strength E . The overall probability of error in detecting edge

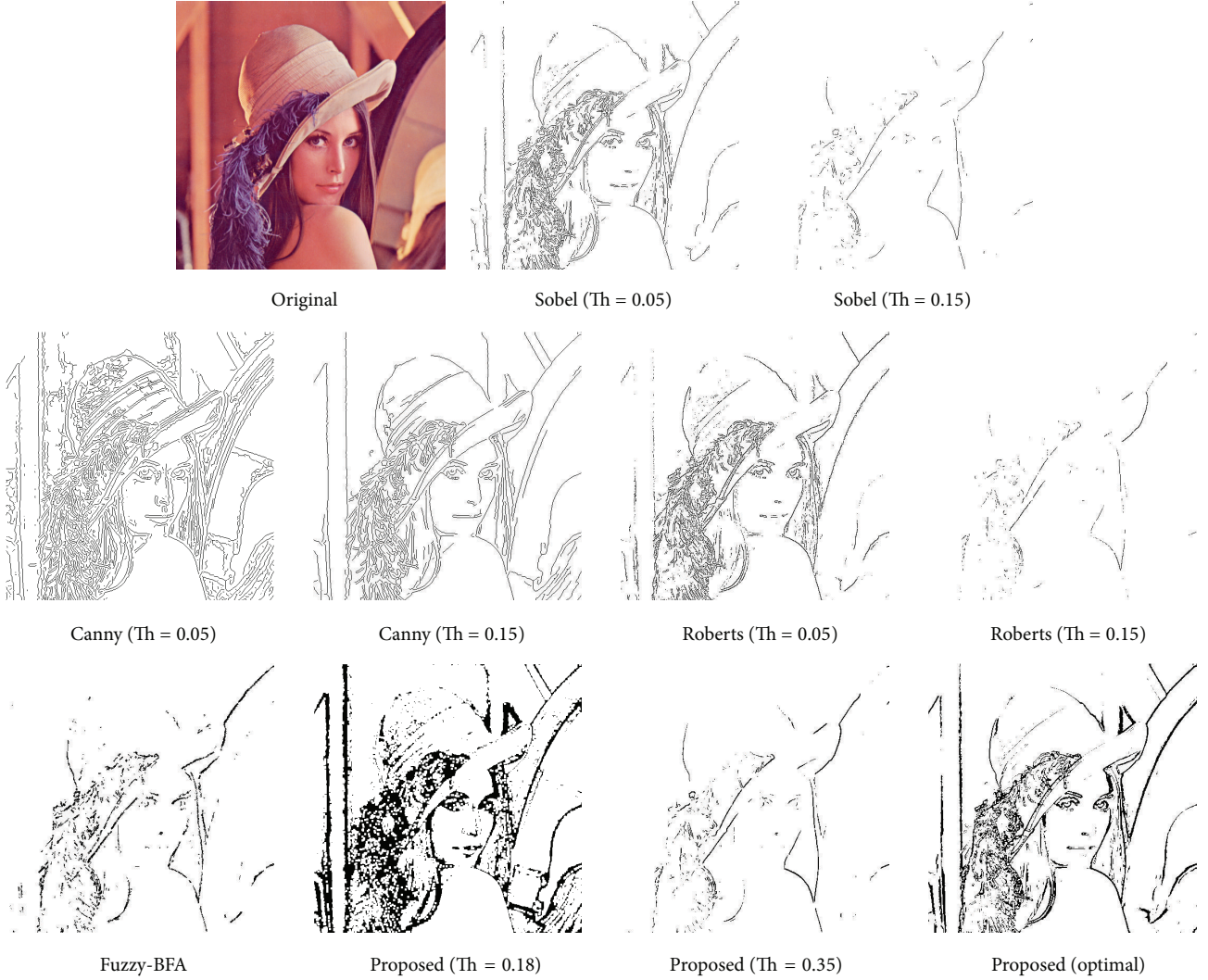


FIGURE 9: Edge detection results of “lena” image: Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

pixels in an image is the sum of the above two probabilities. That is,

$$P_{\text{error}} = P_{\text{MD}} + P_{\text{FA}}. \quad (12)$$

3.2.2. Defining Objective Function. In our proposed work, the overall probability of error is used as the objective function, that is, to be minimized in determining the values of E_1 and E_2 . It is to be noted that although the edge-strength E is a continuous random variable in the range from 0 to 1, in practice we obtain only a finite number of discrete edge-strength values in this range from a finite-sized image having finite number of pixels. In view of this, the above objective function needs to be expressed in its discrete version, given as

$$P_{\text{error}} = \sum_{k=E_1}^{E_{\text{th}}} \frac{(k - E_1)}{(E_2 - E_1)} \frac{n_k}{N} + \sum_{l=E_{\text{th}}}^{E_2} \frac{(E_2 - l)}{(E_2 - E_1)} \frac{n_l}{N}, \quad (13)$$

where n_k denotes the number of pixels having edge-strength equal to k and N is the total number of pixels in the image. The summation in the first term is over all available discrete edge-strength values in the range of E_1 to E_{th} . Similarly, the summation in the second term is over all discrete edge-strength values in the range of E_{th} to E_2 .

As discussed above, the aim now is to search for the values of E_1 and E_2 that minimize P_{error} ; the search space for the same is illustrated in Figure 4. It may be noted that any point on the line $E_1 = E_2$ yields $P_{\text{error}} = 0$ and hence corresponds to a trivial solution. Therefore, in order to avoid this trivial solution, the line corresponding to $E_1 = E_2$ is excluded from the search space.

The search space mentioned above consists of an infinite number of search points. Therefore, an exhaustive search is not feasible. In view of this, we propose using particle swarm optimization (PSO) [36]. It is a metaheuristic search procedure that has the capability of escaping from local minima and converging to the global minimum. PSO is

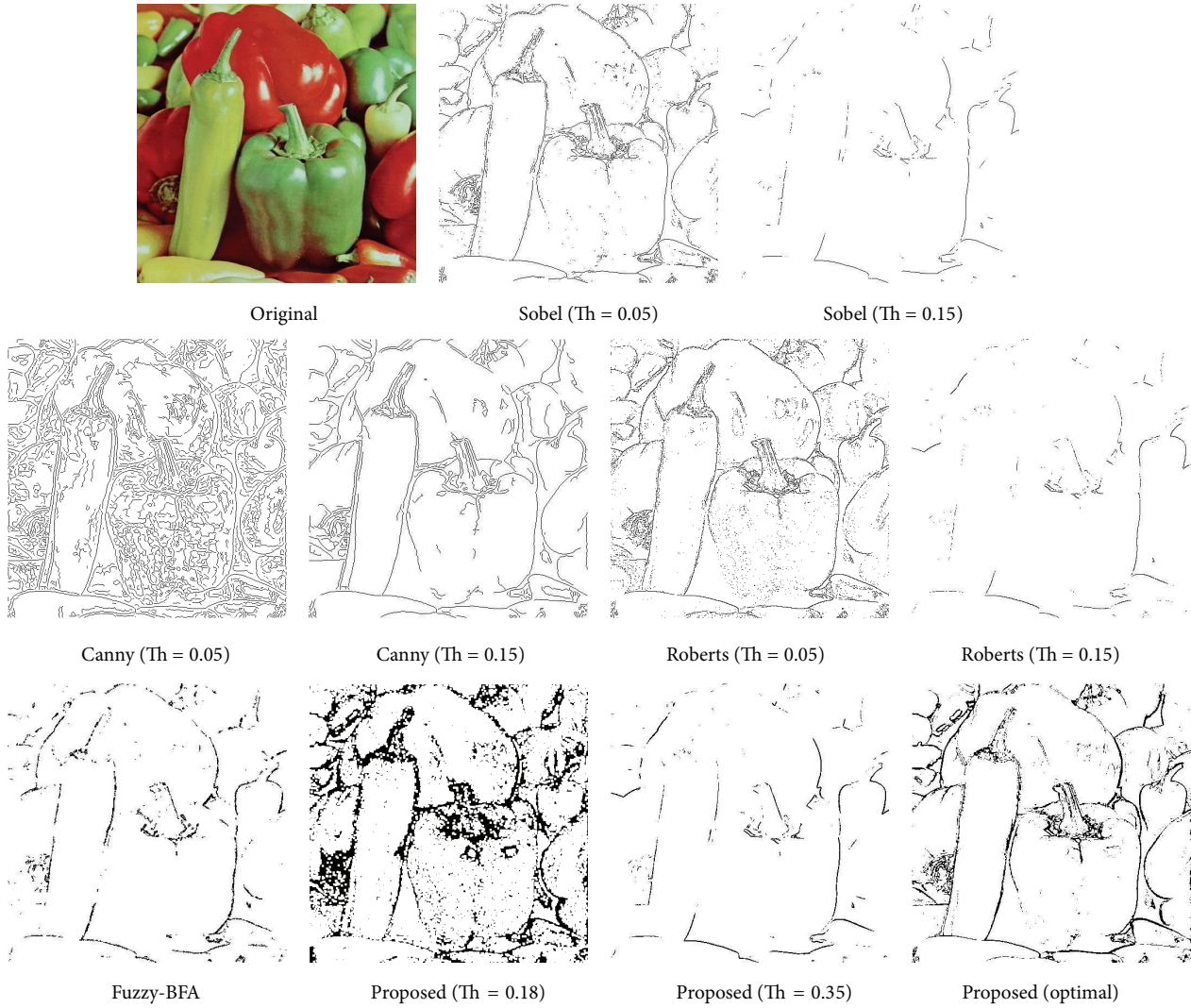


FIGURE 10: Edge detection results of “peppers” image: Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

simple, fast, and relatively stable from convergence point of view. This makes PSO a popular choice for carrying out the optimization operation in this particular situation.

3.2.3. Objective Function Optimization Using PSO. Now as expressed in (13), thresholding can be achieved by optimizing the given function and obtaining the values of E_1 and E_2 . The steps of the algorithm for minimization of P_{error} using PSO are as follows.

- (1) Set PSO parameters, namely, population size, maximum generation number, and inertia weight.
- (2) Set a swarm within the search space. The position of every swarm particle in the search space is given by a pair of E_1 and E_2 values.
- (3) Calculate P_{error} for every swarm particle corresponding to its position.

- (4) Assign initial $pbest$ and $gbest$ values for this swarm; $pbest$ “personal best” refers to the best position of a swarm particle till now and $gbest$ “global best” is the position of the particle having minimum P_{error} in the whole swarm.
- (5) Run PSO to modify velocity and position of every particle in the swarm using the following equations:

$$\begin{aligned} \mathbf{v}_i(t+1) &= \omega \mathbf{v}_i(t) + c_1 r_1 [pbest_i(t) - \mathbf{x}_i(t)] \\ &\quad + c_2 r_2 [gbest_i(t) - \mathbf{x}_i(t)], \end{aligned} \quad (14)$$

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1),$$

where ω is some predefined inertia weight, c_1 and c_2 are positive constants, r_1 and r_2 are random numbers between $[0, 1]$, $\mathbf{x}_i(t)$ is the position of the i th particle at time t , and $\mathbf{v}_i(t)$ is the velocity of the i th particle at time t . Generally, a large value of the inertia weight

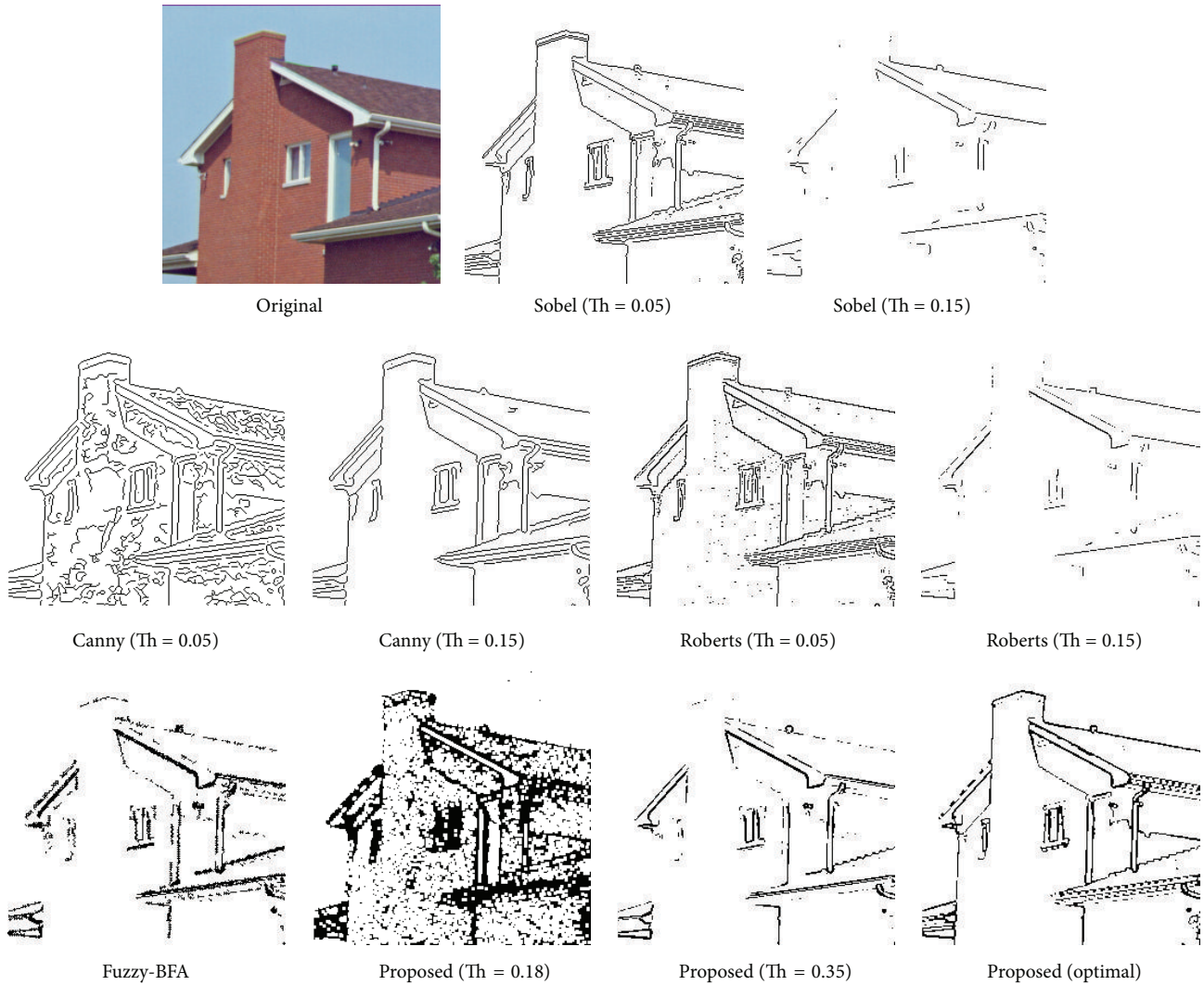


FIGURE 11: Edge detection results of “house” image: Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

ω is desired for global convergence. Note that the location of a particle is restricted at the boundary of the search space if its new position is outside the search space.

- (6) Calculate P_{error} for every swarm particle at its new position. The particle is retained at its new position if its recent P_{error} value is less than its previous value; else the particle is moved back to its earlier location.
- (7) Update p_{best} and g_{best} values in the swarm.
- (8) Repeat for maximum number of PSO generations. The final solution is given by the g_{best} obtained in the final step of iteration.

Thus, the E_1 and E_2 pair of values corresponding to the final solution in the search space is the desired result. The required optimal threshold is now calculated using (9) and then applied to obtain the final edge-map image.

4. Experimental Results and Comparison

In our experiments, we applied our proposed method for edge detection on gray and color images and compared its performance with adaptive threshold to that with fixed threshold values. We also compared the performance of our proposed method with three classical edge detectors, namely, Canny, Roberts, and Sobel, and with one of the recently reported algorithms, namely, bacterial foraging edge detection method given in [33].

In a first set of experiments, we used four gray-scale images, namely, “goat,” “hyena,” “elephant,” and “rhino,” available at [53]. For each of these images, the associated desired binary contour map, drawn manually by experts, was also available alongwith. The edge detection results obtained in all these cases, by applying the various edge detectors mentioned above, are shown in Figures 5, 6, 7, and 8. Using the performance measure introduced in [9], we evaluated the P

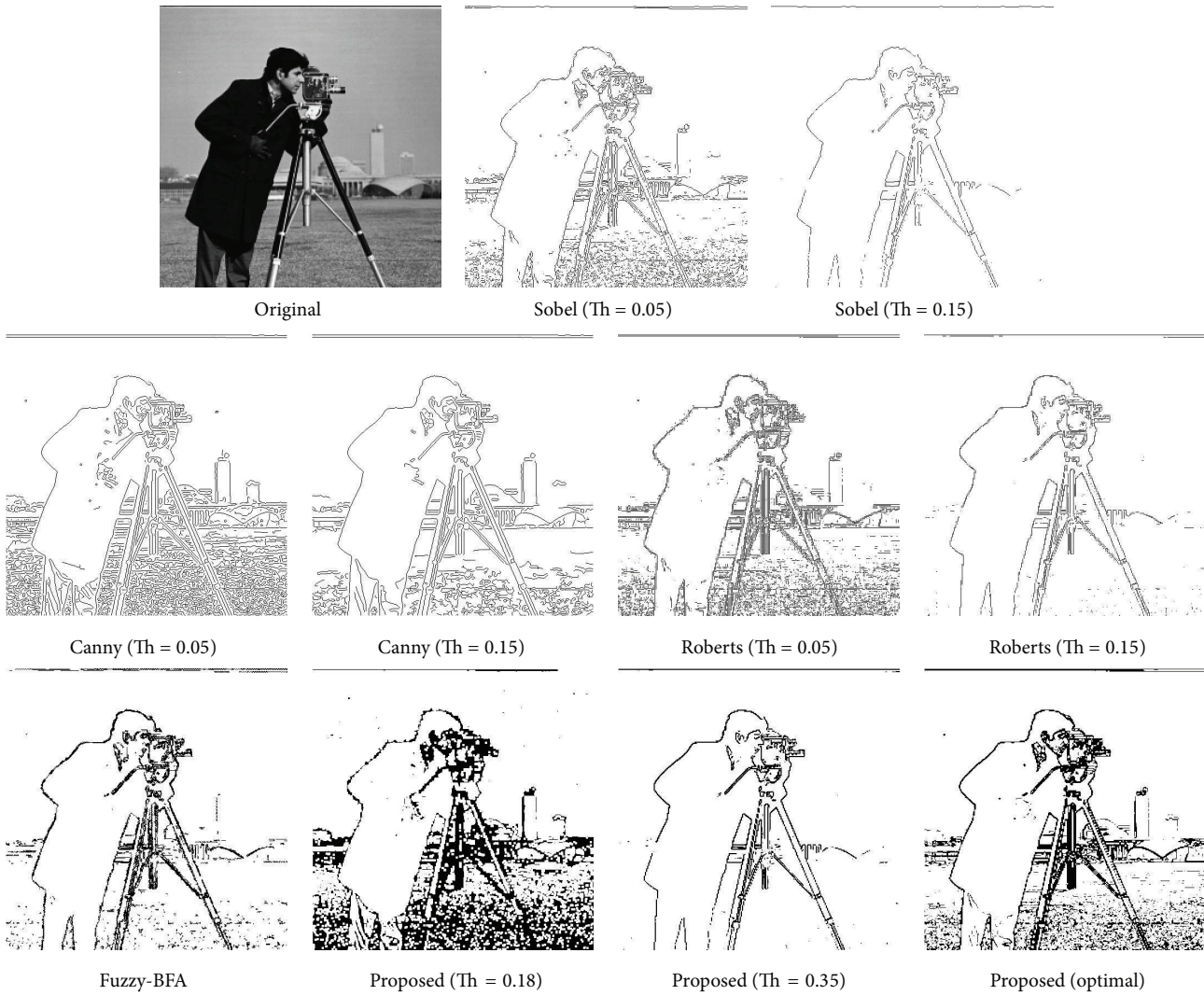


FIGURE 12: Edge detection results of “cameraman” image: Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

values, as defined below, for all these cases and tabulated in Table 1. Consider

$$P = \frac{\text{card}(E)}{\text{card}(E) + \text{card}(E_{FP}) + \text{card}(E_{FN})}, \quad (15)$$

where E is the set of correctly detected contour pixels, E_{FP} is the set of false positives, that is, non-edge pixels detected as edge pixels, E_{FN} is the set of false negatives, that is, edge pixels not detected, and $\text{card}(\cdot)$ denotes the cardinality of a set. It may be noted that the manually drawn contour does not necessarily always exactly coincide with a local maximum of the gradient magnitude operator. Hence, a contour pixel is considered to be correctly detected if a desired contour pixel is present in a 5×5 neighbourhood around the concerned pixel, as suggested in [9].

In a second set of experiments, we applied our proposed method for edge detection and the various other methods mentioned above on two gray and four color images as

shown in Figures 9, 10, 11, 12, 13, and 14. We observe that the performance of the Sobel and Roberts method is highly dependent on the choice of the threshold and shows high rate of misdetection even at lower threshold values. This problem is taken care of through additional processing such as edge-strength estimation and optimal thresholding in our method, as evident from the results shown. Canny’s classical edge detector is considered to be one of the best performing edge detection algorithms available in literature. We observe that while Canny’s method is highly dependent on the value of threshold selected, true edges are appropriately detected in our method by automatic thresholding. In case of BFA-based edge detection thick or double edges are obtained while better edge localization is achieved in our proposed method. We also observe that when fixed values of threshold are used in our method, there is more probability of misdetection (for high value of threshold) or more false alarm (for low value of threshold). On the other hand, selection of optimal threshold

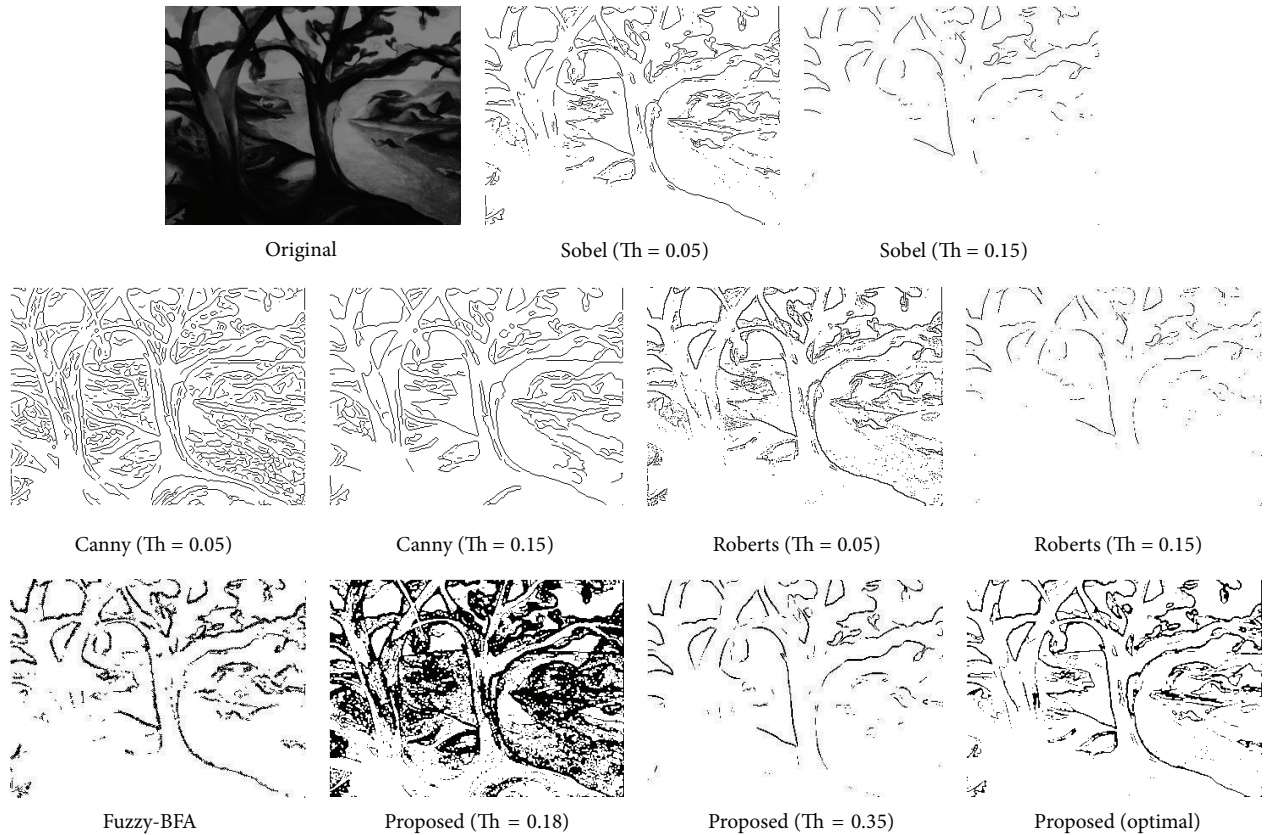


FIGURE 13: Edge detection results of “trees” image: Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

TABLE 1: P values obtained for different test images.

Methods	Test images			
	Goat	Hyena	Elephant	Rhino
Sobel (threshold = 0.05)	0.18	0.14	0.22	0.24
Sobel (threshold = 0.15)	0.21	0.29	0.12	0.10
Canny (threshold = 0.05)	0.08	0.10	0.13	0.09
Canny (threshold = 0.15)	0.11	0.13	0.20	0.16
Roberts (threshold = 0.05)	0.24	0.12	0.27	0.25
Roberts (threshold = 0.15)	0.23	0.30	0.09	0.12
Fuzzy-BFA	0.31	0.35	0.32	0.33
Proposed (threshold = 0.18)	0.13	0.07	0.17	0.09
Proposed (threshold = 0.35)	0.3	0.27	0.35	0.27
Proposed (optimal threshold)	0.23	0.37	0.34	0.29

through optimization aids in better edge detection. Threshold values obtained in our experiments are tabulated in Table 2.

5. Conclusion

This paper presents a fuzzy approach to edge detection via estimation of edge-strength at every pixel location in the input image. Following this, optimal thresholding is achieved by minimizing the probability of error in edge identification.

The method is an extension of the gradient-based edge detectors. Here the horizontal and vertical gradients are combined using some fuzzy rules to give a measure of the edge-strength. Decision regarding pixel to be considered as an edge or non-edge is based on the threshold value selection by optimizing (minimization) the edge detection error. In order to fast search through the infinite number of search points, meta-heuristic optimization approach such as the PSO is employed. Thus, this method does away with manual selection of threshold and hence is free from any user intervention.

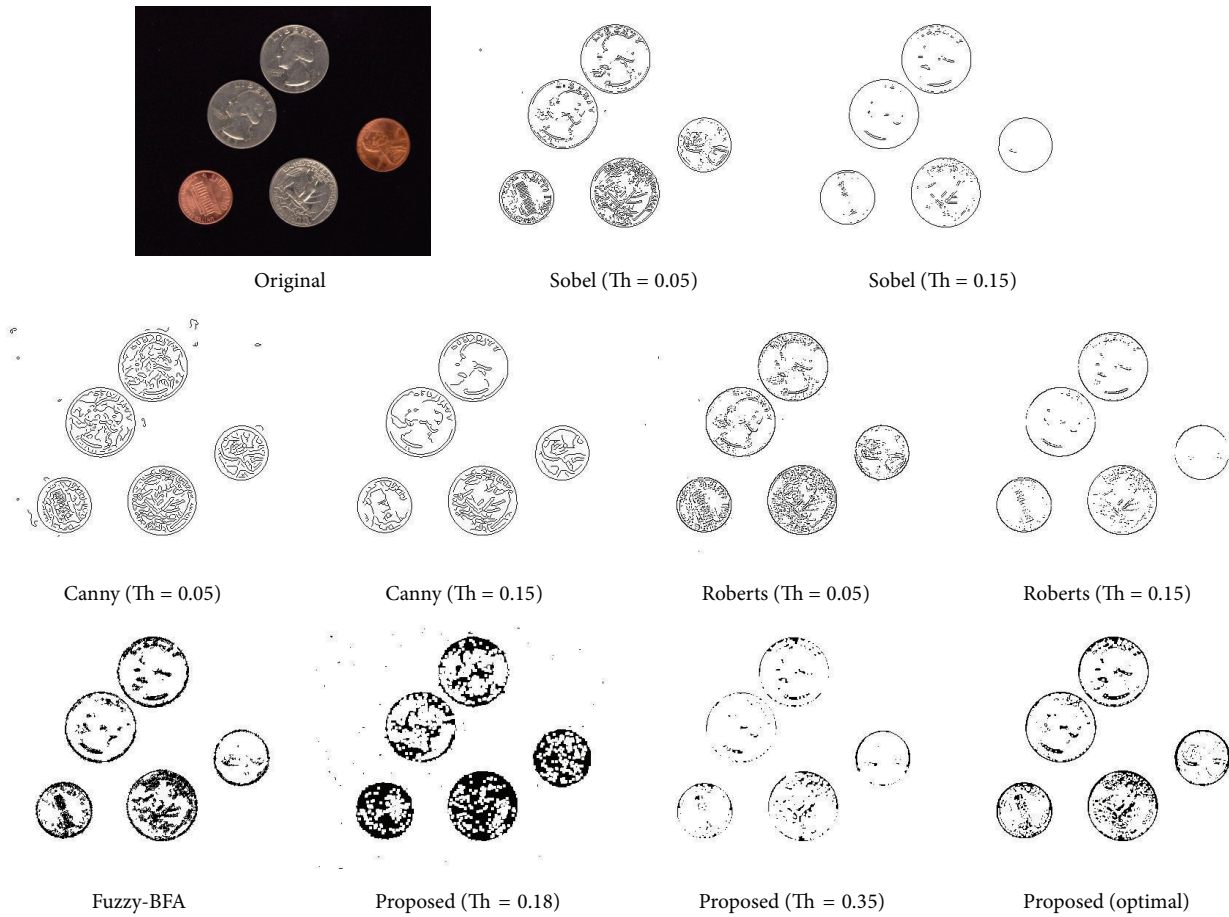


FIGURE 14: Edge detection results of “coins” image: Sobel, Canny’s, and Roberts methods using thresholds = 0.05 and 0.15; fuzzy-BFA method; proposed method with fixed thresholds (0.18 and 0.35) and optimal threshold.

TABLE 2: Threshold values obtained for different test images.

Test images	Different threshold values		
	E_1	E_2	Final threshold $(E_1 + E_2)/2$
Goat	0.1842	0.3078	0.2460
Hyena	0.1706	0.3020	0.2363
Elephant	0.1848	0.2552	0.2200
Rhino	0.1612	0.2888	0.2250
Lena	0.1989	0.2567	0.2278
Peppers	0.1969	0.2535	0.2252
House	0.1990	0.2654	0.2322
Cameraman	0.1978	0.2564	0.2271
Trees	0.1968	0.2564	0.2266
Coins	0.1979	0.2923	0.2451

However, as in case of most fuzzy systems, the performance of our proposed method obviously depends on the choice of the fuzzy sets and the fuzzy membership functions. Therefore, proper choice of the fuzzy functions is necessary to detect true edges in the input image. Experimental results show that the true edges are detected with high efficiency.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] D. Marr and E. Hildreth, “Theory of edge detection,” *Proceedings of the Royal Society of London B*, vol. 207, no. 1167, pp. 187–217, 1980.
- [2] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, Pearson Education (Prentice-Hall), 3rd edition, 2009.
- [3] J. Canny, “A computational approach to edge detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.
- [4] E. C. Hildreth, “The detection of intensity changes by computer and biological vision systems,” *Computer Vision, Graphics and Image Processing*, vol. 22, no. 1, pp. 1–27, 1983.
- [5] R. M. Haralick, “Digital step edges from zero-crossings of second directional derivatives,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 1, pp. 58–68, 1984.
- [6] V. S. Nalwa and T. O. Binford, “On detecting edges,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 699–714, 1986.

- [7] D. Ziou and S. Tabbone, "Edge detection techniques—an overview," *International Journal of Pattern Recognition and Image Analysis*, vol. 8, no. 4, pp. 537–559, 1998.
- [8] Y. Lu and R. C. Jain, "Behavior of edges in scale space," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 4, pp. 337–356, 1989.
- [9] C. Grigorescu, N. Petkov, and M. A. Westenberg, "Contour detection based on nonclassical receptive field inhibition," *IEEE Transactions on Image Processing*, vol. 12, no. 7, pp. 729–739, 2003.
- [10] C. Grigorescu, N. Petkov, and M. A. Westenberg, "Contour and boundary detection improved by surround suppression of texture edges," *Image and Vision Computing*, vol. 22, no. 8, pp. 609–622, 2004.
- [11] G. Papari and N. Petkov, "An improved model for surround suppression by steerable filters and multilevel inhibition with application to contour detection," *Pattern Recognition*, vol. 44, no. 9, pp. 1999–2007, 2011.
- [12] G. Papari and N. Petkov, "Edge and line oriented contour detection: state of the art," *Image and Vision Computing*, vol. 29, no. 2-3, pp. 79–103, 2011.
- [13] S. K. Pal and R. A. King, "On edge detection of x-ray images using fuzzy sets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, no. 1, pp. 69–77, 1983.
- [14] C. W. Tao, W. E. Thompson, and J. S. Taur, "A fuzzy if-then approach to edge detection," in *Proceedings of the 2nd IEEE International Conference on Fuzzy Systems*, vol. 2, pp. 1356–1360, San Francisco, Calif, USA, 1993.
- [15] F. Russo and I. G. Ramponi, "Edge extraction by fire operators," in *Proceedings of the IEEE International Conference on Fuzzy Systems*, pp. 249–253, Orlando, Fla, USA, June-July 1994.
- [16] F. Russo, "Edge detection in noisy images using fuzzy reasoning," *IEEE Transactions on Instrumentation and Measurement*, vol. 47, no. 5, pp. 1102–1105, 1998.
- [17] L. R. Liang and C. G. Looney, "Competitive fuzzy edge detection," *Applied Soft Computing Journal*, vol. 3, no. 2, pp. 123–137, 2003.
- [18] M. Hanmandlu, J. See, and S. Vasikarla, "Fuzzy edge detector using entropy optimization," in *Proceedings of the International Conference on Information Technology: Coding Computing (ITCC '04)*, pp. 665–670, Las Vegas, Nev, USA, April 2004.
- [19] R. R. Rakesh, P. Chaudhuri, and C. A. Murthy, "Thresholding in edge detection: a statistical approach," *IEEE Transactions on Image Processing*, vol. 13, no. 7, pp. 927–936, 2004.
- [20] O. Mendoza, P. Melin, and G. Licea, "A new method for edge detection in image processing using interval type-2 fuzzy logic," in *Proceedings of the IEEE International Conference on Granular Computing (GrC '07)*, pp. 151–156, San Jose, Calif, USA, November 2007.
- [21] J. Wu, Z. Yin, and Y. Xiong, "The fast multilevel fuzzy edge detection of blurry images," *IEEE Signal Processing Letters*, vol. 14, no. 5, pp. 344–347, 2007.
- [22] L. Hu, H. D. Cheng, and M. Zhang, "A high performance edge detector based on fuzzy inference rules," *Information Sciences*, vol. 177, no. 21, pp. 4768–4784, 2007.
- [23] Y. Yang, "An adaptive fuzzy-based edge detection algorithm," in *Proceedings of the International Symposium Intelligent on Signal Processing and Communication Systems*, pp. 276–279, Xiamen, China, November-December 2007.
- [24] V. K. Madasu and S. Vasikarla, "Fuzzy edge detection in biometric systems," in *Proceedings of the 36th Applied Imagery Pattern Recognition Workshop*, pp. 139–144, Washington, DC, USA, October 2007.
- [25] A. Alshennawy and A. Aly, "Edge detection in digital images using fuzzy logic technique," *World Academy of Science, Engineering and Technology*, vol. 27, pp. 178–186, 2009.
- [26] L. Zhang, M. Xiao, J. Ma, and H. Song, "Edge detection by adaptive neuro-fuzzy inference system," in *Proceedings of the 2nd International Congress on Image and Signal Processing (CISP '09)*, Tianjin, China, October 2009.
- [27] P. Melin, O. Mendoza, and O. Castillo, "An improved method for edge detection based on interval type-2 fuzzy logic," *Expert Systems with Applications*, vol. 37, no. 12, pp. 8527–8535, 2010.
- [28] M. Yoshimura and S. Oe, "Edge detection of texture image using genetic algorithms," in *Proceedings of the 36th SICE Annual Conference, International Session Papers (SICE '97)*, pp. 1261–1266, Tokushima, Japan, July 1997.
- [29] G. Sun, Q. Liu, C. Ji, and X. Li, "A novel approach for edge detection based on the theory of universal gravity," *Pattern Recognition*, vol. 40, no. 10, pp. 2766–2775, 2007.
- [30] N. Khalid, M. Manaf, and M. Aziz, "Fusion of fuzzy heuristic and particle swarm optimization as an edge detector," in *Proceedings of the International Conference on Information Retrieval Knowledge Management*, pp. 250–254, Shah Alam, Malaysia, March 2010.
- [31] O. P. Verma, P. Singhal, S. Garg, and D. S. Chauhan, "Edge detection using adaptive thresholding and ant colony optimization," in *Proceedings of the World Congress on Information and Communication Technologies (WICT '11)*, pp. 313–318, December 2011.
- [32] O. P. Verma, M. Hanmandlu, P. Kumar, S. Chhabra, and A. Jindal, "A novel bacterial foraging technique for edge detection," *Pattern Recognition Letters*, vol. 32, no. 8, pp. 1187–1196, 2011.
- [33] O. P. Verma, M. Hanmandlu, A. Sultania, and A. S. Parihar, "A novel fuzzy system for edge detection in noisy image using bacterial foraging," *Multidimensional Systems and Signal Processing*, vol. 24, no. 1, pp. 181–198, 2013.
- [34] P. Kaur and R. Maini, "BFO based multithresholding edge detection technique," *International Journal of Engineering Research and Applications*, vol. 3, no. 4, pp. 1872–1880, 2013.
- [35] A. Khunteta and D. Ghosh, "Edge detection via fuzzy rule-based edge strength estimation and optimal threshold selection using PSO," in *Proceedings of the IEEE International Conference on Industrial and Information Systems*, Kandy, Sri Lanka, December 2013.
- [36] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Perth, Australia, December 1995.
- [37] T. J. Ross, *Fuzzy Logic with Engineering Applications*, John Wiley & Sons, 3rd edition, 2010.
- [38] T. Pun, "Entropic thresholding: a new approach," *Computer Graphics and Image Processing*, vol. 16, no. 3, pp. 210–239, 1981.
- [39] J. N. Kapur, P. K. Sahoo, and A. K. C. Wong, "A new method for gray-level picture thresholding using the entropy of the histogram," *Computer Vision, Graphics, & Image Processing*, vol. 29, no. 3, pp. 273–285, 1985.
- [40] P. V. Henstock and D. M. Chelberg, "Automatic gradient threshold determination for edge detection," *IEEE Transactions on Image Processing*, vol. 5, no. 5, pp. 784–787, 1996.
- [41] M. H. F. Wilkinson, "Optimizing edge detectors for robust automatic threshold selection: coping with edge curvature and

- noise,” *Graphical Models and Image Processing*, vol. 60, no. 5, pp. 385–401, 1998.
- [42] M. Zhao, A. Fu, and H. Yan, “A technique of three-level thresholding based on probability partition and fuzzy 3-partition,” *IEEE Transactions on Fuzzy Systems*, vol. 9, no. 3, pp. 469–479, 2001.
- [43] S. E. El-Khamy, I. Ghaleb, and N. A. El-Yamany, “Fuzzy edge detection with minimum fuzzy entropy criterion,” in *Proceedings of the 11th IEEE Mediterranean Electronical Conference*, pp. 498–503, Cairo, Egypt, May 2002.
- [44] R. Koren and Y. Yitzhaky, “Automatic selection of edge detector parameters based on spatial and statistical measures,” *Computer Vision and Image Understanding*, vol. 102, no. 2, pp. 204–213, 2006.
- [45] S. Wan, F. Yang, and M. He, “Gradient-threshold edge detection based on perceptually adaptive threshold selection,” in *Proceedings of the 3rd IEEE Conference on Industrial Electronics and Applications (ICIEA '08)*, pp. 999–1002, Singapore, June 2008.
- [46] A. G. N. Nain, G. Jindal, A. Garg, and A. Jain, “Dynamic thresholding based edge detection,” in *Proceedings of the World Congress on Engineering (WCE '08)*, vol. 1, London, UK, July 2008.
- [47] Q. Chen, Q. S. Sun, P. A. Heng, and D. S. Xia, “A double-threshold image binarization method based on edge detector,” *Pattern Recognition*, vol. 41, no. 4, pp. 1254–1267, 2008.
- [48] J. Zhang, Y. Lian, L. Dong, X. Zhao, and J. Liu, “A new method of fuzzy edge detection based on gauss function,” in *Proceedings of the 2nd International Conference on Computer and Automation Engineering (ICCAE '10)*, pp. 559–562, Singapore, February 2010.
- [49] Y. Li and Z. Gao, “Edge detection based on fuzzy 2-partition entropy approach,” in *Proceedings of the 3rd IEEE International Conference on Advanced Computer Control (ICACC '11)*, pp. 616–619, Harbin, China, January 2011.
- [50] M. El-Sayed, “A new algorithm based entropic threshold for edge detection in images,” *International Journal of Computer Science Issues*, vol. 8, no. 5, pp. 71–78, 2011.
- [51] M. Thakkar and H. Shah, “Edge detection techniques using fuzzy thresholding,” in *Proceedings of the World Congress on Information and Communication Technologies (WICT '11)*, pp. 307–312, Mumbai, India, December 2011.
- [52] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2001.
- [53] Contour image database, http://www.cs.rug.nl/imaging/databases/contour_database/contour_database.html.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

