

An Automatic Thresholding Approach to Gravitation-Based Edge Detection in Grey-Scale Images

Kimia Rezaei

Department of Electrical and Electronic Engineering, University College Cork, Ireland
kimia_Rezaei@yahoo.com

Hamed Agahi*

Department of Electrical Engineering, Shiraz Branch, Islamic Azad University, Shiraz, Iran.
agahi@iaushiraz.ac.ir

Received: 14/Nov/2020

Revised: 25/Jul/2021

Accepted: 25/Aug/2021

Abstract

This paper presents an optimal auto-thresholding approach for the gravitational edge detection method in grey-scale images. The goal of this approach is to enhance the performance measures of the edge detector in clean and noisy conditions. To this aim, an optimal threshold is automatically found, according to which the proposed method dichotomizes the pixels to the edges and non-edges. First, some pre-processing operations are applied to the image. Then, the vector sum of the gravitational forces applied to each pixel by its neighbors is computed according to the universal law of gravitation. Afterwards, the force magnitude is mapped to a new characteristic called the force feature. Following this, the histogram representation of this feature is determined, for which an optimal threshold is aimed to be discovered. Three thresholding techniques are proposed, two of which contain iterative processes. The parameters of the formulation used in these techniques are adjusted by means of the metaheuristic grasshopper optimization algorithm. To evaluate the proposed system, two standard databases were used and multiple qualitative and quantitative measures were utilized. The results confirmed that the methodology of our work outperformed some conventional and recent detectors, achieving the average precision of 0.894 on the BSDS500 dataset. Moreover, the outputs had high similarity to the ideal edge maps.

Keywords: Auto-thresholding; edge detection; force feature; the law of universal gravity; the grasshopper optimization algorithm.

1- Introduction

Edge detection is a fundamental step in the image-understanding, which aims to locate the edge points in an image. An edge, is a point at which image brightness changes sharply. In other words, edges are boundary pixels connecting two sections with different amplitude attributes. Edges play a key role in image processing field and contain important information of objects in images [1]. Via edge detection, it is possible to find the boundaries and separate objects from the background. Successful edge detection can valuably simplify the analysis of the information contents in high-level processing tasks such as image segmentation, feature extraction and object identification [2]. However, correct edge recognition in images, depends on using the appropriate edge detector in the existing condition. Therefore, it is required to create an algorithm which discovers most accurate edges.

Several edge detectors have been proposed by researchers. Some of the earliest works include the methods of Sobel [3], Prewitt [4], Canny [5], Roberts [6] and the Laplacian of Gaussian (LoG) [7]. Romani et al. [8]

used an iterative method based on interpolation with variably scaled kernels for edge detection. Eser and Derya [9] designed a method based on neutrosophy set using maximum norm entropy for the edge detection. To this aim, they made use of various entropy types. Yuesheng and Lionel [10] proposed an edge detection system based on the physical law of diffusion. Their algorithm dealt with the edge detection as a character of an energy diffusion in media space. Finally, the energy information could be extracted. Bhogal and Agrawal [11] evaluated three algorithms for the image edge detection. They used Sobel, type-1 and interval type-2 fuzzy logic detectors. Their experimental result demonstrated that type-2 fuzzy edge detector achieved better output compared with the other two. Banharnsakun [12] proposed an enhancing edge detection technique via artificial bee colony (ABC) algorithm. In this method, first, an optimal edge filter was found; finally, the threshold value was optimized. Verma and Parihar [13] used a fuzzy system for the edge detection. They applied the 'Uni-value Segment Assimilating Nucleus' (USAN) area calculation. Then, the edge map was fuzzified and the bacterial foraging algorithm (BFA) was used for the membership functions

* Corresponding Author

optimization. Ansari et al. [14] developed a new technique based on intuitionistic fuzzy divergence and entropy measures for the edge detection problem. Then, the results were compared to some other methods using mean square error (MSE) and peak signal-to-noise ratio (PSNR) measures. Shen et al. [15], the authors utilized some hierarchical features learned by Convolutional Neural Networks (CNN) to attain multi-scale representations. Xie et al. [16] introduced an end-to-end detection model that leveraged the outcomes from different intermediate layers with skip-connections. Xu et al. [17] presented a hierarchical deep model to compute multi-scale features and a gated conditional random field (CRF) to combine them.

An effective approach to handle some complex engineering problems is to utilize nature-inspired algorithms. Newton's law of universal gravitation [18] is one of the typical physical models, based on which, several image-processing algorithms have been proposed for applications including segmentation [19], multi-level thresholding [20] and color texture classification [21]. The edge detection method we present in this note is based on the original use of gravitational forces by Sun et al. [22]. We will refer to this method as the gravitational approach. In this approach every pixel is considered as a celestial body, which has relationships with other neighboring pixels in the image. Afterward, the vector sum of all gravitational forces of a pixel applied on by pixels in a limited neighborhood is computed. Finally, a threshold for the force magnitude is considered, due to which pixels are separated to edges and non-edges. Several variations of this method are presented in the literature, due to the advantages of the gravity field. Lopez-Molina et al. [23] analyzed the effect of the substitution of the product operation by other triangular norms when computing the gravitational forces. They treated edge points as fuzzy sets and the membership degrees were computed from the resulting gravitational force on each pixel. Verma and Sharma [24] presented an approach for the edge detection using the universal law of gravity and ant colony optimization (ACO). They computed a heuristic function using law of universal gravity as a way to food sources for the ants to detect the edge pixels. Verma et al. [25] proposed an edge detector which combined the universal law of gravity and the gravitational search algorithm (GSA). They treated the edges as masses for which using the law of universal gravity and the GSA, movement of edges were computed. Deregeh and Nezamabadi-Pour [26] assumed image pixels to be celestial objects. Further, the edge points were found using a number of moving agents in the image space. The movement of agents were determined via the forces of objects located in their neighborhood region. Sun et al. [27] proposed an edge detector for hyperspectral images (HSI) using the gravitational theory. Each spatial-spectral vector pixel in

an HSI was considered as a celestial body to which forces were exerted by its neighboring pixels. Thus, each body traveled until it reaches a stable equilibrium. Finally, the edges were distinguished by computing the gravitational potential energy.

Wang et al. [28] proposed an edge detector based on the NAGD and MGMF methods. They decomposed each color image into six components in the RGB and HSV model and found the gradient amplitude of the image edge by the CLAHE. Then they constructed an NAGD to extract the edge map of the color image and finally, used the SVD to fuse each channel component to improve the accuracy of the edge detection. Hi et al. [29] proposed a bi-directional cascade network for the edge detection of objects at different scales. In this method, a layer is supervised by labeled edges at its scale, rather than applying the supervision to the network outputs. To enrich the multi-scale representations, a module used the dilated convolution to generate multi-scale features. Li et al. [30] proposed a novel technique to resolve the balance between the fast training and accurate testing aspects of multi-scale representation in the edge detection task. According to multi-stream structures and the image pyramid principle, they constructed two pyramid networks to enrich the multi-scale representation. Then using a backbone network, the overall bi-directional pyramid network (BDP-Net) architecture was constructed. Lu et al. [31] proposed a vector co-occurrence morphological operator for the edge detection, which considers both the pixel and boundary information. This method resists the influence of the noise points and detects the edges from the color image rather than the gray image. Anand and Sangeethapriya [32] presented an isotropic Gaussian modulated hyperbolic tangent high-pass filter to remove the high frequency components in noisy images, as a pre-process in the edge detection task. Their filter achieved better directional selectivity and offered less noise sensitivity along with regularization by the least square error design.

The main focus of this paper is to find an optimal threshold for the force magnitudes, according to which, the pixels are dichotomized to the edge and non-edge classes. First, a normalizing map is applied to the intensity values from the domain $[0,255]$ to the range $(0,1)$. Through applying this mapping, even zero-intensity pixels could be considered as edges. Afterwards, similar to the original work, the gravitational force exerted by neighboring pixels on each pixel is computed. Following this, a fuzzy membership function is applied to the overall force magnitude in order to control the uncertainties due to discretization and noises in the original image. The output of this step is called the *force feature*, for which an optimal threshold is desired to be found. Auto-thresholding is the main novelty of this paper. To this aim, the histogram of the force feature is computed. Then, an iterative process is applied to tune the threshold, starting from the mean value

of the histogram. Once the threshold value is converged, its final value is chosen as the optimal threshold for the force feature. Selecting the initial point for the iterative process has a substantial effect on the speed of convergence and accuracy of the classification. Thus, we present a new formulation for $T_{initial}$ which contains the mean and the standard deviation of the force feature distribution together with two constant regulating parameters. This formulation includes both information from the image under study and general information provided from all the images in the dataset, hidden in the constant parameters. To find the optimal values of these parameters and one parameter in the fuzzy membership function of the previous step, we perform a metaheuristic optimization algorithm. For this goal, the grasshopper optimization algorithm (GOA), proposed by Saremi et al. in 2017 [33] is chosen. This algorithm is inspired from the life of grasshoppers in large swarms and their effort for finding new sources of food. This algorithm works well in both the exploration and exploitation phases in complex uni-modal and multi-modal problems and accurately discovers the global optimum solution. This solution yields the optimal values of the required parameters.

The proposed algorithm is tested on two standard image datasets with and without the contamination of noise. The first one is the USC-SIPI and the second dataset is the BSDS500. The results are compared with the ideal edge maps to evaluate the performance of the system. Some qualitative and quantitative measures are employed for this purpose, including visual comparison, Pratt's figure of merit, Shannon Entropy and the Average Precision. Comparisons are also performed with some conventional edge detectors and also some recent methods from the literature on the mentioned datasets, to show the superiority of the proposed approach. Four cases are considered according to the noise type and density, comprising of clean image, two Gaussian noise with different variances and the salt and pepper noise. The results are also compared with some conventional edge detectors. Moreover, four scenarios are studied for the thresholding phase, containing: (i) the Otsu method, (ii) auto-thresholding starting from T_{mean} , (iii) direct finding the optimal threshold using the GOA, and (iv) auto-thresholding initiating from the formulation of $T_{ini.opt.}$ in (16) whose parameters are determined by the GOA. The latter three scenarios are the proposed thresholding approaches of this paper. The fourth scenario shows to be competitive, obtaining better results than different published methods and also other proposed scenarios.

The remainder of this paper is structured as follows. Section 2 describes the background of the Newton's law of gravity. The proposed approach is presented in Section 3. In this section, first the basic gravitational edge detector is reviewed. Then the proposed thresholding methods are introduced. At last, the section is extended by laying out

the steps of computing an optimal threshold value by the GOA. Section 4 is devoted to the evaluating the proposed method and comparing it with some others, while applying to two standard databases. In the end, Section 5 presents a brief summary and includes a conclusion and future lines of research into this area.

2- Background of the Universal Law of Gravitation

Based on the Newton's law of universal gravitation [18], objects attract and being attracted by other nearby objects. The mutual attractive force is directly proportional to the product of the masses and inversely to the square of their distance, as illustrated in Fig. 1. The mathematical formula describing the gravitational force can be written as follows:

$$\vec{f}_{1,2} = \frac{G m_1 m_2 \vec{r}}{\|\vec{r}\|^3} \quad (1)$$

Where, $\vec{f}_{1,2}$ represents the force vector applied by mass #2 on mass #1. The universal gravitational constant is represented by G while the objects gravitational masses are shown by m_1 and m_2 respectively. The mass is a measure of the strength of the gravitational field on a particular object. The gravitational field of a body has a direct proportional relation with its gravitational mass. Also \vec{r} stands for the distance vector connecting the centers of the masses.

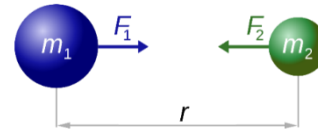


Fig. 1 Model of Newton's law of universal gravitation for masses m_1 and m_2

The gravitational force obeys the superposition principle. The force exerted on object k by its neighbor objects in the region Ω is found by the vector sum of forces exerted by each object.

$$\vec{F}_k = \sum_{i \in \Omega} \vec{f}_{k,i} \quad (2)$$

3- The Auto-thresholded Gravity-based Edge Detection

3-1- Overview

Based on the universal law of gravitation, several edge detection systems are presented in the literature. In this section, the proposed approach for auto-thresholding the edge detection system based on the gravitational approach is presented. The general platform of our approach consists of two phases, as Fig. 2 shows. In the first phase,

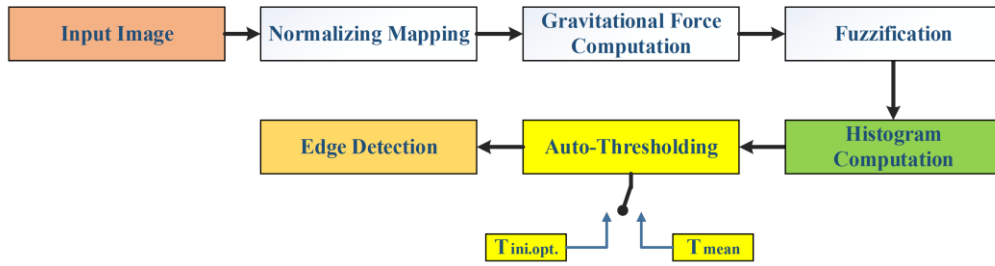


Fig. 2 The block diagram of the proposed edge detection system

the gravitational edge detector is generated to determine the force exerted on each pixel. Then, the force magnitude is mapped to the force feature using a fuzzy membership function; and the histogram of this feature for the whole image is created. In the second phase, an optimal threshold is found for that histogram with respect to which, the pixels are classified to edges and non-edges. The next subsections describe the details of these steps.

3-2- Generation of the Gravitational Edge Detector

In order to construct an edge detector, it is supposed that every pixel in the image is a celestial body, which affects and being affected by other neighboring pixels through gravitational forces [22]. For each pixel, the forces of pixels beyond a pre-specified distance are assumed to be zero. In other words, far-away points do mutually not exert nor receive any forces. The magnitude and the direction of the vector sum of all gravitational forces that each pixel receives from its neighbors include significant information about the intensity changes and the existence of an edge in that point.

To start with, we denote a pixel at location (i, j) by $x_{i,j}$ and its gray level intensity value by $I(i, j)$. As the next relations will show, the force exerted on zero-intensity pixels will be zero. Thus, they will never be considered as edge points, even when their neighbors' intensities change sharply. To avoid this problem, a mapping of intensities from the original domain to the range $(0,1]$ is implemented. Suppose that the maximum intensity value in the image is I_M . The following operation are performed with $\eta \ll 1$ [23].

$$G(i, j) = I(i, j)/I_M, \quad g(i, j) = \frac{G(i, j) + \eta}{1 + \eta} \quad (3)$$

This mapping enables the gravitational edge detector to identify the possible edge pixels with zero intensities. The edge detection technique using the law of universal gravitation is described below [22]:

- i. An $m \times n$ neighborhood Ω with pixels $x_{k,l} \in \Omega$ and $(k, l) \neq (i, j)$ is considered for each image point $x_{i,j}$. With respect to (1), the gravitational force exerted by each neighbor $x_{k,l}$ on $x_{i,j}$ is computed as follows:

$$\vec{f}_{i,j;k,l} = \frac{G m_{i,j} m_{k,l} \vec{r}}{\|\vec{r}\|^3} \quad (4)$$

Here, $\vec{f}_{i,j;k,l}$ is the applied gravity force and $m_{i,j}$ and $m_{k,l}$ are the mapped intensity values of pixels, i.e., $g(i, j)$ and $g(k, l)$. Vector \vec{r} , represents the distance vector from $x_{k,l}$ to $x_{i,j}$, whose magnitude is computed by (4):

$$\|\vec{r}\| = \sqrt{(k-i)^2 + (l-j)^2} \quad (5)$$

- ii. Considering an image as a 2D plane, the vectors of the gravitational forces in the horizontal and vertical directions (i.e., \hat{x} and \hat{y}) can be computed as below:

$$\begin{aligned} f_{i,j;k,l}^x &= |\vec{f}_{i,j;k,l}| \sin \theta = \frac{G m_{i,j} m_{k,l} (k-i)}{\|\vec{r}\|^2 \|\vec{r}\|} = \frac{G m_{i,j} m_{k,l} (k-i)}{\|\vec{r}\|^3}, \\ f_{i,j;k,l}^y &= |\vec{f}_{i,j;k,l}| \cos \theta = \frac{G m_{i,j} m_{k,l} (l-j)}{\|\vec{r}\|^2 \|\vec{r}\|} = \frac{G m_{i,j} m_{k,l} (l-j)}{\|\vec{r}\|^3} \end{aligned} \quad (6)$$

Hence, the vector $\vec{f}_{i,j;k,l}$ is represented as below:

$$\vec{f}_{i,j;k,l} = f_{i,j;k,l}^x \hat{x} + f_{i,j;k,l}^y \hat{y} \quad (7)$$

The vector sum of all gravitational forces applied by neighbors on $x_{i,j}$ is demonstrated as:

$$\vec{F}_{i,j} = \sum_{\Omega} F^x \hat{x} + F^y \hat{y}, \quad (k, l) \in \Omega \ \& \ (k, l) \neq (i, j) \quad (8)$$

Where,

$$F^x = \sum f_{i,j;k,l}^x, \quad F^y = \sum f_{i,j;k,l}^y, \quad (k, l) \in \Omega \ \& \ (k, l) \neq (i, j) \quad (9)$$

- iii. The magnitude and direction of the vector $\vec{F}_{i,j}$ are determined as in (10):

$$F = \sqrt{F_x^2 + F_y^2}, \quad \theta = \tan^{-1}\left(\frac{F_x}{F_y}\right) \quad (10)$$

- iv. Finally, the edge map is produced by setting an appropriate threshold on the force magnitude histogram.

For simplicity, $G \cdot m_{i,j}$ is replaced with a constant C . Base on experiments, we set $C = 1$ in clean images and $C = sig(g(i, j))$ in noisy images, where $sig(\cdot)$ is a sigmoid function. The effect of noise on edge detection may also be reduced if more neighboring pixels are considered. In this paper, the neighborhood is represented by a square of size 5×5 .

Digital images have some inherent uncertainties due to discretization and noise. Hence, the fuzzy logic is proved to be an appropriate tool for representing and processing

of the edges. Since the magnitude of the force characterizes the difference in intensities of the neighboring points, we generate a membership function to map this feature to a new one which will be used for finding the edge image. In order to do so, the Gaussian membership function is considered to act on the force magnitude of each pixel [34]. The function M_G , given by (11) and shown in Fig. 3, has only a single parameter δ which represents the standard deviation of the distribution. The maximum of the force magnitude F is represented by γ . We mapped the force magnitude from $[0, \gamma]$ to $[0, 1]$, so that, $M_G(0) = 0$ and $M_G(\gamma) = 1$. Hence, the function is: $q = M_G(F) = \exp(-(\ln(F) - \ln(\gamma))^2 / 2\delta^2)$, (11) We call q the ‘force feature’. This operation can improve the results of the edge detection method. The parameter δ is one of the three parameters which will be adjusted through an optimization technique in next steps.

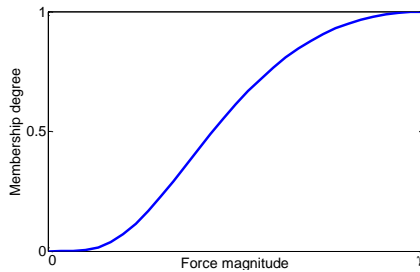


Fig. 3 Gaussian membership function for the conversion of force magnitudes into membership degrees

3-3- Thresholding

Thresholding is a non-linear operation used for edge detection and segmentation [35]. This operation converts a grayscale image into a binary image in which, two levels are assigned to pixels that are below or above a specified threshold value. Selecting an optimal threshold value is the most important step in every image thresholding algorithm. This paper mainly focuses on computing the optimal value for the threshold by which, the edge points of the images are well distinguished. Using this method, finding weak edges of noisy and clean images will be attainable. First, the histogram of distribution of the force feature q is determined. The goal is to find an optimal threshold for this histogram. Those pixels whose force feature is above this threshold are classified as edges, while the rest will be non-edges. To do this, an iterative approach based on the averages of force features is used as follow:

1. Compute the average force feature of all pixels in the image, to be used as the initial threshold value:

$$T = T_{initial} = T_{mean} = \text{mean}(q) \quad (12)$$

2. Dichotomize the histogram according to the threshold T :

$$Q_1 = \{q | q \leq T\}, \quad Q_2 = \{q | q > T\}, \quad (13)$$

3. Compute the means of Q_1 and Q_2 :

$$M_1 = \text{mean}(Q_1), M_2 = \text{mean}(Q_2) \quad (14)$$

4. The new threshold value is obtained by computing mean of M_1 and M_2 :

$$T_{new} = 1/2 (M_1 + M_2) \quad (15)$$

5. If $T_{new} - T < \epsilon$, stop and choose T_{new} as the optimal threshold value. Otherwise, set $T = T_{new}$ and go to step 2.

As it will be seen in the next section, this approach yields accurate results, both in clean and noisy images with higher indices, compared with other existing methods. However, we tried to still improve our technique by finding an appropriate initial threshold value for the first step. Smart selection of $T_{initial}$ helps in the fast convergence of the iterative process and more robustness against different noises, as the simulation results validate in the simulation section. Hence, we redefine $T_{initial}$ as in (16).

$$T_{initial} = T_{ini.opt.} = \frac{\mu + \alpha \cdot \sigma}{\mu + \beta \cdot \sigma} \quad (16)$$

In which, μ and σ represent the mean and the standard deviation of q respectively, and α and β are two regulating parameters. It should be noted that, $T_{initial}$ in (16) is not fixed for all images, but it depends on some constants α and β together with two characteristics of the image under study (i.e., μ and σ). In order to find α , β and also δ in (11), we make use of an optimization technique. The grasshopper optimization algorithm, reviewed in the next sub-section, is run on several images and their ideal edge maps to find the optimal values of these parameters. The position of each grasshopper is represented by a vector of length three in the form $x = [\alpha \ \beta \ \delta]^T$. Figure of Merit (FOM), defined in (24), is used as the objective function for parameter optimization. The optimization problem can be stated as maximizing FOM subjected to $\alpha, \beta, \delta > 0$. Once the optimal valued of these parameters are found, $T_{initial}$ in (16) will be used as the initial point of the iterative process (12-15) for the test images.

3-4- Overview of the Grasshopper Optimization Algorithm

The grasshopper optimization algorithm (GOA), proposed by Saremi et al. in 2017 [33], is a novel metaheuristic optimization algorithm that models and mimics the swarm behavior of grasshoppers. As notorious insects for damaging the farms, grasshoppers move as large swarms to seek the food sources. In the nature-inspired GOA, grasshoppers can globally search the given space, discover the regions of answers and move locally in the final steps in order to find the optimal solutions. GOA, models the mutual forces exerted by the grasshoppers for the repulsion and attraction. While the repulsion forces encourage the grasshoppers to explore the search space, the attraction forces allow them to exploit the local regions. Both exploitation and exploration of this

algorithm are satisfactory for uni- and multi-modal test functions and these two tendencies are balanced in the GOA. The objective function is improved over the course of iterations to provide a more accurate approximation of the global optimum solution. This algorithm has been applied to many engineering problems and shown a remarkable performance in many fields, such as feature selection [36], vibration analysis [37] and more. The movement of a grasshopper is subjected to three factors: (i) social interaction, (ii) gravity force and (iii) wind advection. In general, the position of the i -th grasshopper in a swarm can be modeled as follows:

$$X_i = S_i + G_i + A_i \quad (17)$$

In which, S_i , G_i and A_i represent the social interaction, the gravity force and the wind advection for the grasshopper, respectively. The social interaction for the i -th grasshopper is defined as follows:

$$S_i = \sum_{j=1, j \neq i}^N s(d_{ij}) \hat{d}_{ij} \quad (18)$$

In which, d_{ij} is the Euclidian distance between the two grasshoppers i and j . Also, $\hat{d}_{ij} = (x_j - x_i)/d_{ij}$ is the unit vector directing from the location of the i -th grasshopper (x_i) to that of the j -th one (x_j). The number of grasshoppers is shown by N . The s function indicates the social force, as defined in (19).

$$s(r) = f \cdot \exp(-r/ls) - \exp(-r) \quad (19)$$

Where, f is the attraction intensity and ls represents the attractive length scale. Additionally, the gravity component is computed as follows:

$$G_i = -g \cdot \hat{e}_g \quad (20)$$

In (20), the parameter g is the gravity constant and \hat{e}_g is the unit vector towards the center of the earth. As well, the wind advection component is defined in (21).

$$A_i = u \cdot \hat{e}_w \quad (21)$$

Where, u is the constant drift and \hat{e}_w is the unit vector which represents the wind direction. To solve optimization problems in D dimension, an improved version of (17) was presented in [33]:

$$X_i^d = c \left(\sum_{j=1, j \neq i}^N c \frac{ub_d - lb_d}{2} s(|x_j^d - x_i^d|) \frac{(x_j^d - x_i^d)}{d_{ij}} \right) + \hat{T}_d, \quad (22)$$

$$d = 1, \dots, D$$

In which, ub_d and lb_d are the upper and lower bounds in the d -th dimension and \hat{T}_d indicates the value of the d -th dimension in the best solution found so far. In (22), the gravity coefficient is ignored and the wind advection component is set equal to \hat{T}_d . Additionally, c is a multiplying coefficient which decreases the exploration while increases the exploitation proportional to the number of iterations:

$$c = c_{max} - l \frac{c_{max} - c_{min}}{L} \quad (23)$$

Where c_{max} and c_{min} indicate the maximum and minimum values. Also l and L are the current and

maximum number of iterations, respectively. In this paper, we set: $f = 0.5$, $ls = 1.5$, $c_{max} = 1$ and $c_{min} = 1e - 5$.

4- Experimental Results

This section compares the performance results of the proposed edge detection system in different scenarios to those produced by some classic and some recent methods from the literature. First, the datasets for evaluating the methods are introduced, then the methodology of the experiments is described. Here, both qualitative and quantitative measures are utilized. Finally, the results are given and compared for clean and noisy images.

4-1- The Datasets

In this experiment, two datasets are used for evaluating the methods. The first one is the USC-SIPI (Signal and Image Processing Institute, USC) [38] which offers a collection of digitized images with different sizes. In this paper, the images of size 256×256 are used (Fig. 4a-b). The second dataset is the BSDS500 (Berkeley Segmentation Dataset, Computer Vision group, UCB) [39]; which consists of 500 natural images and ground-truth annotations (Fig. 4c-e). Each image in this dataset is manually annotated by multiple annotators; whose outcomes are averaged to give the final ground-truth for that image. The final annotations are used as the ideal edge maps, which are solutions for the edge detection problem. Moreover, the results for the cameraman image (Fig. 4f) are also given. The 10-fold cross-validation method on the datasets is used for evaluating the methods.

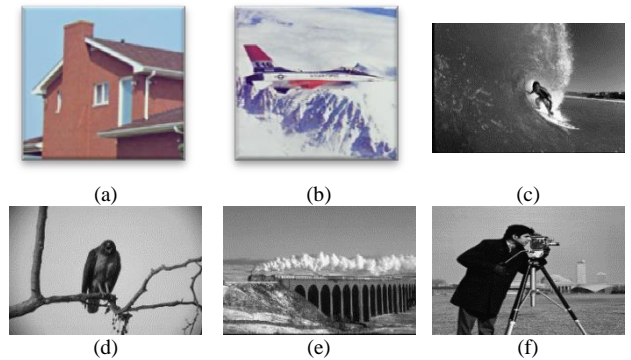


Fig. 4 Samples of the datasets images. (a-b) USC-SIPI, (c-e) BSDS500, (f) the cameraman

4-2- The Measures

In this section, by the *proposed method* we mean computing the histogram of the feature force and applying the auto-thresholding iterative process initiating from $T_{ini.opt}$. whose optimal parameters are found via the GOA. To evaluate the proposed method applied to the mentioned

datasets, two sets of measures are used. The first set contains the qualitative measure, while the second one comprises the quantitative measures. Four experimental cases were considered according to the noise type and density. The first case contains clean images while, in Cases II-III, images are corrupted with two zero-mean Gaussian noises with $var = 0.01$ and 0.05 , respectively. In the latter cases, a 5×5 mask is used to decrease the Gaussian noise effect. Also, the images in Case IV are contaminated with the salt and pepper noise with density of 0.05 . In this case, a median filter is applied to the images in pre-processing step, in order to moderate the effect of the noise and enhance the performance. Besides, notice that while generating the gravitational edge detector, we set $C = 1$ in clean images and $C = sig(g(i, j))$ in noisy images.

4-2-1- The qualitative Measure

The most common and simplest way of evaluating an edge detector is the visual comparison of output images which could be done in a range of different ways. For instance, image gradients can be compared visually where an edge image is assessed by a group of individuals and the average score may serve as the quality index. Fig. 5

shows the output images of various classic detectors with *clean image* input (Case I). It is obvious that, most of the edge structures are well preserved by the proposed detector compared with others such as Canny, Sobel or Robert. Therefore, the proposed technique is more efficient than others for the edge detection of these images. When noise is added to the image, finding an optimal threshold value becomes more important, since the aims are to discover true edges and ignore those made due to the noise. Results on images corrupted with Gaussian noise (Case II) is shown in Fig. 6. It should be noted that, we used Case III only in evaluating the system by quantitative measures. Visual comparison clearly reveals the superiority of the proposed approach compared to the others. Not only sharp edges are successfully found, but also more artefacts and irrelevant information are removed using the proposed detector. The results of Case IV are shown in Fig. 7. According to the visual differences, it goes without saying that, the noise density in the edge map resulted from the proposed method is considerably lower than that of the output images attained by other methods. Additionally, sharp edges are successfully found.

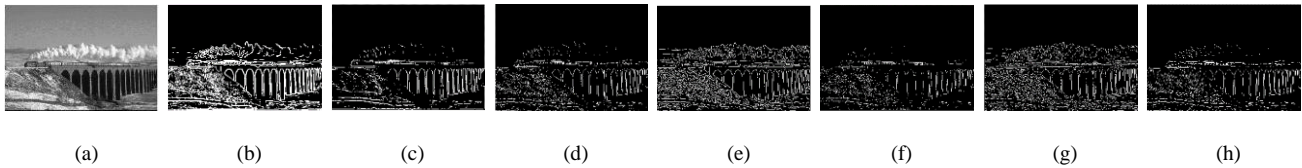


Fig. 5 Edge Detection of a clean image (Case I): (a) the original image, (b) the proposed edge detector, (c) Otsu using 5×5 mask, (d) Sobel, (e) Canny, (f) Robert, (g) LOG, (h) Prewitt.

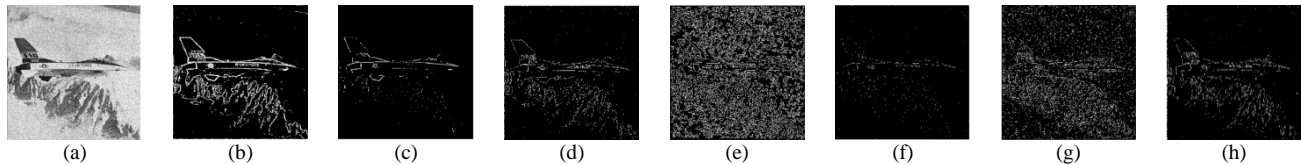


Fig. 6 Edge detection of images corrupted with Gaussian noise (Case II). The sub-figure captions are same as those in Fig. 5.

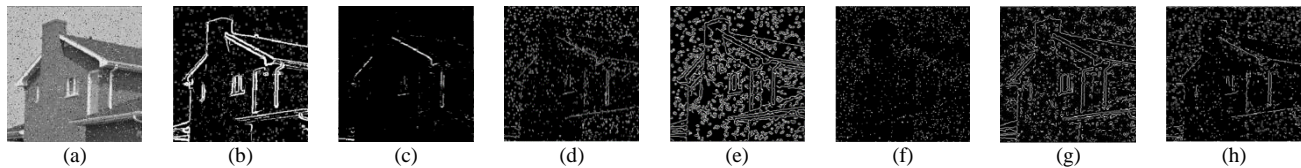


Fig. 7 Edge detection of images contaminated with the salt and pepper noise (Case IV). The sub-figure captions are same as those in Fig. 5.

4-2-2- The quantitative Measure

To perform quantitative comparison, we use three measures: the 'figure of merit' (FOM), the 'Shannon Entropy' and the Average Precision (AP). With respect to

these measures, a more accurate evaluation on the performance of the proposed and existing methods in different conditions is provided.

Figure of Merit (FOM)

Pratt's figure of merit is a popular measure for evaluating the performance of an edge detector system. The miss-detection errors can be classified as (i) missing valid edge pixels, (ii) detecting noises and artefacts as valid edges, and (iii) smearing of edges [40]. The Pratt's FOM is defined as Follows:

$$F = \frac{1}{\max(N_l, N_D)} \sum_{i=1}^{N_D} \frac{1}{1+\varepsilon d_i^2} \quad (24)$$

In (24), number of the edge pixels in the detected and ideal edge images are indicated by N_D and N_l respectively. The Euclidean distance between the location of the i -th detected edge pixel to the nearest edge pixel in the ideal edge image is indicated by d_i . Also ε is a scaling constant chosen to be $\frac{1}{9}$ based on the Pratt's work. F is an index in the range of $[0,1]$, which measures the accuracy of the edge localizations. The larger the value of F , the better the performance is. An edge detector with $F = 1$ is perfect. For some images in Fig. 4, the FOM of the proposed and some existing methods are reported in Table 1. The proposed algorithm found meaningful edges in cleanest images (Case I) in comparison with other algorithms. In fact, it kept the main objects in the images and provided the most similarity with the ideal edge maps. Also, the proposed method significantly decreased the effect of Gaussian noise in Case II; while the performances of other detectors were considerably lower. When the density of noise was increased (Case III), our method yielded higher similarity between detected and ideal edges compared with other detectors. Moreover, the performance of the proposed technique on images corrupted by salt and pepper noise (Case IV), was still higher compared to other detectors. However, it was less than those for Cases I-III. As a matter of fact, the salt and pepper noise existed in the output image beside the edge points, yet with a lower density compared with other methods.

The Shannon Entropy Measure

The Shannon entropy presents a measure for estimating the average of a random flux of information in an image I , which is formulated as follows [41]:

$$H(I) = - \sum_{k=0}^{K-1} p_k \cdot \log(p_k) \quad (25)$$

Where K is the number of gray levels (here, $K = 256$) and p_k is the probability associated with the gray level k . In general, high Shannon entropy means high disorder in the image data. To put it another way, the edge pixels which represent the discontinuity are better identified in an image with higher entropy. Therefore, Shannon entropy is an efficient method for evaluating the performance of an edge detector. However, a high value of Shannon entropy is not necessarily equal to rich information, due to the effect of uncontrolled noise on the entropy value. Tables 2-3 compared the performance of the proposed and some classic edge detectors in term of the Shannon entropy for all images in Fig. 4.

To attain a fair evaluation, the entropy and FOM measures should be considered at once. That is to say, if the entropies of two methods are close to each other, their FOMs should be compared in order to determine the more accurate one. For example, although the canny detector resulted in high entropy values, extra and undesirable edges were detected, leading to output image distortion. To illustrate, a visual comparison of the outputs of the canny and the proposed approaches is shown in Fig. 8. The former method detected some undesirable points as edges, while the latter approach attained a better outcome. For all clean images (except Fig. 4c), the entropy of the proposed method was higher than others which generally demonstrates its higher efficiency. Moreover, higher FOM results achieved by the proposed approach verified its effectiveness.

If the variations in the entropy of a system in clean and noisy conditions are small, that system is robust against the noise existence and the effect of noise is controlled in the system output. Due to the non-gradient structure of the gravitational edge detection method, such robustness is expected. Comparing the variations in Cases I-III of Tables 3-4 with the existence of Gaussian noises, it is obvious that, our approach not only has high entropy values (i.e., remaining the edges), but also reveals less entropy variations with respect to the noise-free condition, compared to other methods. Results in Case IV (image with salt and pepper noise), also show that the entropy variation is more acceptable than the other techniques.

Table 1: Comparison of the edge detection techniques based on the FOM measure. 'PrM.' stands for the 'Proposed Method'.

Detector	Case I			Case II			Case III			Case IV		
	Fig. 4c	Fig. 4d	Fig. 4e	Fig. 4c	Fig. 4d	Fig. 4e	Fig. 4c	Fig. 4d	Fig. 4e	Fig.4c	Fig.4d	Fig.4e
<i>Sobel</i>	0.3948	0.2843	0.3901	0.2951	0.2592	0.3010	0.2523	0.2394	0.2401	0.1735	0.3464	0.3691
<i>Canny</i>	0.2247	0.4091	0.7589	0.1053	0.1889	0.7463	0.1021	0.1818	0.7255	0.1270	0.2314	0.7286
<i>Roberts</i>	0.3604	0.3308	0.2754	0.1278	0.1688	0.0957	0.0619	0.0868	0.0475	0.1427	0.1902	0.1636
<i>LoG</i>	0.2516	0.3335	0.5895	0.1459	0.3678	0.6561	0.1277	0.2874	0.7285	0.1429	0.3063	0.7110
<i>Prewitt</i>	0.3947	0.2847	0.3857	0.2993	0.2618	0.3039	0.2543	0.2439	0.2432	0.1784	0.3544	0.3627
<i>PrM.</i>	0.5164	0.7663	0.8312	0.4547	0.6248	0.7718	0.4561	0.6781	0.7431	0.3319	0.5311	0.7992

Table 2: Comparison of the edge detection techniques based on the Shannon Entropy measure for images in Fig. 4a-c.

Detector	Case I			Case II			Case III			Case IV		
	Fig. 4a	Fig. 4b	Fig. 4c	Fig. 4a	Fig. 4b	Fig. 4c	Fig. 4a	Fig. 4b	Fig. 4c	Fig. 4a	Fig. 4b	Fig. 4c
<i>Sobel</i>	0.2198	0.2243	0.1943	0.1832	0.1894	0.1494	0.1645	0.1751	0.1488	0.2955	0.2828	0.2564
<i>Canny</i>	0.3842	0.4062	0.4664	0.7180	0.7136	0.8117	0.7696	0.7953	0.8393	0.4770	0.5111	0.5569
<i>Roberts</i>	0.2150	0.2190	0.1606	0.0700	0.0729	0.0648	0.0437	0.0468	0.0411	0.1898	0.1687	0.1643
<i>LoG</i>	0.2967	0.3197	0.3364	0.4291	0.4264	0.5612	0.5201	0.5080	0.6176	0.3459	0.3779	0.4412
<i>Prewitt</i>	0.2187	0.2236	0.1944	0.1857	0.1910	0.1505	0.1672	0.1764	0.1484	0.3215	0.3318	0.3084
<i>PrM.</i>	0.4914	0.4765	0.3918	0.4423	0.4542	0.4391	0.4835	0.4711	0.4516	0.5258	0.5958	0.4726

Table 3: Comparison of the edge detection techniques based on the Shannon Entropy measure for images in Fig. 4d-f.

Detector	Case I			Case II			Case III			Case IV		
	Fig. 4d	Fig. 4e	Fig. 4f	Fig. 4d	Fig. 4e	Fig. 4f	Fig. 4d	Fig. 4e	Fig. 4f	Fig. 4d	Fig. 4e	Fig. 4f
<i>Sobel</i>	0.1651	0.3043	0.2339	0.1533	0.2519	0.2174	0.1553	0.2098	0.2102	0.2461	0.2786	0.2341
<i>Canny</i>	0.2714	0.5007	0.4713	0.8142	0.5910	0.7270	0.8333	0.7535	0.7673	0.4327	0.5552	0.8143
<i>Roberts</i>	0.1795	0.2327	0.2223	0.1005	0.1038	0.1314	0.0623	0.0567	0.0873	0.1923	0.2329	0.1187
<i>LoG</i>	0.2219	0.4278	0.3223	0.4064	0.4606	0.3992	0.5069	0.5281	0.4784	0.3263	0.3543	0.4285
<i>Prewitt</i>	0.1648	0.3012	0.2344	0.1541	0.2512	0.2177	0.1543	0.2139	0.2098	0.2899	0.2822	0.2014
<i>PrM.</i>	0.4758	0.7471	0.5725	0.3819	0.6888	0.5622	0.4511	0.7691	0.5418	0.5161	0.6214	0.6072

Table 4: Comparison of the thresholding scenarios in four mentioned cases based on the quantitative measures for images in Fig. 4. Each value states the average of each measure for all seven images.

	Scenario #1		Scenario #2		Scenario #3		Scenario #4	
	FOM	Entropy	FOM	Entropy	FOM	Entropy	FOM	Entropy
Case I	0.4688	0.2668	0.6819	0.4976	0.6772	0.4734	0.7046	0.5258
Case II	0.4689	0.2718	0.6214	0.4530	0.5663	0.4450	0.6171	0.4948
Case III	0.4407	0.2772	0.6099	0.4817	0.6148	0.4754	0.6258	0.5280
Case IV	0.4892	0.2916	0.5295	0.5264	0.5081	0.4764	0.5541	0.5566

Table 5: Comparison of the Average Precision with some other methods from the literature on BSDS500 dataset.

SCG [42]	0.773	DeepContour [15]	0.800	COB [43]	0.859	Deep Boundary [44]	0.789
PMI [45]	0.799	HFL [46]	0.795	DCD [47]	0.849	CED [48]	0.847
OEF [49]	0.820	HED [16]	0.840	AMH-Net [17]	0.869	BDCN [29]	0.796
MEDM [31]	0.77	NAGK-MF [28]	0.72	BDP-Net [30]	0.847	PrM.	0.894

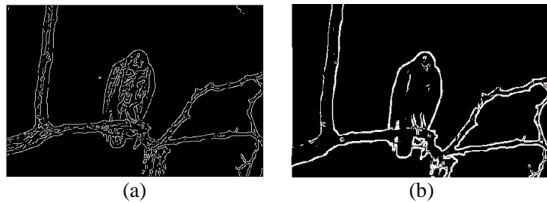


Fig. 8 The outputs of (a) the Canny and (b) the proposed approaches for Fig. 4d. The entropy of the first method is higher than that of the second method. Though, the output of the latter method seems more appropriate.

Four Scenarios were also considered and compared for the thresholding of the force feature histogram. In the first scenario, we applied the Otsu thresholding method to the histogram of the force feature to find the optimal threshold

according to which, we classified the pixels to edges and non-edges. The second scenario found the optimal threshold by starting from the mean of the histogram and following the iterative process (12-15). The third scenario directly considered (16) as the optimal threshold of the histogram. The parameters of the model were found using the GOA. Finally, the fourth scenario (i.e., the proposed method) made use of the model in (16) as $T_{initial}$ whose optimal parameters were found using the GOA. Then, the iterative process (12-15) was performed to find the optimal threshold. It should be noticed that, the optimal parameters for Scenarios #3 and #4 are computed independently, by the GOA. The results of applying these four scenarios to four previously mentioned cases evaluated by the average of FOM and Entropy measures are reported in Table 4. It should be noticed that the results of Scenario #4 were

already presented in Tables 1-3. The results established that, the effectiveness of the proposed method is the highest one among the four scenarios in all the cases. The Scenarios #2 and #3 are generally positioned in the second and the third ranks, while the Otsu thresholding (Scenarios #1) yields the lowest rate.

Average Precision

To compare the performance of the proposed approach with some recent edge-detection methods according to the Average Precision (AP) measure, we used the BSDS500 benchmark with clean images; results are summarized in Table 5. It should be noticed that the APs of the other methods are directly reported from the literature. As shown in the results, the proposed method achieves the AP of 0.894 which outperforms all of these competing methods.

5- Conclusion

Various heuristic and nature-inspired techniques for the image-understanding have been proposed in the last decade. In this paper, a new thresholding approach for the gravitational edge detector is designed to obtain an optimal edge image in clean and noisy conditions. The quantitative and qualitative measures confirmed that the performance of the proposed method was higher than a variety of other classic and recent detectors. On the BSDS500 dataset, the proposed method obtained the Average Precision value of 0.894. Experiments on a number of images proved that, this method was reliable and efficient for the edge detection, also it achieved more acceptable results in comparison with other techniques. Furthermore, the similarity of the output images and the ideal edge maps were far more, compared with other detectors. That is to say, the extracted edges using the proposed method had a close Euclidean distance with the ideal edges; as validated by high FOM. Selecting an optimal threshold value in the developed edge detector resulted in covering the weak edges of the image and obtaining the maximum information (entropy) in the output. The main disadvantage of this approach is its sensitivity to noise which can be solved by the pre-processing operations. Using a 5×5 mask decreased the sensitivity to the Gaussian and salt and pepper noises. This, demonstrated the method robustness in noisy conditions, as entropy slightly changes. Furthermore, the higher average precision of our method validated its dominance over some recently published methods. The results of the quantitative and qualitative evaluations of the detected edges showed that the conducted method had merit in detecting edges in noisy conditions among all tested detectors. For further research, it is suggested to use a multi-modal thresholding approach to enhance the

segmentation performance in multi-class images. Also, the evaluation of the method of this work in color images can be considered. Additionally, the effect of applying non-square neighboring area will be studied in a further work.

References

- [1] R. C. Gonzalez and R. E. Woods, "Digital image processing," ed: Prentice hall Upper Saddle River, 2002.
- [2] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, and U. Stilla, "Classification with an edge: Improving semantic image segmentation with boundary detection," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 135, pp. 158-172, 2018.
- [3] I. Sobel, "Camera models and machine perception [PhD Dissertation]," ed: Stanford, CA: Stanford University, 1970.
- [4] J. M. Prewitt, "Object enhancement and extraction," *Picture processing and Psychopictorics*, vol. 10, no. 1, pp. 15-19, 1970.
- [5] J. Canny, "A computational approach to edge detection," *IEEE Transactions on pattern analysis and machine intelligence*, no. 6, pp. 679-698, 1986.
- [6] L. S. Davis, "A survey of edge detection techniques," *Computer graphics and image processing*, vol. 4, no. 3, pp. 248-270, 1975.
- [7] L. Bin and M. S. Yeganeh, "Comparison for image edge detection algorithms," *IOSR Journal of Computer Engineering*, vol. 2, no. 6, pp. 1-4, 2012.
- [8] L. Romani, M. Rossini, and D. Schenone, "Edge detection methods based on RBF interpolation," *Journal of Computational and Applied Mathematics*, vol. 349, pp. 532-547, 2019.
- [9] S. Eser and A. Derya, "A new edge detection approach via neutrosophy based on maximum norm entropy," *Expert Systems with Applications*, vol. 115, pp. 499-511, 2019.
- [10] Y. He and L. M. Ni, "A novel scheme based on the diffusion to edge detection," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1613-1624, 2018.
- [11] R. K. Bhogal and A. Agrawal, "Image Edge Detection Techniques Using Sobel, T1FLS, and IT2FLS," in *Information and Communication Technology for Intelligent Systems*: Springer, 2019, pp. 303-317.
- [12] A. Banharsakun, "Artificial bee colony algorithm for enhancing image edge detection," *Evolving Systems*, vol. 10, no. 4, pp. 679-687, 2019.
- [13] O. P. Verma and A. S. Parihar, "An optimal fuzzy system for edge detection in color images using bacterial foraging algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 1, pp. 114-127, 2016.
- [14] M. D. Ansari, A. R. Mishra, and F. T. Ansari, "New divergence and entropy measures for intuitionistic fuzzy sets on edge detection," *International Journal of Fuzzy Systems*, vol. 20, no. 2, pp. 474-487, 2018.
- [15] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang, "Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3982-3991.

- [16] S. Xie and Z. Tu, "Holistically-nested edge detection," in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1395-1403.
- [17] D. Xu, W. Ouyang, X. Alameda-Pineda, E. Ricci, X. Wang, and N. Sebe, "Learning deep structured multi-scale features using attention-gated crfs for contour prediction," in Advances in Neural Information Processing Systems, 2017, pp. 3961-3970.
- [18] R. P. Olenick, T. M. Apostol, D. L. Goodstein, and A. Arons, "The Mechanical Universe: Introduction to Mechanics and Heat," ed: AAPT, 1986.
- [19] E. Rashedi and H. Nezamabadi-Pour, "A stochastic gravitational approach to feature based color image segmentation," Engineering Applications of Artificial Intelligence, vol. 26, no. 4, pp. 1322-1332, 2013.
- [20] G. Sun, A. Zhang, Y. Yao, and Z. Wang, "A novel hybrid algorithm of gravitational search algorithm with genetic algorithm for multi-level thresholding," Applied Soft Computing, vol. 46, pp. 703-730, 2016.
- [21] J. J. D. M. S. Junior, A. R. Backes, and P. C. Cortez, "Color texture classification based on gravitational collapse," Pattern Recognition, vol. 46, no. 6, pp. 1628-1637, 2013.
- [22] G. Sun, Q. Liu, 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.
- [23] C. Lopez-Molina, H. Bustince, J. Fernández, P. Couto, and B. De Baets, "A gravitational approach to edge detection based on triangular norms," Pattern Recognition, vol. 43, no. 11, pp. 3730-3741, 2010.
- [24] O. P. Verma and R. Sharma, "An optimal edge detection using universal law of gravity and ant colony algorithm," in 2011 World Congress on Information and Communication Technologies, 2011: IEEE, pp. 507-511.
- [25] O. P. Verma, R. Sharma, M. Kumar, and N. Agrawal, "An optimal edge detection using gravitational search algorithm," Lecture Notes on Software Engineering, vol. 1, no. 2, p. 148, 2013.
- [26] F. Deregeh and H. Nezamabadi-Pour, "A new gravitational image edge detection method using edge explorer agents," Natural Computing, vol. 13, no. 1, pp. 65-78, 2014.
- [27] G. Sun et al., "Gravitation-based edge detection in hyperspectral images," Remote Sensing, vol. 9, no. 6, p. 592, 2017.
- [28] D. Wang, J. Yin, C. Tang, X. Cheng, and B. Ge, "Color edge detection using the normalization anisotropic Gaussian kernel and multichannel fusion," IEEE Access, vol. 8, pp. 228277-228288, 2020.
- [29] J. He, S. Zhang, M. Yang, Y. Shan, and T. Huang, "Bi-directional cascade network for perceptual edge detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3828-3837.
- [30] K. Li, Y. Tian, B. Wang, Z. Qi, and Q. Wang, "Bi-Directional Pyramid Network for Edge Detection," Electronics, vol. 10, no. 3, p. 329, 2021.
- [31] Y. Lu, C. He, Y. F. Yu, G. Xu, H. Zhu, and L. Deng, "Vector co-occurrence morphological edge detection for colour image," IET Image Processing, 2021.
- [32] S. Anand and G. Sangeethapriya, "Gaussian modulated hyperbolic tangent high pass filter for edge detection in noisy images," arXiv preprint arXiv:2005.11432, 2020.
- [33] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: theory and application," Advances in Engineering Software, vol. 105, pp. 30-47, 2017.
- [34] C. Lopez-Molina, B. De Baets, and H. Bustince, "Generating fuzzy edge images from gradient magnitudes," Computer Vision and Image Understanding, vol. 115, no. 11, pp. 1571-1580, 2011.
- [35] N. Otsu, "A threshold selection method from gray-level histograms," IEEE transactions on systems, man, and cybernetics, vol. 9, no. 1, pp. 62-66, 1979.
- [36] M. Mafarja, I. Aljarah, H. Faris, A. I. Hammouri, A.-Z. Ala'M, and S. Mirjalili, "Binary grasshopper optimisation algorithm approaches for feature selection problems," Expert Systems with Applications, vol. 117, pp. 267-286, 2019.
- [37] X. Zhang, Q. Miao, H. Zhang, and L. Wang, "A parameter-adaptive VMD method based on grasshopper optimization algorithm to analyze vibration signals from rotating machinery," Mechanical Systems and Signal Processing, vol. 108, pp. 58-72, 2018.
- [38] U. SIPI, "The usc-sipi image database," ed, 2016.
- [39] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation," IEEE transactions on pattern analysis and machine intelligence, vol. 33, no. 5, pp. 898-916, 2010.
- [40] S. E. Umbaugh, Computer imaging: digital image analysis and processing. CRC press, 2005.
- [41] C. E. Shannon, "A mathematical theory of communication," Bell system technical journal, vol. 27, no. 3, pp. 379-423, 1948.
- [42] R. Xiaofeng and L. Bo, "Discriminatively trained sparse code gradients for contour detection," in Advances in neural information processing systems, 2012, pp. 584-592.
- [43] K.-K. Maninis, J. Pont-Tuset, P. Arbeláez, and L. Van Gool, "Convolutional oriented boundaries," in European Conference on Computer Vision, 2016: Springer, pp. 580-596.
- [44] I. Kokkinos, "Pushing the boundaries of boundary detection using deep learning," arXiv preprint arXiv:1511.07386, 2015.
- [45] P. Isola, D. Zoran, D. Krishnan, and E. H. Adelson, "Crisp boundary detection using pointwise mutual information," in European Conference on Computer Vision, 2014: Springer, pp. 799-814.
- [46] G. Bertasius, J. Shi, and L. Torresani, "High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision," in Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 504-512.
- [47] Y. Liao, S. Fu, X. Lu, C. Zhang, and Z. Tang, "Deep-learning-based object-level contour detection with CCG and CRF optimization," in 2017 IEEE International Conference on Multimedia and Expo (ICME), 2017: IEEE, pp. 859-864.
- [48] W. Yupei, Z. Xin, and K. Huang, "Deep crisp boundaries," 2017: CVPR.
- [49] S. Hallman and C. C. Fowlkes, "Oriented edge forests for boundary detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1732-1740.

Kimia Rezaei received her BSc and MSc from Islamic Azad University, Fasa and Shiraz Branches, Fars, Iran. She is currently a PhD student at Department of Electrical and Electronic Engineering, University College Cork, Ireland. Her research interests are pattern recognition and Image processing for practical applications.

Hamed Agahi received B.Sc. M.Sc. and Ph.D. degrees in Electrical Engineering from University of Shiraz, Amirkabir University of Technology and University of Tehran, Iran, in 2005, 2008 and 2013, respectively. From 2009, he was with the Islamic Azad University, Shiraz Branch, Shiraz, Iran. His research interests include pattern recognition, image and signal processing, and fault detection and diagnosis applications.