Republic of Iraq

Ministry of Higher Education & Scientific Research

University of Diyala

College of Education For pure science

Edge detection &Enhancement of Medical Images

A project Submitted to the Department of Computer Science at the College of Education for pure science in partial Fulfillment of the Requirements for the Bachelor of Science in Computer Science.

Prepared By

Abeer jassim abdolah

Supervised By

***Dr.*Adel A. Abd ALWahab**

2018-2019

CHAPTER ONE

GENERAL INTRODUCTION

1.1 Introduction

The finding of edges of the image is done in several traditional ways. The unconventional method is the use of an algorithm derived from nature. A swarm is a large number of homogenous , simple agents interacting locally among themselves , and their environment, with no central control to allow a global interesting behavior to emerge. Swarm-based algorithms have recently emerged as a family of nature-inspired, population-based algorithms that are capable of producing low cost, fast, and robust solutions to several complex problems. Swarm Intelligence (SI) can therefore be defined as a relatively new branch of Artificial Intelligence that is used to model the collective behavior of social swarms in nature, such as ant colonies honey bees, and bird flocks. Although these agents (insects or swarm individuals) are relatively unsophisticated with limited capabilities on their own, they are interacting together with certain behavioural patterns to cooperatively achieve tasks necessary for their survival. The social interactions among swarm individuals can be either direct or indirect . Examples of direct interaction are through visual or audio contact, such as the waggle dance of honey bees. Indirect interaction occurs when one individual changes the environment and the other individuals respond to the new environment, such as the pheromone trails of ants that they deposit on their way to search for food sources. This indirect type of interaction is referred to as stigmergy, which essentially means communication through the environment [1].

In more than three decades, swarm intelligence has been successfully used to solve optimization problems in the engineering, the financial, and the management fields. For example, particle swarm optimization (PSO) techniques have successfully been used to construct the portfolios of stock and to construct parameters in neural network systems ; ant colony optimization (ACO) techniques have successfully been used to solve the traveling salesman problem

(TSP) and the routing problem of networks ; cat swarm optimization (CSO) techniques have successfully been used to discover proper positions for information hiding.

In 2010, Yang proposed a new optimization algorithm, namely, Bat Algorithm (BA), based on swarm intelligence and the inspiration form observing the bats . Although the original BA presents superior results in the experiments than PSO, we notice that the performance and the accuracy of the original BA still have the capacity to present better[2].

1.2 The Relationship Between Finding Edges and Medical Images

The edge detection includes a variety mathematical methods that help define points in the medical image when the screen light changes sharply or intermittent lighting . the point of the image in which the light is sharply changed in the set of discontinuous curves is called" Edges'. The finding of the edges of the medical images makes the diagnosis more efficient and more capable of reading data and analysis, which means accurate access to diagnose the disease and then treatment. This provided many operations that require surgery for the patient and may cause more complications in the future.

1.3 Applications of Bat Algorithm

In past several years, PSO has been successfully applied in many research and application areas. It is demonstrated that PSO gets better results in a faster, cheaper way compared with other methods.

Another reason that PSO is attractive is that there are few parameters to adjust. One version, with slight variations, works well in a wide variety of applications. Particle swarm optimization has been used for approaches that can be used across a wide range of applications, as well as for specific applications focused

on a specific requirement [3]. The standard bat algorithm and its many variants mean that the applications are also very diverse. In fact, since the original bat algorithm has been developed (Yang,2010), bat algorithms have been applied in almost every area of optimization, classifications, image processing, feature selection, scheduling, data mining and others. In the rest of the paper, we will briefly highlight some of the applications[4] .

1.3.1 Combinatorial Optimisation and scheduling

From computational complexity point of view, continuous optimization problem can be considered as easy, thought it may be still very challenging to solve. However , combinatorial problems can be really hard , often non-deterministic polynomial time hard(NP-hard).Multi-stage , multi-machine ,multi-product scheduling and a class of NP hard problems can be solved using bat algorthim.

1.3.2 Continuous Optimisation

Among the first set of applications of bat algorithm, continuous optimisation in the context of engineering design optimisation has been extensively studied, which demonstrated that BA can deal with highly nonlinear problems efficiently and can find the optimal solutions accurately . Case studies include pressure vessel design, car side design, spring and beam design, truss systems, tower and tall building design and others.In addition, numerical optimization , multiobjective ,and brushless DC wheel motors problems can be handled using bat algorithm with superior results .

1.3.3 Classifications, Clustering and Data Mining

Bat algorithm can be applied for studying clustering problem for office workplaces , e-learning and hybrid flow shop scheduling problems.

1.3.4 Image Processing

Bat algorithm can be used for image processing such as image matching ,edge detection,contrast enhancement ,segmentation and and image thresholding which is an important approach for image segmentation .

1.4 Letrature survey

Bat algorithm is a bio-inspired algorithm developed by Xin-She Yang in 2010. He proposed a new optimization algorithm ,namely, Bat algorithm, based on swarn intelligence and the inspiration form observing the bats [5].

P.Tasi , J.Pan , B. Liao , M. Tasi and V. Istanda presented a method for solving numerical optimization problems Based on the framework of the original Bat algorithm [2].

A. Alihodzic and M.Tuba are proposed a global multilevel Thresholding algorithm for image segmentation based on the Bat inspired algorithm [6].

S. Dhar ,S. Alam , M. Santra P. Saha & S. Thakur proposed a noval method for edge detection in a gray image Based on Human Psych visual (HVS) and a relatively new evolutionary algorithm , the bat algorithm [7] .

R. Dwivedi , H. Sethi , M. Rohilla , V. Gray and Y. Nagpal focused on solving the edge detection problem using the concepts of BA from the pool of swarm intellegence algorithms [8].

P. Kora and S. Kalva presented a method of extracting key features from each cardiac beat using improved bat algorithm. Using this algorithm best features are extracted, then these best (reduced) features are applied to the input of the neural network classifier[9].

CHAPTER TWO

EDGE DETECTION TECHNIQUES & ENHANCEMENT

2.1 Introduction

Image is formed in the eye and in the camera by the amount of illumination reflected by an object. In computer vision, image processing is any form of signal processing for which the input is an image, such as photographs or frames of videos. The output of image processing can be either an image or a set of characteristics or parameters related to image. The image processing techniques like image restoration, image enhancement, image segmentation edge detection …ect [10].

The edge detection is widely applied in areas like image recognition, image classification, image enhancement, and in pattern recognition in general. Applying gradient operators on images can result in image edges when the edge gradient values exceed some defined thresholds.

2.2 Image Enhancement

Image enhancement is the process of digitally manipulating a stored image using software .The tools used for image enhancement include many different kinds of software .The such as filters , image editors and other tools for changing various properties of an entire image or parts an image.

Some of the most basic types of image enhancement tools simply change the contrast or brightness of an image or manipulate the grayscale or the red-green – blue color patterns of an image. Some types of basic filters also allow changing a color image to black and white , or to a sepia-tone image , or adding visual effects.

More sophisticated types of image enhancement tools can apply changes more specifically to certain parts of an image .Professional packages like those offered by Adobe allow designers to do a more specialized or professional kind of image enhancement or to pursue results for graphic design projects where the actual image is changed into a stylized or otherwise embellished version of itself. More advanced types of image enhancement tools also include features like Wiener filters for actual de-blurring of images and other complex resources for restoring or clarifying images that may be in poor condition , due to sub-optimal image capture conditions , aging or other causes

2.3 Edge Models

Large changes in image brightness over a short spatial distance indicates the presence of an edge. Edge is a set of connected pixels that lay on the boundary between two regions whose gray level has outstanding change. The edge can by located between objects and background or two objects. Edge models are classified according to their intensity profiles. A step edge involves a transition between two intensity levels occurring ideally over the distant of 1 pixel.

Figure 2-1(a) shows a section of a vertical step and a horizontal intensity profile though the edge. Step edges occur, for example, in images generated by a computer for use in areas such as solid modelling and animation. These clean, ideal edges can occur over the distance of 1 pixel, provided that no additional processing (such as smoothing) is used to make them look real. Digital step edges are used frequently as edge models in algorithm development.

In practice, digital images have edges that are blurred and noisy, determined by limitations in the focusing mechanism and noise level determined by the electronic components of imaging system. In such situations, edges are more closely modeled as having an intensity ramp profile such as edge in Figure 2-1(b). The slope of the ramp is inversely proportional to the degree of blurring in the edge. It can be seen that in this model, there is no thin (1 pixel thick) path. Instead, an edge point which is now any point contained in the ramp, and an edge segment would then be a set of such points that are connected. A third model of an edge is called roof edge, having the characteristics illustrated in Figure 2-1(c). Roof edges are models of lines through a region, with the base(width) of a roof edge being determined by the thickness and sharpness of the line .The figures below show the various edge models [10,11].

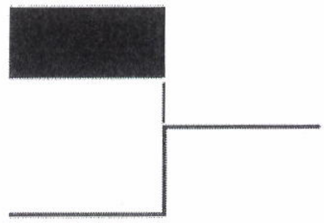


Figure 2-1(a): Step Edge with intensity profile

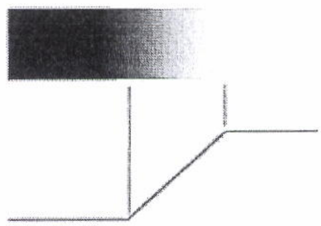


Figure 2-1(b): Ramp Edge with intensity profile

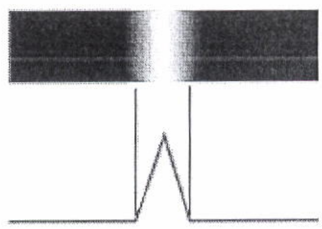


Figure 2-1(c): Roof Edge with intensity profile

2.4 Edge Detection

Edge detection is considered as a paramount step in many image transforming functions. The information obtained from it given as an input to feature extraction and object segmentation for further processing [12]. For edge characterization, the intensity function is calculated and analyzed to identify the sharp discontinuities in an image. For better results, the considered edges should be of good quality as much as possible. When an image is reduced to its edge pixels, it saves memory, as well as these edge details, can be used in feature detection and object recognition. Edge pixels, which play an important role in image analysis, are determined by practicing boundary specification of Sobel ,

Prewitt , Kirsch and Canny .. etc. These rules certainly benefit the purpose but increased the significant overhead of eliciting and managing massive data as compared to the simpler fuzzy methods [13].

Swarm-based algorithms are recent, nature-influenced, culture-based algorithms. Swarm Intelligence (SI) is used to illustrate the course of communal swarms in nature, such as ant colonies, bats, and bird flocks. Swarm Intelligence principles have applications in various problem domains, for example, the discovery of most favorable path, function optimization problems, anatomical commutation, image and data dissection, and scheduling [5]. Computational modeling of swarms has been gaining popularity in a broad-amplitude of distinct realms, for example- in bioinformatics domain, in medical informatics, in dynamical systems, and operations research.

In the recent times, various swarm intelligence algorithms are designed and are used for distinct purposes the popular ones are- Ant Colony Optimization (ACO), Gravitational Search Algorithm (GSA) and Bat Algorithm (BA).

As stated earlier, Bat algorithm is energized by the trait of echolocation attitude of bats. This action helps them to operate and collect their necessities even in the absence of light The echolocation of the bats is observed as:

Every implicit bat shoots with an acceleration pi at location (solution) zi with a varying recurrence or wavelength and vibration Vi. As it hunts and discovers its food, it changes frequency, loudness and pulse emission rate ri. The bats take a local random walk intensifying the search. Until the convinced break stone is faced, selection of the best continues. The frequency-tuning approach is used to curb the disciplined progressive conduct of throng of bats.

Bat Algorithm uses a combination of frequency tuning technique and automatic zooming that empowers it to elaborate distinct solution in the community and provides symmetry in analysis and exploitation in the interim of the hunt operation by imitating or simulating the disparity of pulse emission

rates and loudness of bats when hunting for prey. As an outcome, it confirms to be very competent and powerful with an exemplary abrupt outset [4].

In this research, Bat Algorithm is used to resolve the edge detection problem in an optimal way. The local asymmetry in image acuteness caused by the movement of agents in search of food and prey is used to reveal the edge pixels [8].

2.5 The Standard Bat Algorithm

The standard bat algorithm, developed by Xin-She Yang in 2010, was based on the echolocation or bio-sonar characteristics of microbats . Before we outline the details of the algorithm, let us briefly introduce the echolocation.

2.5.1 Echolocation of Microbes

Most bats uses echolocation to a certain degree; among all the species, microbes use echolocation extensively, while megabes do not.

Microbes typically use a type of sonar, called, echolocation, to detect prey, avoid obstacles, and locate their roosting crevices in the dark. They can emit a very loud sound pulse and listen for the echo that bounces back from the surrounding objects (Richardson, 2008).Their pulses vary in properties and can be correlated with their hunting strategies, depending on the species. Most bats use short, frequency-modulated signals to sweep through about an octave, and each pulse lasts a few thousandths of a second (up to about 8 to 10 ms) in the frequency range of 25 kHz to 150 kHz. Typically, microbats can emit about 10 to 20 such sound bursts every second, and the rate of pulse emission can be sped up to about 200 pulses per second when homing on their prey. Since the speed of sound in air is about $\upsilon = 340$ m/s, the wavelength λ of the ultrasonic sound bursts with a constant frequency f is given by $\lambda = \upsilon / f$, which is in the range

of 2mm to 14mm for the typical frequency range from 25 kHz to 150 kHz. Interestingly, these wavelengths are in the same order of their prey sizes.

Though in reality microbats can also use time delay between their ears and loudness variations to sense three-dimensional surroundings, we are mainly interested in some features of echolocation so that we can some link them with the objective function of an optimization problem, which makes it possible to formulate a smart, bat algorithm[4].

2.5.2 Bat Algorithm

Based on the above description and characteristics of bat echolocation, Xin-She Yang (2010) developed the bat algorithm with the following three idealized rules:

1. All bats use echolocation to sense distance, and they also `know' the difference between food/prey and background barriers in some magical way.
2. Bats fly randomly with velocity υ_i at position x_i with a frequency f (or wavelength λ) and loudness A_o to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission $r \in [0 ; 1]$, depending on the proximity of their target.
3. Although the loudness can vary in many ways , we assume that the loudness varies from a large (positive) A_o to a minimum constant value A_{min}.

For simplicity, we do not use ray tracing in this algorithm, though it can form an interesting feature for further extension. In general, ray tracing can be computationally extensive, but it can be a very useful feature for computational geometry and other applications. Furthermore, a given frequency is intrinsically linked to a wavelength. For example, a frequency range of [20kHz , 500kHz] corresponds to a range of wavelengths from 0.7mm to 17mm in the air. Therefore, we can describe the changes either in terms of frequency f or wavelength λ to suit different applications, depending on the ease of implementation and other factors [4].

2.5.3 Bat Motion

Each bat is associated with a velocity υ_i^t and a location x_i^t, at iteration t, in a d-dimensional search or solution space. Among all the bats, there exists a current best solution x*. Therefore, the above three rules can be translated into the updating equations for x_i^t and velocities υ_i^t :

$$f_i = f_{min} + (f_{max} - f_{min})\,\beta \quad \text{--------} \quad (1)$$

$$\upsilon_i^t = \upsilon_i^{t-1} + (X_i^{t-1} - X^*)\, f_i \quad \text{---------} \quad (2)$$

$$X_i^t = X_i^{t-1} + \upsilon_i^t \quad \text{-------------------} \quad (3)$$

Where $\beta \in [0,1]$ is a random vector drawn from a uniform distribution.

As mentioned earlier, we can either use wavelengths or frequencies for implementation, we will use fmin = 0 and fmax = ٠ (1), depending on the domain size of the problem of interest. Initially, each bat is randomly assigned a frequency which is drawn uniformly from [fmin , fmax]. For this reason, bat algorithm can be considered as a frequency-tuning algorithm to provide a balanced combination of exploration and exploitation. The loudness and pulse emission rates essentially provide a mechanism for automatic control and auto-zooming into the region with promising solutions[4,9,14].

2.5.4 Local Search of bat

A random walk with direct exploitation is used for the local search that modifies the current best solution according the equation:

$$X_{new} = X_{old} + \partial A^t \quad \text{---------------} \quad (4)$$

where $\partial \in [-1,1]$ is a random number, while A^t is the average loudness of all the best at this time step[4,9,14].

2.5.5 Variations of Loudness and Pulse Rates

In order to provide an effective mechanism to control the exploration and exploitation and switch to exploitation stage when necessary, we have to vary the loudness A_i and the rate r_i of pulse emission during the iterations.

Since the loudness usually decreases once a bat has found its prey, while the rate of pulse emission increases, the loudness can be chosen as any value of convenience, between A_{min} and A_{max}, assuming $A_{min} = 0$ means that a bat has just found the prey and temporarily stopped emitting any sound.

With these assumptions, we have

$$A_i^{t+1} = \alpha A_i^t \quad \text{----------------------} \quad (5)$$

and

$$r_i^t = r_i^o [1 - \exp(-\Upsilon t)] \quad \text{---------} \quad (6)$$

where α and Υ are constants.

A_i^t and A_i^{t+1} are the loudness at the proceed and present iteration, respectively.

r_i^o is the initial pulse emission rate.

r_i^t is the pulse emission rate at the iteration t.

In essence, here α is similar to the cooling factor of a cooling schedule in simulated annealing. For any $0 < \alpha < 1$ and $\Upsilon > 0$[4,9,14,15]

.5 General bat algorithm

Objective function $f(X)$, $X = (X_1, \ldots, X_d)^t$

Initialize the bat population X_i ($i = 1, 2, \ldots, n$) and V_i

Define pulse frequency f_i at X_i

Initialize pulse rates r_i and the loudness A_i

While (t < Max number of iterations)

Generate new solutions by adjusting frequency,

and updating velocities and locations / solutions [equations (2) to (4)]

if (rand > r_i)

Select a solution among the best solutions

Generate a local solution around the selected best solution

end if

Generate a new solution by flying randomly

if (rand < A_i & $f(X_i) < f(X^*)$)

Accept the new solutions

Increase r_i and reduce A_i

end if

Rank the bats and find the current best X^*

end while

Postprocess results and visualization

CHAPTER THREE

EXECUTION OF OUR WORK

3.1 Proposed approach

1) Given an image of (x*y) pixels.

2) Taken n of pixels as a population that is distributed over all image.Each one of these pixels represents a bat having two values X and Y.

3) The values of X and Y are the X-axis and Y-axis positions of the bat (pixel).

4) For each bat, suppose the following values :-

The pulse rate ri ,velocity Vi , loudness Ai, minimum frequency (f_{min}) and maximum frequency (f_{max}).

5) calculate values of ri , Ai , fi for each bat as follows:-

A) ri represents the largest difference in intensity between the bat (pixel) and it's 8 neighbors.

B) Ai represents the smallest difference in intensity between the bat (pixel) and it's 8 neighbors.

C) fi from equation:-

$$F_i = f_{min} + (f_{max} - f_{min})\, \beta$$

Where β is random value.

6) Determine is the value of the best pixel for each bat, which initially represent the value of the bat itself and this will be updated when the bat moves to new pixels in the search for the goal. Where it will always represents the best pixel passed by the bat . this priority is measured base on the pulse rate (from step 5).

7) Start a main loop with total number of repeats tmax , each repetition includes n of bats (pixels).

8) Start a loop represents number of bats $i=1 \rightarrow i=n$, n is the number of bats for the population.

9) Calculate a new value for the bat (pixel) using equations 1, 2 and 3 .

10) Calculate the new pulse rate for the resulting point of step 9 .

11) Compare the best pulse rate with the pulse rate of the new pixel to decide whether reject or accept the new pixel.

If it is rejected, go to local search in equation 4 to find anew pixel.

12) Check if the new pixel obtained from step 11 to be an edge or not by comparing its gray level with the fitness that depends on a given threshold value.

13) Repeat steps (9-11) for each bat and for n bats.

14) Repeat steps (8-13) to the value of t max.

15) Arrange the display of results that represent the edges of the image.

16) start a loop to Enhance the results by trying to increase losted edges by testing the neighborhoods of each resulted pixel.

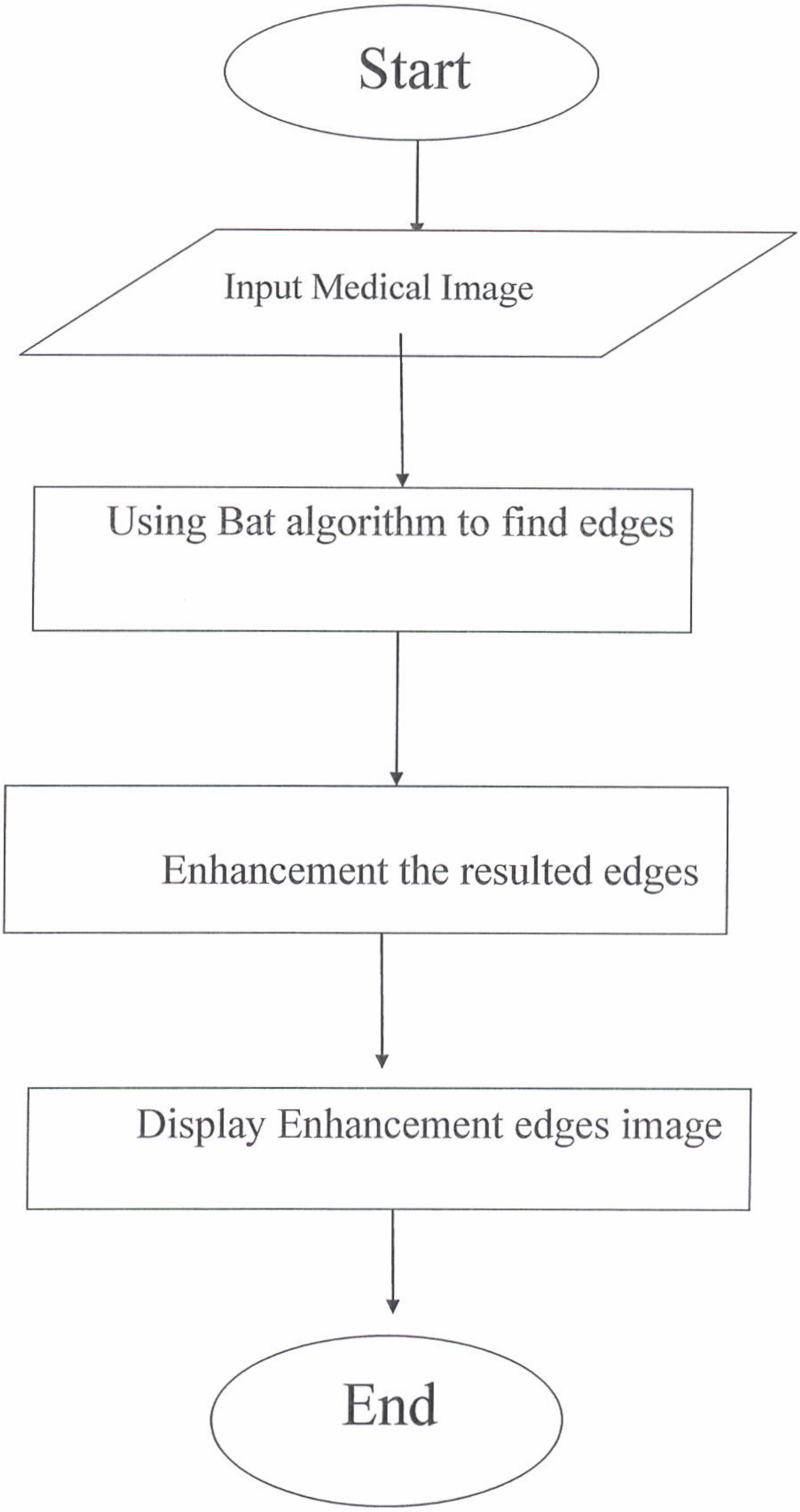
17)If we find a new edge it must be added to the array of the edges array .

18)Continue of finding new edges for each old edge .

19)Stop searching for edges when no new edges are found.

20)Display the image edges often enhancement.

3.2 Flowchar of proposed approacl



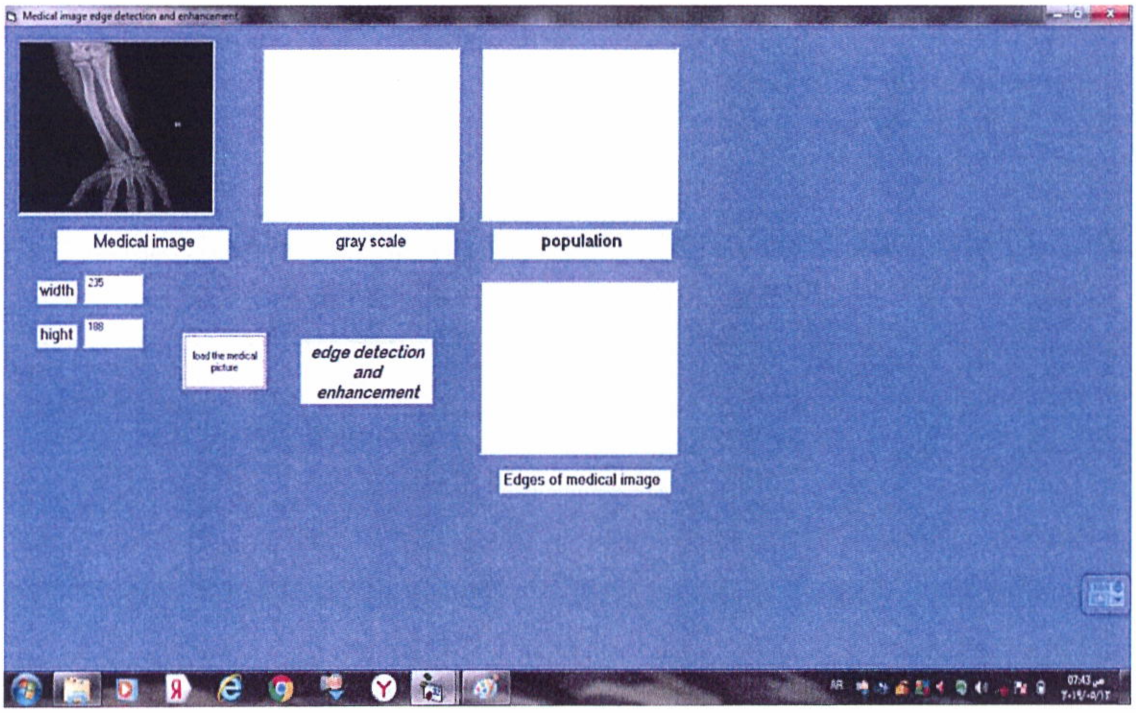
3.3 Experimental results :

Bat algorithm was implemented in visual basic programming language . Different images are taken as the test images . Parameters of the Bat algorithm are population size (n) which represents the number of bats (pixel of image) , the maximum number of iterations tmax , loudness (A) , pulse emission rate (r), minimum frequency fmin , maximum frequency f max and velocity (v). We used a random size of population and number of iteration (t max = 100) , fmin=0 , fmax=2, velocity=1.

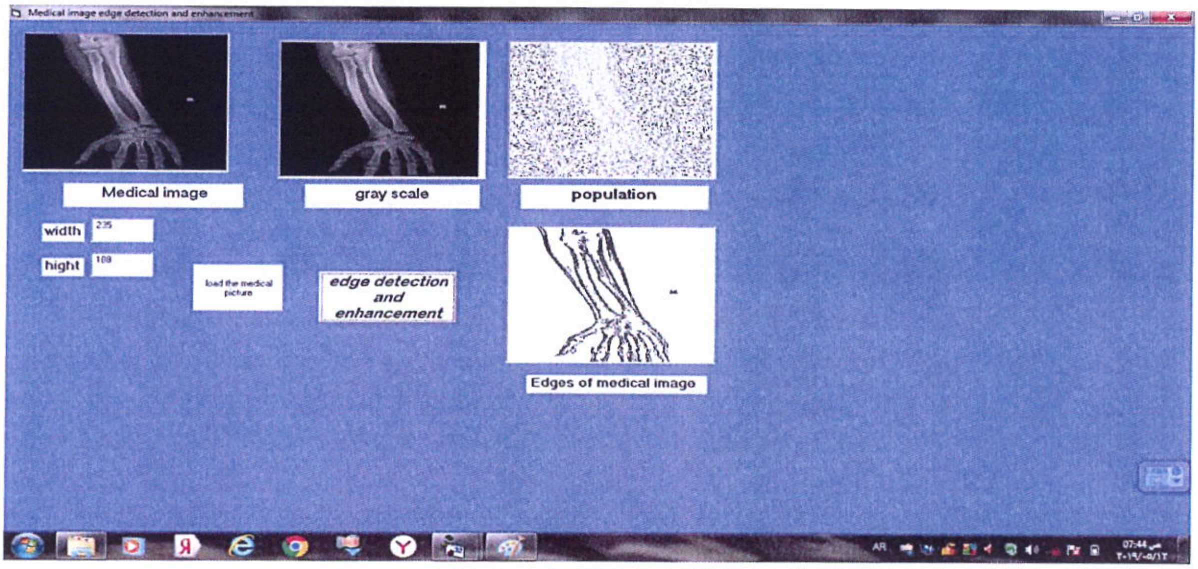
Implementation of Bat algorithm code obtained the edges of test images shown in following figures .

Clarify the program interface

The software interface includes a picture box in which the medical image must be loaded by doing click on the button "load the medical image "then it will be ready to find edges for the loaded image



When entering the medical image in the program will appear image dimensions(length and width):-



When you press the Edge detection and Enhancement button will appear in picture box gray scale because this will help to find edges easily depending on ours method .

In the other picture box show us a picture in a manner population .
The third shows us the medical picture box after showing the edges and enhancement.