Chapter 2

Optimization of Methods for Image-Texture Segmentation Using Ant Colony Optimization

Manasa Nadipally
JNTU Hyderabad, Telangana, India

Chapter Outline

2.1 INTRODUCTION

Image segmentation is regarded as an integral component in digital image processing which is used for dividing the image into different segments and discrete regions. The outcome of image segmentation is a group of segments that jointly enclose the whole image or a collection of contours taken out

Intelligent Data Analysis for Biomedical Applications. DOI: https://doi.org/10.1016/B978-0-12-815553-0.00002-1

from the image. We can divide image segmentation into different methods. For instance, methods based on compression techniques propose that the best method of segmentation is the one which minimizes data's coding length and the general probable segmentations. Methods based on histograms are known to be extremely well-organized to evaluate additional segmentation schemes as they need only single exceed in the progression of the pixels [1]. In this scheme, all of the pixels of an image are taken into consideration to figure the histogram, and the valleys and peaks in the histogram are utilized for establishing the clusters in an image. Inside-image processing, edge detection is a robust field on its own. Region edges and boundaries are connected directly since there is often a quick modification in strength at the area of boundaries [2]. Another significant part of image processing is thresholding, which is used for conversion of a grayscale image into a binary image. For this purpose, the threshold value is chosen after selection of multiple-levels. We assume that an image is divided into the following two parts: foreground and background. The Hachemi Guerrout method exhibits its resistance and robustness to noise by employing a Hidden Markov Random Field-Particle Swarm Optimization (HMRF-PSO) technique over threshold-based schemes [3]. Liang and Leung have described a genetic algorithm with adaptive, exclusive, population tactics for function optimization in multimode [4]. Two other scientists, Wang and Huang [5], have described a thresholding method using a selection of an adaptive window for irregular lighting images. The Wang and Jiang methods explain color image segmentation whose basis is region-merging and homogram-thresholding [5]. A clustering technique based on a genetic algorithm has been proposed by the Maulik method [6]. Further, the Chang method has introduced a rapid, multilevel, thresholding technique based on high and low pass filters. The choice of a poor population can result in poor segmentation in multilevel thresholding [7].

The image segmentation of brain MRI is a significant and difficult problem dealing with grain mapping [8]. Correct categorization of magnetic resonance imagery by types of tissues of gray matter (GM), cerebrospinal fluid (CSF), and white matter (WM) at voxel level offers an approach to evaluate brain architecture. Among these, the quantization of WM and GM volumes has large significance for various neurodegenerative disorders; for example in movement disorders like Parkinson's disease and its associated syndromes, Alzheimer's disease, in inflammatory diseases, or WM metabolic in posttraumatic syndrome or congenital brain malformations [9].

However, the automatic image segmentation of brain magnetic resonance image (MRI) is still a persistent problem. Reliable and automatic categorization of tissues is further intricate due to the common characteristics among resonance intensities of various classes of tissues and by the presence of a spatially smooth altering intensity inhomogeneity. Hence, intensity-based

algorithms which are fully automated demonstrate high sensitivity to a variety of noise artifacts, such as intertissue intensity and intratissue noise contrast reduction. Over the past few years, numerous algorithms have been proposed for image segmentation. In image segmentation, the popular method is the thresholding method owing to its efficiency and simplicity. If the target can be distinguished from the background, there will be a bimodal image of the histogram, after which it can easily reach the threshold simply by selecting the bottom of the valley as a threshold point. Nevertheless, in the majority of real images, no visibly noticeable marks between the background and the target are present. Image segmentation by using genetic algorithm methods has suggested a favorable threshold method to be extensively applied [10]. Using this method, the drawback can be overcome. However, calculating the optimum threshold requires a lengthy calculation time.

For resolving numerous optimization issues, the ACO algorithm has been successfully applied, even though it has a limited number of applications in the field of image processing. Recently, researchers have started to implement ACO algorithm to image processing problems, such as texture classification and edge detections [11–13].

In this chapter, we have adopted a novel approach by implementing the idea of ant colonies to the segmentation of iris and brain MRI images. Apart from Tabu search (TS), genetic algorithm (GA), artificial bee (AB), and simulated annealing (SA), ACO [14,15] is another special metaheuristic search algorithm that is useful in complex combinatorial optimization issues, for instance, graph-coloring issues, traveling salesman-based problems, vehicle routing issues, or quadratic assignment based problems, etc. The parallel and discretionary nature of ACO is appropriate for digital images. The purpose behind this is that ACO can search smartly and also possesses fine characteristics like positive feedback, distributed computation, and robustness [16].

In this research, image segmentation is viewed as delineating the area of pixels having a similar background texture. For each of the pixels present in an image, its gradient and brightness, collectively with the brightness and gradient of bordering pixels, are considered as local texture features [17]. In the case of ACO, artificial ants' movement is inclined by such local texture features, and the global pheromone (e.g., food trail secretions or pheromones) distribution on the image of a sizeable quantity of artificial ants tells the texture representation and region segmentation results. Following segmentation, texture representation outcomes are compared for few specific regions of iris and MRI images of patients suffering from iris infection or brain injury. The results obtained from experiments have proved the usefulness of ACO in the segmentation of images; moreover, they have also proved the discriminability of the texture representation based on ACO [18]. Thus the motivation behind this paper is to apply the ant colony approach to the image thresholding problems for medical images (e.g., iris and brain MRI images).

2.2 IMPLEMENTATION OF ANT COLONY OPTIMIZATION ALGORITHM

ACO-based algorithm was invented by an Italian researcher, M. Dorigo [19,20]. The algorithm was inspired by the examination of a real colony of ants and used for finding the best possible path to the source of food in the food-hunting process. In real life, ants are known as social insects living in colonies. Observations reveal that their behavior is targeted more on the survival of their colony as a whole instead of survival of an individual constituent of the colony. The key defining factors for ACO algorithms are discussed next.

2.2.1 Isula Framework

The Isula framework is a computer framework that permits a simple implementation of ACO algorithms using Java Programming Language. This framework includes the common elements that are present in the metaheuristic to permit algorithm designers to reuse common behaviors. Through Isula, optimization problems can be solved with ACO in a few code lines. For solving a problem via an ACO algorithm, we require a colony of agents (known as Ants), a graph demonstrating the problem statement, and a pheromone data structure for allowing communication between these agents.

The Isula framework provides the basic flow for execution of an algorithm in the ACO metaheuristic. We can use the implementations previously accessible for AntColony and AcoProblemSolver; however, we can also extend it based on our requirements. We need to create our Ant instance as per the requirement of our project; still, a lot of functionality is already available for base implementation. For reference, we can look at the projects in the section "Isula in action." Every algorithm of ACO has a set of custom-made behaviors which are executed throughout the solution processes; such behaviors can have a global effect (DaemonAction instances, such as rules of pheromone update) or only impact an ant and its solution (such as rules of component selection: regarded as subclasses of Ant Policy). Isula contains few such behaviors for some representative algorithms; however, we may need to define our policies or expand the ones that are available already (Fig. 2.1).

A significant and fascinating fact concerning ants is their foraging behavior, particularly, how they can discover the shortest path between food resources and their nest [22].

This section presents image segmentation process with ants having different feature use for hunting food resources. Additionally, the source of food is merely the best threshold of image segmentation. Fig. 2.2 illustrates a block diagram of the proposed approach for image segmentation.

We can describe the "food" in our proposed algorithm as the reference object which is remembered by ants during image segmentation

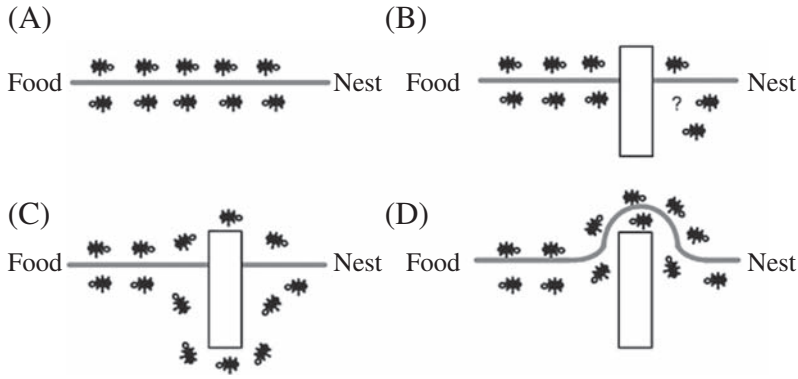

FIGURE 2.1 Ants' food searching approach: (A) ants traveling in a pheromone between food and their nest; (B) interruption of the trail due to an obstacle; (C) discovering two routes for trail movement; and (D) establishing the most favorable route [21].

phenomenon. For ease, we manually chose an r-radius locality $n_r(o)$ of a particular pixel o in the image. Subsequently, the food in the memory of ith ant for time $t = 0$ can be initialized as below:

$$f_{i,t=0} = n_r(o), \tag{2.1}$$

$$n_r(o) = \{e \in I| \quad ||e - o|| < r\}, \tag{2.2}$$

Here, r is an individual ant's state of being exhibited by its position; whereas, e denotes pixels that ants bypass through. I denote the pixels present in the image undergoing segmentation. When ants discover a new source of food, the food in their memory will be revitalized as per the rules of the transition.

After the food definition, the job of the ants is to find pixels having similar properties. For finding similar pixels, ants possess the capability to evaluate pixels to the particular reference food for which they have been searching.

The ants will move from one pixel to another at each step of the iteration. We suppose that the probability of transition of an ant also gets affected by the presence of other ants around it. The influence caused by other ants is limited to a given window, w. Subsequently, on the lattice, the normalized probability of transition to move from pixel to pixel at any time, t, is defined below:

$$p_i = \frac{w(\sigma_i)((v^* w\sigma_i) + E_i)}{\sum w(\sigma_i)((v^* w\sigma_i) + E_i)} \tag{2.3}$$

$$w(\sigma) = \left(1 + \frac{\sigma}{1 + \delta\sigma}\right)^{\beta}, \quad E = \frac{n_r - \text{food}}{nr}, \sigma$$

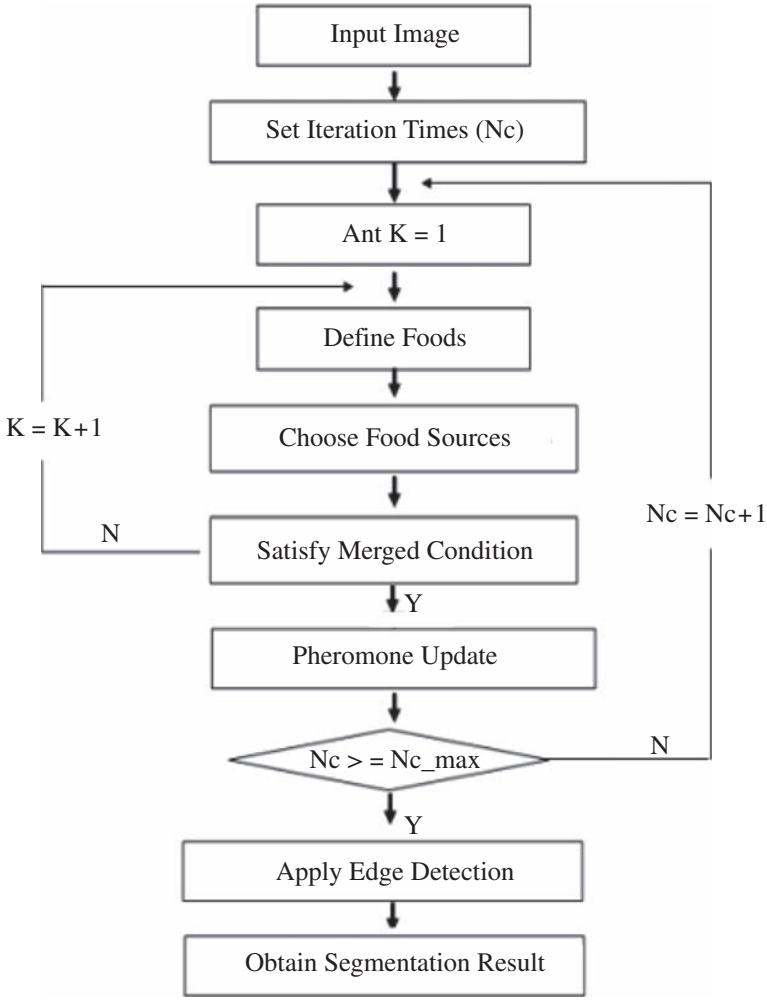

FIGURE 2.2 The flowchart of the proposed image segmentation algorithm.

Here, a pixel r has a pheromone density $\sigma(r)$, whereas, n_r − food is the amount of defined food, v denotes a pheromone rate of decay at each iteration, and nr is the quantity of the whole food. According to this definition, the movement of ants is more like a mass action, which can improve the ability of ants to find a food source.

After the value of p surpasses the threshold, an ant will then consider the target as a source of food. Throughout the process of searching for a source of food, every ant has the value of the threshold. Afterwards, we can define the pheromone deposition τ at pixel as follows:

$$\tau = \begin{cases} \eta & p_i < \lambda \\ \eta + p^* u & p_i \geq \lambda \end{cases} \tag{2.4}$$

In this equation, η is a constant quantity of pheromone; whereas, p is a constant weighting coefficient; and u is the rate at the ants looking for a source of food. Hence, the amount of pheromone changes on every path with the movement of the ant and is adjusted on every path through one circulation. According to Eq. (2.4), a new method has been adopted on updating the pheromone. Lastly, the general termination condition is to set the iterations number.

As individuals, ants are unsophisticated living beings and are unable to correspond or hunt effectively for food sources. However, they are intelligent when functioning in the form of a group, and successfully hunt and gather food for their colony. ACO techniques have been inspired by this collective intelligent behavior of ants. During ants' foraging process, they communicate via a chemical substance known as pheromone. While traveling, an ant deposits a constant quantity of pheromone that can be followed by several other ants. When searching for food, ants are usually inclined to follow trails with a higher concentration of pheromones [23]. Two main working operators in ACO algorithms are discussed in the next section.

2.2.2 Ant Route Construction

During the first stage, the traveling ants usually construct a route randomly on their path to food. Nonetheless, the ants succeeding chase the probability-based path construction scheme.

2.2.3 Ant Pheromone Update

Two important stages are involved in this process. Initially, a special chemical known as a pheromone is left on the path crossed by the individual ants. Later, this deposited pheromone evaporates. The amount of pheromone that is updated on an individual route is a cumulative outcome of these two stages.

2.3 IMAGE SEGMENTATION TECHNIQUES

For acquiring information from pictures through digital image analysis, the standard characteristics of images must be identified. This can be achieved by employing several characteristics of the contents of the images, such as edges, color strength, and regions. Techniques based on image content can be further divided into edge-based, thresholding-based, clustering-based, region-based, and grid-based methods.

2.3.1 Threshold-Based Segmentation

For portioning of the image, slicing techniques and histogram thresholding techniques are used. These techniques can be directly applied to an image; however, they can also be combined with post- and pre-processing techniques. Thresholding is possibly the most widely used technique for the segmentation of an image.

An algorithm that uses a histogram for probability values to define the optimal threshold value is one that maximizes the variance between classes, and was proposed by Otsu [24].

2.3.1.1 Otsu' Algorithm

1. Chose an initial approximate for T.
2. Segment the image through T. As a result, two groups of pixels will be produced: G_1 containing all pixels having values of intensity $>=T$; and G_2 containing all pixels having intensity values $<T$.
3. Calculate the average values of intensity $\mu 1$ and $\mu 2$ for the pixels present in regions G_1 and G_2, respectively.
4. Calculate a new value of the threshold.
5. Repeat steps 2−4 until the difference in the value of T in consecutive iterations becomes less than the predefined parameter T_0.

2.3.1.2 Ant Colony Optimization-Based Multilevel Thresholds Selection

The process of multilevel segmentation by using a genetic algorithm fails to provide sufficient results. To solve this problem through ACO, the basic steps of ACO are described in detail.

Ants exist in highly organized colonies in which they interact with others in perfect harmony. As stated earlier, the pheromone which ants leave on the ground while traveling decreases with time due to evaporation. Moreover, the amount of pheromone deposited by ants is dependent on the number of ants who use that trial. To find the shortest possible path between their colony and the source of food, ants adopt a unique collective organization technique which can be simulated to solve both static as well as dynamic optimization problems [25]. Ants can find the shortest path between their nest and food source without using visual information and, therefore, do not possess any global world model, and are adaptable to the environmental changes.

The probability of an ant choosing a particular path over another is presided by the quantity of pheromone on the prospective path of interest. As the quantity of pheromone on a path evaporates with time, the ants adopting the shorter path will return first to their nest with food. The most pheromones are present in shorter routes since the paths have fresh pheromones

which have not yet evaporated and, hence, will be more attractive for ants returning to the source of food. This possibility (even though small) allows other trails to evaporate which is helpful as it allows discovery of alternate, or shorter, pathways or fresh food sources. As the pheromone trail evaporates with the passage of time, the trail will become less detectable. Hence, the longer trails are less attractive which benefits the ant colony.

2.3.1.3 Algorithm for Ant Colony Optimization

1. Ants travel around the colony to find sources of food.
2. After finding a source, the ant returns back to its nest.
3. During traversing, ants leave a trail of pheromones.
4. The follower ants of the first ant go after the pheromones deposited by the first ant.
5. As a result of this transaction, the deposition of the pheromone on the trail will be strengthened.
6. The quantity of pheromones in each traversal will evaporate.
7. If there are two paths to get to the same source of food, the ant finds the shortest path between their nest and food with the help of the fresh pheromones.

2.3.2 Edge-Based Segmentation

Edge detection is a technique used to recognize and locate severe discontinuities in an image. This technique represents the detected edges in images as the object boundaries which are utilized to recognize the objects. There are several ways to execute edge detection using different types of detectors, for example, Canny, ACO, Robert, Sobel, Prewitt, etc. However, an ACO-based edge detection technique has been investigated, which will be discussed next.

2.3.2.1 Ant Colony Optimization-Based Edge Detection Initialization

The initialization of parameters α and β takes place at the beginning of the experiment. The probable heuristic data collected. The calculation of some ants is done as K:$\sqrt{M_1 \cdot M_2}$ where M_2 is the width and M_1 is the image length. All the K number of ants are spread on a 2D image such that almost one ant is present on each pixel. Each image pixel contains a node and an initial pheromone matrix value, that is, a constant value is assigned to τ^0 [26].

2.3.2.2 Ant Colony Optimization-Based Structuring Process

By picking from a cluster of K ants at each construction step, one of the ants is moved L steps on the image (say I). The ant (A_k) supposedly moves from

a node (*l, m*) to a neighboring node (*i, j*) by the probabilistic transition matrix as defined in Eq. (2.5).

$$P^{n}_{(l,m)(i,j)} = \frac{\left(\tau^{n-1}_{i,j}\right)^{\alpha}.\left(\eta_{i,j}\right)^{\beta}}{\sum(i,j)\varepsilon\Omega_{(l,m)}\left(\tau^{n-1}_{i,j}\right)^{\alpha}.\left(\eta_{i,j}\right)^{\beta}} \tag{2.5}$$

where $\tau^{n-1}_{i,j}$ denotes the value of a pheromone at a particular node (*i, j*) and it has been established the neighboring node of a (*l, m*) node is $\Omega_{(l,m)}$. In other words, $\Omega_{(l,m)}$ represents all the pixels which can be present in the 8-*i, j* neighborhood of a pixel positioned at (*l, m*). The metaheuristic data of a node (*i, j*) is represented by η. To quantify the heuristic data [26,27], the configuration which is typically local at each level (i.e., *i, j*) is defined as:

$$\eta_{i,j} = \frac{1}{z} V_{c(I_{i,j})} \tag{2.6}$$

where *Z* is known as the normalization factor utilized to isolate error and is defined as:

$$Z = \sum 1{:}M_1 \sum 1{:}M_2 V_{c(I_{i,j})} \tag{2.7}$$

For Eq. (2.7), $I_{i,j}$ signifies the intensity value a pixel (*i, j*) present in the image (*I*).

The disparity in the intensity values of an image's pixels typically depends on "*c*" which is formed by a group of similar pixels. It has been established that the group of pixels constitute the function $V_{c(I_{i,j})}$.

The function $V_c(I_{i,j})$ mainly depends on its neighboring pixel groups (*c*), which is defined in Eq. (2.8).

$$\begin{aligned} V_c(I_{i,j}) = f(&|I_{i-2,j-1} - I_{i+2,j+1}| + |I_{i-2,j+1} - I_{i+2,j-1}| + \\ &|I_{i-1,j-2} - I_{i+1,j+2}| + |I_{i-1,j-1} - I_{i+1,j+1}| + |I_{i-1,j} - \\ &I_{i-1,j} + I_{i-1,j+1} - I_{i-1,j-1} + I_{i-1,j+2} - I_{i-1,j-2} + I_{i,j-1} - I_{i,j+1}) \end{aligned} \tag{2.8}$$

Eq. (2.8) provides insight into the function which guarantees that the likelihood of the sharp-turn shapes in the image is less than that of the small-angle turns. Therefore every ant in a particular colony has the propensity to travel in the forward direction.

Modeling of Eq. (2.8) for bringing out changes in the respective shapes can be performed as follows:

$$f(x) = \lambda x \text{ for } x \geq 0 \tag{2.9}$$

$$f(x) = \lambda x^2 \text{ for } x \geq 0 \tag{2.10}$$

$$f(x) = \sin\left(\frac{\pi x}{2\lambda}\right) \quad 0 <= x <= \lambda \tag{2.11}$$

$$\text{else } f(x) = \frac{\pi x \sin\left(\frac{\pi x}{\lambda}\right)}{\lambda} \tag{2.12}$$

The parameter λ incorporated in these equation functions indicates that this function is responsible for adjusting the respective shapes of each function.

2.3.2.3 Ant Colony Optimization-Based Updating Process

This section proactively emphasizes the updating process which involves the steps moved through by each ant after each neighboring ant, followed by all ants on each construction step. An effort has been made to alter only a single updating process in the ACO algorithm to yield a binary image with missing data [28].

After the movement of each ant, the updating process updates the pheromone matrix which is given in Eq. (2.13).

$$\tau_{i,j}^{n-1} = \frac{(1-\rho), \tau_{i,j}^{n-1} + \rho,\ \Delta_{i,j}^{k} \text{ if } (i,j) \text{ is visited by } k\text{th ant}}{\tau_{i,j}^{n-1},\ \text{otherwise}} \tag{2.13}$$

where ρ is signifies the evaporation rate and a heuristic matrix is used to determine $\Delta_{i,j}^{k}$, that is, $\Delta_{i,j}^{k} = \eta_{i,j}$

The heuristic data from a heuristic matrix is gradually added into the memory of the ants. The second update is carried out at the conclusion of each building step (after the movement of all the (K) ants within the construction step. The movement of all the ants at the conclusion of the construction step yields Eq. (2.14).

$$\tau^{n} = (1-\psi),\ \ \tau^{n-1} + \psi\tau^{n} \tag{2.14}$$

where ψ is known as the pheromone decay coefficient.

The step demonstrated in Eq. (2.14) signifies the pheromone matrix which is updated in this particular instance with the consideration of the pheromone matrix and the decay coefficient.

2.3.2.4 Decision Process

A threshold value (i.e., T) is typically used on τ^{N} (pheromone matrix) to obtain the edge information for a particular image. The iterative method has been proposed by various researchers for computing the threshold in ant ACO algorithms. It has been reported that a normalized intensity value (typically in a range of [0,1]) is adapted to carry out the conversion of the intensity image to a binary image [29]. Concerning the initial threshold value, the segmentation of the histogram into two distinct parts takes place. Computation of the mean values (for gray values) related to the foreground pixels along with sample mean (for the gray values) related to the pixels' background is carried out. Therefore this current threshold value can be

regarded as the average value of the two specimens. The starting threshold value (T^0) is ultimately considered as the pheromone matrix's mean value. Every index value in the pheromone matrix is signified as either above the initial threshold value or below the threshold value. Based on these two classifications, the average of all the mean values is computed that is considered as a (new) fresh threshold value. This averaging process is carried out until the threshold value approaches a constant.

Step 1: Initialization of $T^{(0)}$ as given in Eq. (2.15)

$$T^{(0)} = \frac{\Sigma_{i=1:M_1}\Sigma_{j=1:M_2}\tau_{i,j}^{(N)}}{M_1 M_2} \tag{2.15}$$

Step 2: Separation of $\tau^{(N)}$ (pheromone matrix) into two separate classes using $T^{(l)}$ in which one of the classes contains entries of τ with values less than the threshold value of $T^{(l)}$, while the other class contains the remaining entries of τ. Concerning these categories, the mean calculation for the above categories is carried out as follows:

$$m_L^{(l)} = \frac{\Sigma_{i=1:M_1}\Sigma_{j=1:M_2} g_T^L(l)(\tau_{i,j}^{(N)})}{\Sigma_{i=1:M_1}\Sigma_{j=1:M_2} h_T^L(l)(\tau_{i,j}^{(N)})} \tag{2.16}$$

$$m_U^{(l)} = \frac{\Sigma_{i=1:M_1}\Sigma_{j=1:M_2} g_T^U(l)(\tau_{i,j}^{(N)})}{\Sigma_{i=1:M_1}\Sigma_{j=1:M_2} h_T^U(l)(\tau_{i,j}^{(N)})} \tag{2.17}$$

where

$$g_{T^{(l)}}^L(x) = \{x, \text{if } x \le T^{(l)} \text{ otherwise } 0 \tag{2.18}$$

$$h_{T^{(l)}}^L(x) = \{x, \text{if } x \le T^{(l)} \text{ otherwise } 0 \tag{2.19}$$

$$g_{T^{(l)}}^U(x) = \{x, \text{if } x \le T^{(l)} \text{ otherwise } 0 \tag{2.20}$$

$$h_{T^{(l)}}^U(x) = \{x, \text{if } x \le T^{(l)} \text{ otherwise } 0 \tag{2.21}$$

Step 3: Iteration index is maintained as $l = 1 + 1$ which ultimately updates the threshold as given in Eq. (2.22).

$$T^{(l)} = \frac{m_L^{(l)} + m_U^{(l)}}{2} \tag{2.22}$$

Step 4: The new threshold value ($T^{(l)}$) is assessed for similarity with $T^{(n-1)}$. If the similarity index is zero, then step 2 is repeated; otherwise the procedure is ended and the threshold value is documented.

Consequently, the determination of the definite edge ($E_{i,j}$) in an image is carried out using the pixel (i, j).

$$E_{i,j} = \begin{cases} 1 \text{ if } \tau_{i,j}^{(N)} \geq T^{(l)} \\ 0 \end{cases} \tag{2.23}$$

where $E_{i,j}$ denotes the image obtained at pixel (i, j)

A variety of parameters and their values employed in the above calculations are listed below:

1. $\alpha = 1$, that is, weighting factor for information of pheromone matrix.
2. $\beta = 0.1$, that is, weighting factor for information of heuristic matrix.
3. $N = 4$, that is, sum of all construction processes.
4. $\Psi = 0.05$, that is, pheromone decay coefficient.
5. $L = 40$, that is, movement steps of an ant.
6. $\lambda = 1$, that is, the adjustment factor.
7. $\rho = 0.1$, that is, evaporation rate.
8. $\Omega = 8$, that is, connectivity with neighborhood.
9. $\tau_{\text{init.}} = 0.0001$, that is, the primary value of every constituent in pheromone matrix.
10. $K = \sqrt{M_1 M_2}$, that is, number of the total ants (the image size).

2.4 EVALUATION OF SEGMENTATION TECHNIQUES

Assessment is essential in deciding which segmentation algorithm is suitable to be selected. For extraction of the boundary, the edge detection is assessed by using root-mean-square-error (RMSE), mean-square error (MSE), peak signal-to-noise ratio (PSNR), and signal-to-noise ratio (SNR) [30–32].

2.4.1 Mean-Square Error

MSE value denotes the average difference of the pixels all over the image. A higher value of MSE designates a greater difference amid the original image and processed image. Nonetheless, it is indispensable to be extremely careful with the edges. The following equation provides a formula for calculation of the MSE.

$$\text{MSE} = \frac{1}{N} \sum \sum (E_{ij} - o_{ij})^2 \tag{2.24}$$

here, N is the image size, O is the original image, whereas, E is the edge image.

2.4.2 Root-Mean-Square-Error

The RMSE is extensively used for measuring the differences between values forecasted by an estimator, or model, and the values that are observed. The RMSE aggregates the magnitudes of the predictions errors several times into

a single distinct measure of predictive power. It is a measure of precision. The RMSE value can be calculated by taking the square root of MSE.

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum\sum(E_{ij} - o_{ij})^2} \tag{2.25}$$

2.4.3 Signal-to-Noise Ratio

SNR describes the total noise present in the output edge detected in an image, in comparison to the noise in the original signal level. SNR is a quality metric and presents a rough calculation of the possibility of false switching; it serves as a mean to compare the relative performance of different implementations [33−35]. SNR is estimated by Eq. (2.26):

$$\mathrm{SNR} = \left[\frac{\Sigma_i \Sigma_j (E_{ij})^2}{\Sigma_i \Sigma_j (E_{ij} - o_{ij})^2}\right] \tag{2.26}$$

2.4.4 Peak Signal-to-Noise Ratio

The PSNR calculates the PSNR ratio in decibels amid two images. We often use this ratio as a measurement of quality between the original image and the resultant image. The higher the value of PSNR, the better will be the quality of the output image. For calculating the PSNR, MSE is used. We calculate the PSNR by using Eq. (2.27).

$$\begin{aligned} PSNR &= 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \\ &= 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \\ &= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE) \end{aligned} \tag{2.27}$$

Here, MAX_I is the maximum possible pixel value of the image

2.5 EXPERIMENTS AND RESULTS

Boundary extraction was initially carried out in image segmentation process. The operators of edge detection were employed on the real iris and brain MRI images, while some of the test experiments were also performed on images obtained from different image datasets, for example, Berkley dataset, Caltech dataset, etc. [36,37]. The experimental results exhibit the visual output of the

edge detection. Two types of images (brain MRI and iris) are experimentally shown next.

2.5.1 Ant Colony Optimization-Image-Segmentation Using the Isula Framework

A Java Program (for ACO-image-segmentation) was implemented to identify image segments using the ACO algorithm. Segmentation is a two-phase process. Initially, the image (brain MRI/iris) is extracted using the Ant Colony algorithm followed by the application of a clustering procedure. The ACO algorithm for extraction of image segments is centered on Max−Min Ant System which is implemented using the Isula Framework.

The implemented ACO process has the following characteristics:

1. The solution components are Clustered Pixels, that is, each image pixel (of brain MRI or iris image) is assigned to one of the three clusters.
2. The available Ant class in the Isula framework was extended to back the image-clustering class.
3. The ants used for the particular types of problems possessed memory concerning the current position in an image along with the Map for storage of Cluster.
4. The determination of the Solution Quality is quite expensive; therefore this was incorporated as an instance variable.
5. These Ants were constructed so that they only considered certain pixels (cerebrum or iris) in the images. That data was provided by the earlier binary thresholding algorithms.
6. The Max−Min Ant Systems' policies accessible in Isula were reprocessed with minor tailoring. These policies include node selection and pheromone update.

Boundary extraction was initially carried out in image segmentation process. The operators of edge detection were employed on the images obtained from a segmentation image dataset, for example, Berkley dataset, Caltech dataset, Corel dataset, etc. The experimental results exhibit the visual output of the edge detection. Two types of images (brain MRI and iris) are experimentally shown next.

2.5.2 Performance Testing Ant Colony Optimization Image Segmentation Algorithm

Various test images were initially used to explore the ACO Image Segmentation Algorithm's (i.e., ACO-ISA) performance. The key parameters were given following values throughout the experiment.

A total number of ants was around 20% of the sum of pixels on a particular image, that is, $\eta = 0.3$; $K = 0.985$, while the maximum number of iterations was 200. These parametric values seem to be suitable for

approximately all of the images, and can be altered with less effective (minor) difference in the resulting quality [38–40]. However, the other experimental parameters were selected in accordance with different image types, because of their close association with the histogram of various types of images. Examples of the performance of ACO segmentation along with its comparison with conventional edge detection techniques are illustrated in Figs. 2.5 and 2.6.

2.5.3 Application of Ant Colony Optimization on Segmentation of Brain MRI

The performance of the ACO algorithm was assessed by its implementation on the real brain MRI. Processing of the tests was performed on the high-speed workstation. Fig. 2.3 displays the original brain MR images (256×256) which are 8-bit grayscale having the intensity values in the range of 0–255. The experimental study offered proof of the efficacy of the proposed method. The results are analyzed in the following section.

The results of image segmentation carried out using the indigenously developed algorithm for a different set of iterations are shown in Fig. 2.4. Considering the time of iterations, the segmentation of the WM takes place conveniently after 200 iterations for a T_1-weighted image. Additionally, the segmentation of the GM also takes place easily after 200 iterations for a T_2-weighted image. The same MRI image was investigated using the metaheuristic techniques for image segmentation which included the algorithms based on canny, artificial bee, Robert, Sobel, and Prewitt detectors. Figs. 2.4 and 2.5 exhibit the comparison of the image segmentation results concerning the number of iterations and type of detectors. It can be observed that our proposed approach (using an ACO algorithm) provides more accurate results than other conventional techniques.

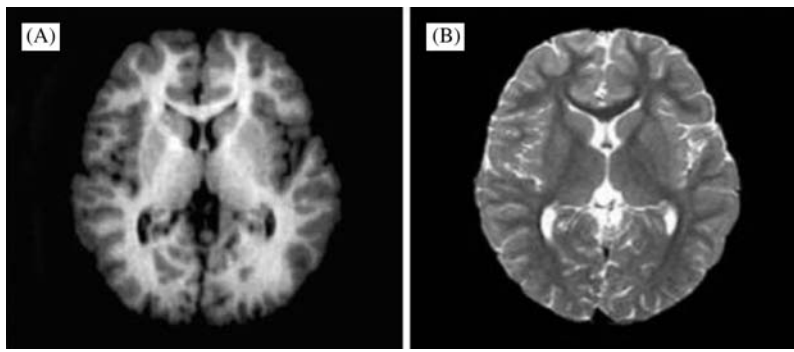

FIGURE 2.3 The original brain MRI images. (A) Skull-stripped (or T_1)-weighted image; (B) skull-stripped (or T_2)-weighted image.

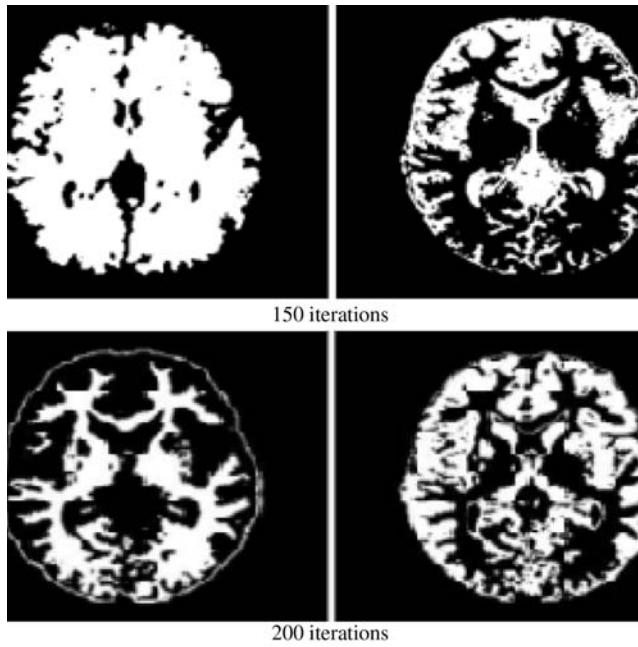

FIGURE 2.4 The image segmentation results obtained at different iterations utilizing the ACO algorithm. In both cases, the first column illustrates the T_1 image results while the second column demonstrates the T_2 image results.

2.5.4 Ant Colony Optimization-Image Segmentation on Iris Images

The performance of an ACO-based image segmentation algorithm for iris images is demonstrated in Fig. 2.6. It can be easily observed that the pigmentation and collarette regions are distinct under the ACO-ISA based detector. It also establishes that the ACO-ISA processed image depicts better quality for further application of region edging techniques.

In our experimental processes, the image segmentation ability of ACO-ISA was investigated on iris areas of three types of typical textures, that is, normal texture, the radii Solaris, and the hyperpigmentation textures. After the processing of ACO-IRIS and ACO-TRA (texture representation ability, the alignment of each region to 3×3 sub-zones took place followed by the calculation of histogram for each sub-zone. The comparison of two regions was made by taking the mean of histogram distances for all corresponding pairs of sub-zones. A sum of 900 sample regions was involved for each kind of the texture. The average distances (interclass distance) are demonstrated in Table 2.1. It is obvious from the results that different texture regions are well-distinguished for the ACO segmentation technique compared to the

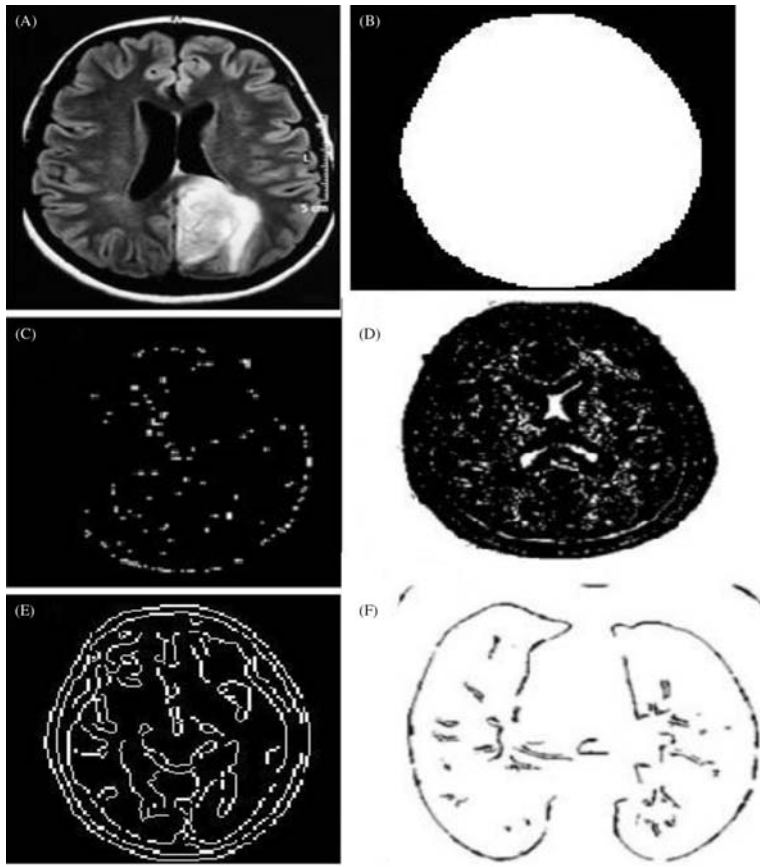

FIGURE 2.5 Brain MRI edge detection information for different image segmentation and image edge detection methods. (A) Original image of brain MRI. (B) Sobel edge detector's performance (threshold = 0.55; iterations = 200). (C) The Prewitt edge detector's performance (threshold = 0.26; iterations = 200). (D) The Robert edge detector's performance (threshold = 0.19; iterations = 200). (E) Canny edge detector's performance (threshold = 0.55; iterations = 200). (F) ACO edge detector's performance (threshold = 0.55; iterations = 200).

results obtained from other techniques. Additionally, we also designed a self-organized feature map (SOFM) neural network a tool for texture recognition, and around 95.1% accurate recognition rate was attained. This unveiled that our proposed technique has the potential to be useful in automatic disease diagnosis systems using different types of images (iris, MRI, etc.).

To scrutinize the performance of the ACO algorithm, real iris and brain MRI images were used in the experiments. The output results obtained from traditional edge detection techniques (Figs. 2.5B–F and 2.6B–E) exhibit the results with 200 iterations and the parameters were adjusted to $\beta = 3.5$, $\eta = 0.08$, $v = 0.017$, $p = 1.3$; while many ants were 10% of all the pixels.

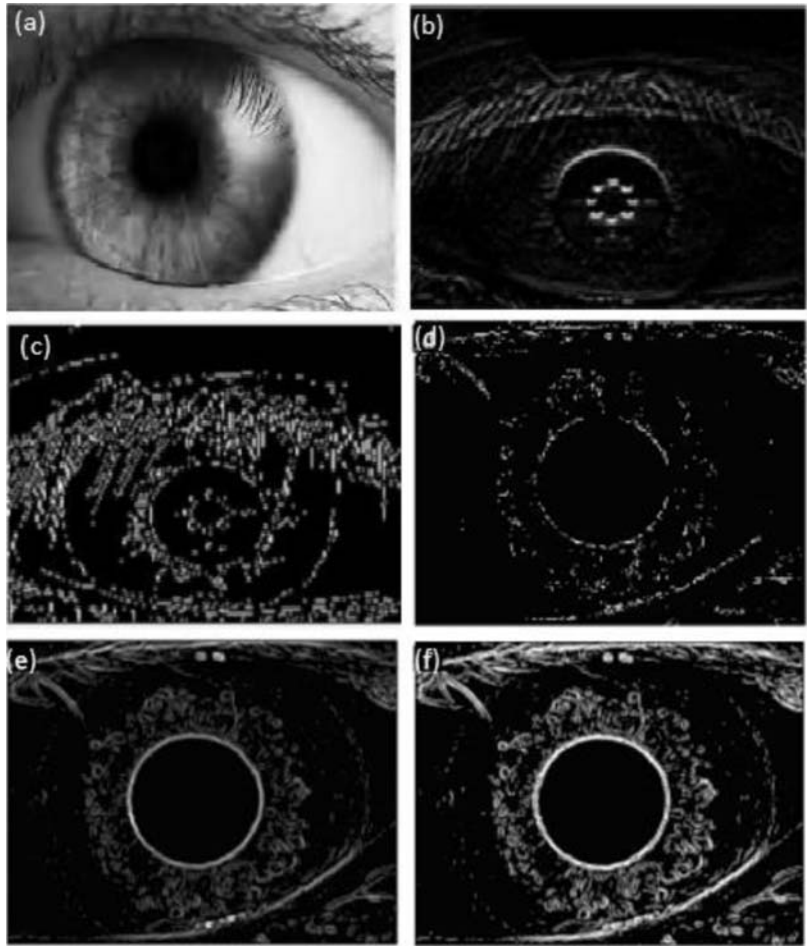

FIGURE 2.6 (A) Original iris image. (B) Performance of Sobel detector (at 200 iterations). (C) Performance of Canny detector at 200 iterations. (D) Performance of Robert edge detection at 200 iterations. (E) Performance of Prewitt detector at 200 iterations. (F) Performance of ACO detector at 200 iterations.

TABLE 2.1 Interclass Spaces Between Different Iris Regions

Interclass Distance	Radii Solaris	Hyperpigmentation	Normal Region
Hyperpigmentation	13.67	–	84.53
Normal region	97.95	84.47	–
Radii Solaris	–	13.62	97.87

TABLE 2.2 Time Passed (in Seconds) by Traditional Edge Detection Techniques as Compared to ACO

Image Type	Canny	Robert	Sobel	Prewitt	Artificial Bee	ACO
Brain MRI	0.285	0.868	0.084	0.734	0.688	114.75
IRIS	2.67	0.085	0.078	0.067	0.078	60.08

Higher values of v and smaller values of β lead to higher probability of probing new paths. Therefore for the deposition of further pheromones in a digital habitat, the number of iterations must be increased. Experimental results for the ACO-based technique showed promising results. The process times of ACO has been improved (shortened) compared to the processing times of ACO employed in former research [41]. However, ACO still requires more time than conventional edge detection techniques, as demonstrated in Table 2.2.

Our current experiment involves the experimentation of five different images (five each of iris and brain MRI images). Figs. 2.5A and 2.6A display the original images while Figs. 2.5B–F and 2.6B–E show the results obtained from conventional algorithms (Canny, Robert, Prewitt, Sobel, etc.). Both of the image sets processed with the ACO technique exhibited superior performance compared to the images processed with conventional algorithms regarding efficiency and effectiveness. To validate the effectiveness of ACO results, experiments were conducted on various other (real and synthetic) images. The segmentation (using images other than iris and brain) results were filtered by the Robert, Canny, Sobel, Prewitt, and ACO edge detectors using the Java framework in the MATLAB toolbox. Number of ants in digital habitat were 10% of the total number of pixels and essential parameters were $\delta = 0.03$, $\beta = 3.5$, $v = 0.017$, $p = 1.3$, $\lambda = 0.5$, and $f_i = 0.09$. Tables 2.3 and 2.4 depict the difference in performance for various edge detectors.

2.5.5 Comparison of Results

ACO metaheuristics are specialized in solving the optimization problems in every iteration step by vaporizing and updating the pheromone density. In Figs. 2.4–2.6 ACO image segmentation with a different number of iterations using different techniques are illustrated. Fig. 2.4 depicts the brain MRI images segmented with 100 and 200 iterations using the ACO technique. Increasing the number of iterations improves the segmentation performance of ACO using a lower number of ants, as depicted in Fig. 2.4. The decrease in some ants to 1% of the total number of pixels and the increase in the number of iterations from 20 to 200 results in significantly improved segmented images with a higher probability of edge detection.

TABLE 2.3 Edge Operator (Detector) Evaluation Table for Brain MRI Image

	Sobel	Prewitt	Robert	Artificial Bee	Canny	ACO
RMSE	5.5760	5.5826	5.9294	4.6394	4.6394	3.3519
SNR	43.0944	42.8559	29.2034	68.2989	68.2989	84.3721
PSNR	33.2047	33.1944	32.6719	34.8026	34.8026	37.6483

TABLE 2.4 Edge Operator (Detector) Evaluation Table for IRIS Image

	Sobel	Prewitt	Robert	Artificial Bee	Canny	ACO
RMSE	2.2728	2.2774	2.2940	2.1175	2.1175	1.8119
SNR	43.8058	43.4136	42.2534	55.4086	55.4086	67.5980
PSNR	41.0000	40.9827	40.9195	41.6154	41.6154	43.7026

The comparison of ACO performance with conventional edge detection techniques is shown in Figs. 2.5 and 2.6, respectively. Fig. 2.5 shows the comparison of brain MRI image segmentation using algorithms of Canny, Artificial Bee, Sobel, Robert, Prewitt, and ACO techniques. The traditional edge detection techniques show unclear edges with poor segmentation. However, the results of the ACO technique depict the compensation of the results regarding edge detection for suitable image segmentation [17].

The ACO method utilized two thresholds to identify weak and strong edges. It only incorporated those weak edges in the output which were linked to strong edges. Consequently, the technique is accurate and robust to noise with higher probability of detecting the true weak/strong edges.

Additionally, the proposed method exploits the movement of a definite number of ants on the image based on the localized disparity in the image's intensity value. The localized disparity in the image's intensity value is used to develop a pheromone matrix, which provides the edge data of the image.

The statistical comparison of the ACO technique with other traditional techniques is carried out using the operator parameters which include RMSE, SNR, and PSNR. Tables 2.3 and 2.4 show the comparative values of the operator parameters for five iris and five brain MRI images evaluated through six algorithms (Canny, Sobel, Robert, Artificial Bee, Prewitt, and ACO). RMSE provides the difference between the values of the edge-detected pixels and the original pixels for a particular technique. Low RMSE signifies less difference between the values of the original and processed images, which indicates the

accuracy of the output image. SNR is a relative value of the signal concerning noise and must be high for the accuracy of the output. On the other hand, the PSNR peak should be higher for enhanced quality.

Table 2.3 shows the comparative results for RMSE, SNR, and PSNR values obtained by different detector types (algorithms) used on MRI images. Although the ACO algorithm offers more clear edges, a comparative study of different edge detection techniques was carried out for validating the boundary extraction of the images. Table 2.3 shows that RMSE for ACO is at a minimum (3.7519) which indicates higher accuracy. On the other hand, comparatively higher values of SNR and PSNR (82.3721 and 36.6483, respectively) depict higher image quality for ACO in MRI images.

Table 2.4 shows the comparative results for RMSE, SNR, and PSNR values obtained by different detector types (Canny, Robert, Sobel, Prewitt, ACO, etc., algorithms) used on iris images. Although the ACO algorithm offers more clear edges, a comparative study of the different edge detection techniques was carried out for validating the boundary extraction of the images. Table 2.4 shows that RMSE for ACO is at a minimum (1.9119) which indicates higher accuracy. On the other hand, comparatively higher values of SNR and PSNR (65.4980 and 42.5026, respectively) depict higher image quality for ACO in the iris images.

The following figures display the graphs showing the comparison of RMSE, SNR, and PSNR values for 10 (5 iris and 5 brain MRI) images.

Fig. 2.7 shows a graph for depicting the RMSE performance of different algorithms using five iris and five brain MRI images. It can be observed that ACO peaks are smallest among all, which indicates lowest RMSE. On the other hand, Robert and Sobel's detectors exhibit highest RMSE values, which indicate lower accuracy in image segmentation. Similarly, Figs. 2.8 and 2.9 show the SNR and PSNR for different algorithms applied on iris and brain MRI images. Both graphs display higher ACO peaks, which indicate higher

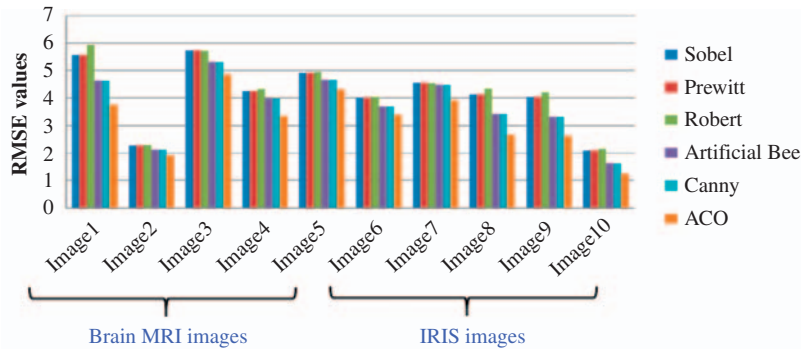

FIGURE 2.7 Comparison of image edge detection methods using root-mean-square-error (RMSE).

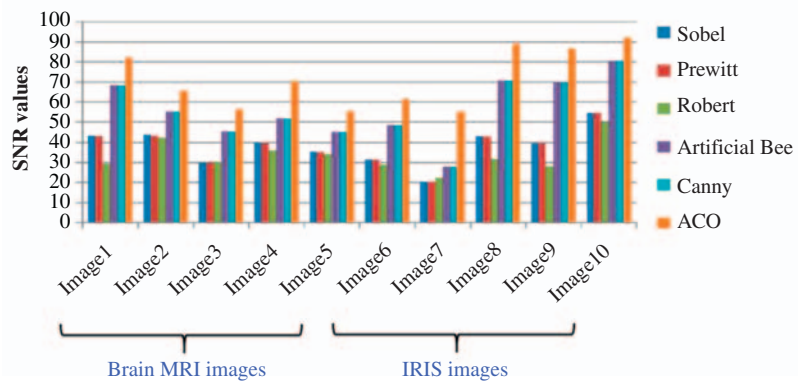

FIGURE 2.8 Comparison of image edge detection methods using signal-to-noise-ratio (SNR).

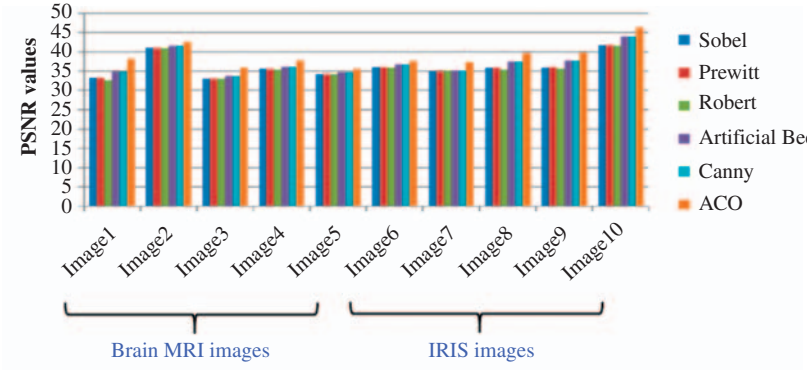

FIGURE 2.9 Comparison of image edge detection methods using peak-signal-to-noise ratio (PSNR).

segmentation and edge detection quality for the ACO-based technique. On the other hand, Robert and Prewitt's techniques show lowest peaks, which indicate their poor segmentation characteristics.

An adaptive edge detector is essential for obtaining a robust solution which is adaptable to the fluctuating noise levels of the images and helps to distinguish valid contents of images from visual defects induced by noise. The ACO's performance heavily depends on the changeable parameters (e.g., σ, typically known as the standard deviation of the Gaussian filter) and threshold values (e.g., T_1 and T_2). σ also regulates the Gaussian filter's size. Higher values of σ result in larger sizes of the Gaussian filter. This phenomenon implies higher and more blurring levels, essential for noisy images along with detection of the larger edges. The results and graphs shown in this chapter depict the higher effectiveness of the indigenously modified ACO algorithm proposed. The tuning of the parameters has yielded excellent results (better

than the previous results quoted in the literature) concerning lower RMSE and higher SNR and PSNR, as indicated in Tables 2.3 and 2.4. Our main aim was to improve the image segmentation with enhanced image edge detection by lowering the error margin and enhancing the image quality. The task has been accomplished using the Java framework and modified ACO algorithm.

2.6 CONCLUSION

This chapter presents a novel ACO-based algorithm for image segmentation using the Java framework. The ACO method is inspired by the food-searching mechanisms in ant colonies. Experimental parameters for ACO were altered and the results were compared with the conventional segmentation and edge detection techniques. The study was performed on real iris and brain MRI images by the application of different algorithms (Canny, Robert, Sobel, Prewitt, and ACO). Five images each of iris and brain MRI were used to investigate six different segmentation algorithms. The images were iterated at different iterations (20−100−200). However, all the techniques exhibited better results at higher iterations ($\sim$200). On the other hand, the number of ants was increased at lower iterations to observe the effect of increasing the number of ants. An increase in the number of ants resulted in better performance at lower iterations.

A comparison of different results was made using MRI and iris images. The results unveil that unlike conventional algorithms, the ACO algorithm offers superior results concerning image segmentation and edge detection. The images shown in Figs. 2.4−2.6 exhibit the comparison of different segmentation techniques (with different edge detectors). It is obvious from the images that the ACO algorithm provides better edge detection and image as compared to other techniques. Moreover, key operator parameters, such as RMSE, SNR, and PSNR, were also analyzed for all six algorithms. Tables 2.3 and 2.4 and Figs. 2.7−2.9 show the superior and improved performance of the ACO-based algorithm over conventional techniques. The basic drawback of the ACO-based detector is its higher time consumption during the algorithm application. The alteration in parameters has significantly enhanced the performance of the ACO technique by reducing the process time up to 50% of the times achieved previously.

The robustness of the ACO technique against image noise was investigated using real brain MRI and iris images. Experimental results exhibited efficient (significantly superior) performance of the ACO approach in comparison with traditional edge detection approaches. This chapter presents the improved results for solving image segmentation and edge detection problems using a modified (upgraded) ACO algorithm which yields superior results than those offered by the traditional approaches, that is, Canny, Prewitt, Sobel, Robert, etc.

REFERENCES

[1] S. Paul, S. Datta, S. Das, Rough-fuzzy collaborative multi-level image thresholding: a differential evolution approach, Mendel 2015, Springer, Cham, 2015, pp. 329–341.

[2] A. Singla, S. Patra, A fast automatic optimal threshold selection technique for image segmentation, Signal Image Video Process. 11 (2) (2017) 243–250.

[3] E.H. Guerrout, S. Ait-Aoudia, D. Michelucci, R. Mahiou, Hidden Markov random field model and BFGS algorithm for brain image segmentation, Proceedings of the Mediterranean Conference on Pattern Recognition and Artificial Intelligence, ACM, New York, NY, 2016, November, pp. 7–11.

[4] X. Li, L. Lu, L. Liu, G. Li, X. Guan, Cooperative spectrum sensing based on an efficient adaptive artificial bee colony algorithm, Soft. Comput. 19 (3) (2015) 597–607.

[5] C.H. Tung, Z.L. Wu, Binarization of uneven-lighting image by maximizing boundary connectivity, J. Stat. Manage. Syst. 20 (2) (2017) 175–196.

[6] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, A survey of multiobjective evolutionary clustering, ACM Comput. Surv. (CSUR) 47 (4) (2015) 61.

[7] M.G. Pérez, A.S. Calle, A.B. Moreno, E. Nunes, V. Andaluz, A multi-level thresholding method based on histogram derivatives for accurate brain MRI segmentation, Rev. Politéc. 35 (2) (2015) 82.

[8] B. Minnaert, D. Martens, M. De Backer, B. Baesens, To tune or not to tune: rule evaluation for metaheuristic-based sequential covering algorithms, Data Min. Knowl. Discov. 29 (1) (2015) 237–272.

[9] I.A. Hodashinsky, D.Y. Minina, K.S. Sarin, Identification of the parameters of fuzzy approximators and classifiers based on the cuckoo search algorithm, Optoelectron. Instrum. Data Process. 51 (3) (2015) 234–240.

[10] S.L. Jui, S. Zhang, W. Xiong, F. Yu, M. Fu, D. Wang, et al., Brain MRI tumor segmentation with 3D intracranial structure deformation features, IEEE Intell. Syst. 31 (2) (2016) 66–76.

[11] M. Brajović, V. Popović-Bugarin, I. Djurović, S. Djukanović, Post-processing of time-frequency representations in instantaneous frequency estimation based on ant colony optimization, Signal. Process. 138 (2017) 195–210.

[12] K. Singh, R. Kapoor, S.K. Sinha, Enhancement of low exposure images via recursive histogram equalization algorithms, Optik: Int. J. Light Electron Opt. 126 (20) (2015) 2619–2625.

[13] S.H. Lim, N.A.M. Isa, C.H. Ooi, K.K.V. Toh, A new histogram equalization method for digital image enhancement and brightness preservation, Sig. Image Video Process. 9 (3) (2015) 675–689.

[14] M. Mavrovouniotis, S. Yang, Training neural networks with ant colony optimization algorithms for pattern classification, Soft. Comput. 19 (6) (2015) 1511–1522.

[15] M.M.T. Alobaedy, Hybrid ant colony system algorithm for static and dynamic job scheduling in grid computing, Doctoral dissertation, Universiti Utara Malaysia, 2015.

[16] G.G. Wang, A.H. Gandomi, X.S. Yang, A.H. Alavi, A new hybrid method based on krill herd and cuckoo search for global optimisation tasks, Int. J. Bio-Inspir. Comput. 8 (5) (2016) 286–299.

[17] T. Sağ, M. Çunkaş, Color image segmentation based on multiobjective artificial bee colony optimization, Appl. Soft. Comput. 34 (2015) 389–401.

[18] G. Greenfield, P. Machado, Ant-and ant-colony-inspired alife visual art, Artif. Life 21 (3) (2015) 293–306.

[19] M. Salari, M. Reihaneh, M.S. Sabbagh, Combining ant colony optimization algorithm and dynamic programming technique for solving the covering salesman problem, Comput. Ind. Eng. 83 (2015) 244–251.
[20] E.G. Talbi, Combining metaheuristics with mathematical programming, constraint programming and machine learning, Ann. Oper. Res. 240 (1) (2016) 171–215.
[21] T.X. Lin, H.H. Chang, Medical image registration based on an improved ant colony optimization algorithm, Int. J. Pharma Med. Biol. Sci. 5 (1) (2016) 17.
[22] M.D. Toksari, A hybrid algorithm of ant colony optimization (ACO) and iterated local search (ILS) for estimating electricity domestic consumption: case of Turkey, Int. J. Electr. Power Energy Syst. 78 (2016) 776–782.
[23] I. Bloch, Fuzzy sets for image processing and understanding, Fuzzy Sets Syst. 281 (2015) 280–291.
[24] A. Nayyar, R. Singh, Simulation and performance comparison of ant colony optimization (ACO) routing protocol with AODV, DSDV, DSR routing protocols of wireless sensor networks using NS-2 simulator, Am. J. Intel. Syst. 7 (1) (2017) 19–30.
[25] P. Shao, W. Shi, P. He, M. Hao, X. Zhang, Novel approach to unsupervised change detection based on a robust semi-supervised FCM clustering algorithm, Remote Sens. 8 (3) (2016) 264.
[26] T.M. Tuan, A cooperative semi-supervised fuzzy clustering framework for dental X-ray image segmentation, Expert Syst. Appl. 46 (2016) 380–393.
[27] M.A.J. Ghasab, S. Khamis, F. Mohammad, H.J. Fariman, Feature decision-making ant colony optimization system for an automated recognition of plant species, Expert Syst. Appl. 42 (5) (2015) 2361–2370.
[28] Y. Khaluf, S. Gullipalli, 2015, An efficient ant colony system for edge detection in image processing, in: Proceedings of the European Conference on Artificial Life, pp. 398–405.
[29] M. Nayak, P. Dash, Edge detection improvement by ant colony optimization compared to traditional methods on brain MRI image, Commun. Appl. Electron. 5 (8) (2016) 19–23.
[30] C. Pereira, L. Gonçalves, M. Ferreira, Exudate segmentation in fundus images using an ant colony optimization approach, Inf. Sci. (Ny). 296 (2015) 14–24.
[31] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, W. Shi, Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, arXiv preprint, 2016.
[32] S. Goel, A. Verma, K. Juneja, 2015. A framework for improving misclassification rate of texture segmentation using ICA and ant tree clustering algorithm, in: Computing, Communication & Automation (ICCCA), 2015 International Conference on IEEE, pp. 22–27.
[33] Z. Zhang, Q. Wu, Z. Zhuo, X. Wang, L. Huang, Wavelet transform and texture recognition based on spiking neural network for visual images, Neurocomputing 151 (2015) 985–995.
[34] D. Avci, M. Poyraz, M.K. Leblebicioğlu, An expert system based on discrete wavelet transform-ANFIS for acquisition and recognition of invariant features from texture images, *Signal Processing and Communications Applications Conference (SIU), 2015 23th*, IEEE, 2015, pp. 1070–1073.
[35] E. Berkeley,n.d., Berkeley Segmentation Dataset Images. Retrieved February 23, 2018, from https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/.
[36] V. Caltech, n.d. Caltech ImageDatasets. Retrieved February 23, 2018, from http://www.vision.caltech.edu/Image_Datasets/Caltech256/.
[37] M. Xu, M. Cong, T. Xie, Y. Tao, X. Zhu, J. Zhao, Unsupervised segmentation of high-resolution remote sensing images based on classical models of the visual receptive field, Geocarto. Int. 30 (9) (2015) 997–1015.

[38] M. Mahi, Ö.K. Baykan, H. Kodaz, A new hybrid method based on particle swarm optimization, ant colony optimization and 3-opt algorithms for traveling salesman problem, Appl. Soft Comput. 30 (2015) 484–490.

[39] Y. Chen, A. An, Application of ant colony algorithm to geochemical anomaly detection, J. Geochem. Explor. 164 (2016) 75–85.

[40] R. Grycuk, M. Gabryel, R. Scherer, S. Voloshynovskiy, Multi-layer architecture for storing visual data based on WCF and microsoft SQL server database, International Conference on Artificial Intelligence and Soft Computing, Springer, Cham, 2015, June, pp. 715–726.

[41] A. Biniaz, A. Ataollah, Segmentation and edge detection based on modified ant colony optimization for Iris image processing, JAISCR 3 (2) (2013) 133–141. Available from: https://doi.org/10.2478/jaiscr-2014-0010.

FURTHER READING

S. Sarkar, S. Das, S.S. Chaudhuri, A multilevel color image thresholding scheme based on minimum cross entropy and differential evolution, Pattern Recogn. Lett. 54 (2015) 27–35.

This page intentionally left blank