Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 3514258, 25 pages
https://doi.org/10.1155/2019/3514258

Research Article

An Improved Multithreshold Segmentation Algorithm Based on Graph Cuts Applicable for Irregular Image

Yanzhu Hu, Jiao Wang, Xinbo Ai, and Xu Zhuang

School of Automation, Beijing University of Posts and Telecommunications, Beijing 100876, China

Correspondence should be addressed to Jiao Wang; wangjiao0516@bupt.edu.cn

Received 23 January 2019; Revised 21 March 2019; Accepted 7 April 2019; Published 13 May 2019

Academic Editor: Thomas Hanne

In order to realize the multithreshold segmentation of images, an improved segmentation algorithm based on graph cut theory using artificial bee colony is proposed. A new weight function based on gray level and the location of pixels is constructed in this paper to calculate the probability that each pixel belongs to the same region. On this basis, a new cost function is reconstructed that can use both square and nonsquare images. Then the optimal threshold of the image is obtained through searching for the minimum value of the cost function using artificial bee colony algorithm. In this paper, public dataset for segmentation and widely used images were measured separately. Experimental results show that the algorithm proposed in this paper can achieve larger Information Entropy (IE), higher Peak Signal to Noise Ratio (PSNR), higher Structural Similarity Index (SSIM), smaller Root Mean Squared Error (RMSE), and shorter time than other image segmentation algorithms.

1. Introduction

Image threshold segmentation refers to dividing an image into two parts: background and foreground under a certain gray value, and target object can be easily recognized by distinguishing between foreground and background [1, 2]. At present, target recognition based on image segmentation is widely used in medical care, military, geology, agriculture, and many other fields [3–6]. Due to the small difference between the target and the back ground of a complex image, the results of image threshold segmentation are often far from satisfactory. Considering the results of image segmentation are quite different under different thresholds, [7–9], providing an accurate, reliable, and effective method for identifying objects in complex background has a wide range of practical applications [10, 11]. On the other hand, with the development of computer science and technology, the real-time requirement of image segmentation is improved and finding the exact threshold quickly is also an important part [12, 13]. To sum up, it is important and necessary to find a suitable threshold quickly to complete the segmentation of the target object in an image.

In the last few decades, a number of image segmentation techniques have been devised, and image threshold segmentation is mainly divided into two categories: global threshold segmentation and multithreshold segmentation [14, 15]. Both global and multithreshold segmentation select thresholds by optimizing (maximizing or minimizing) some specific parameters [16]. Actually, constructing the appropriate parameters for image multithreshold segmentation is at the heart of solving this problem [17]. And there are two main methods for constructing parameters: entropy-based algorithm and graph cut algorithm [18]. Entropy-based Threshold Method was first proposed in 1980 by Pun [19] and is to develop extremely rapid in the next few decades. Chowdhury et al. used Shannon's entropy and proposed a new multithreshold image segmentation method based on minimization of bientropy function [20]. Hinojosa et al. used three of the most representative entropies—Kapur, minimum cross entropy, and Tsallis as objective functions [21]. Mishra et al. calculated the optimal threshold values using bat algorithm and maximizing different objective function values based on Kapur's entropy [22]. Pare et al. proposed an efficient multithreshold technique based on rendering minimum cross entropy [23]. Application of the various multithreshold approaches discussed above, which are all based on entropy such as bientropy, Kapur's entropy, and cross entropy, becomes computationally costly when

extended to perform multithreshold due to the constructed parameters functions to obtain optimum threshold values. Because of the measurement accuracy in selecting threshold and the simplicity in dealing with the problem of image multithreshold segmentation, the algorithm based on graph cut theory is being developed using a probability theory—that is, different pixels within the same set. The optimal thresholds are gained by minimizing the cut between different pixel sets [24]. Lu et al. proposed an effective approach for particle segmentation based on combing the background difference method and the graph cut based local threshold method [25]. Jimenez et al. presented a specifically designed graph cut methodology that ensures spatial and directional consistency [26]. Zhu et al. developed an optimized parameter based on graph cut to segment liver cysts [27]. Deng et al. obtained the segmentation by optimizing the cost function using graph cuts [28]. Gandhimathi et al. proposed an innovative spatial-spectral method for image segmentation based on graph cut [29]. Guo et al. presented an efficient image segmentation algorithm using neutrosophic graph cut (NGC) [30]. Although the method based on graph cut discussed above can achieve effective image segmentation, the weight function of graph cut, which is used to calculate the possibility that two pixels belong to one class, does not change, and it is inevitable that the segmentation effect will be unsatisfactory due to a slow gradient drop. Additionally, because of the complexity of the weight construction function and the cost function suitable for those algorithms, the computation time of the graph cut based method increases exponentially as the segmentation level increases.

Obviously, because traditional methods based on mathematical models are difficult to achieve ideal results, some new methods are incorporating bionic algorithms, such as Dragonfly algorithm [31], Artificial Bee Colony algorithm [32–34], Bat Algorithm [35], and Grey Wolf Algorithm [36]. Moreover, Gao et al. demonstrated the superiority of the ABC algorithm in finding an optimal value [34]. Inspired by the above algorithms, we propose an image multithreshold segmentation method based on graph cuts with artificial bee colony algorithm. Through constructing a new weight function, which is based on the location and gray value of pixels, the relationship between pixels is obtained. On this basis a new cost function is reconstructed, which can use both regular and irregular images. Then the artificial bee colony algorithm is used to search the optimal multithreshold segmentation values of an image. By comparing the information entropy (IE), Peak Signal to Noise Ratio (PSNR), and Structural Similarity Index (SSIM) of images as well as the time complexity of algorithm with the existing image segmentation methods, the proposed algorithm based on graph cuts in this paper can achieve better segmentation results with the shortest time.

The main work of this paper is given as follows. Section 2 introduces the advantages of the artificial bee colony algorithm briefly and the general process of the algorithm. Section 3 introduces the method to construct the new undirected graph. Based on this new undirected graph, the cost function of multithreshold segmentation is constructed. Section 4 demonstrates the effectiveness of the method through experiments. At the same time, qualitative and quantitative methods are used to evaluate this method. Section 5 concludes this paper.

2. Artificial Bee Colony Algorithm

Artificial bee colony algorithm is a global optimization algorithm by simulating bee foraging behavior. Since Karaboga and Basturk first proposed this algorithm in 2008, the artificial bee colony algorithm (ABC) has developed rapidly [32–34, 37]. The artificial bee colony algorithm contains three kinds of bees: employed bees, onlookers, and scouts. Employed bees bring nectar source back to the hive and share the information in the dancing area. By observing the information brought back by employed bees and calculating the number of food sources, the onlookers can determine the probability of selecting different nectar sources and make a decision on selection. Scouts make random searches near the sources. If the food source is not selected, the employed bee which carries the information of the food source becomes a scout bee and immediately searches near the original food source. As soon as a new food source is found, the scout bee becomes an employed bee again. In summary, every search cycle of artificial bee colony algorithm includes three steps: (1) employed bees are sent to find food sources while calculating the amounts of nectar; (2) employed bees share the information and onlooker bees select the optimal source by calculating the amount of different nectar sources; (3) the scout bees are then chosen and sent out to find the new food sources.

In ABC algorithm, the location of food source represents a possible optimal solution, and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution, calculated by

$$\mathrm{fit}_i = \frac{1}{1 + f_i} \tag{1}$$

where fit_i represents the quality (fitness) of the solution which is inversely proportional to f_i and $f_i$is the cost function which needs to be built for each specific problem. In this paper f_i is the cost function construct based on the undirected weight map in Section 3.1.

In the algorithm, the number of employed and onlooker is equal to the number of optimal solutions. Initially, the artificial bee colony algorithm randomly generates P of SN as the initial result, where SN denotes the size of population and each solution $z_i$is a vector of D dimensions of which elements are represented as z_{ij} ($j \in \{1, 2, ...D\}$). Here, D is the number of product of input size and cluster size for each data set. After initialization, the population of the positions (solutions) is subjected to repeated cycles, C = 1, 2, …, MCN, of the search processes of the employed bees, the onlooker bees, and scout bees. An employed bee produces a modification on the position (solution) in her memory depending on the local information (visual information) and tests the nectar amount (fitness value) of the new source (new solution). If more nectar is found at the new food source than from the previous source, the employed bees will remember the

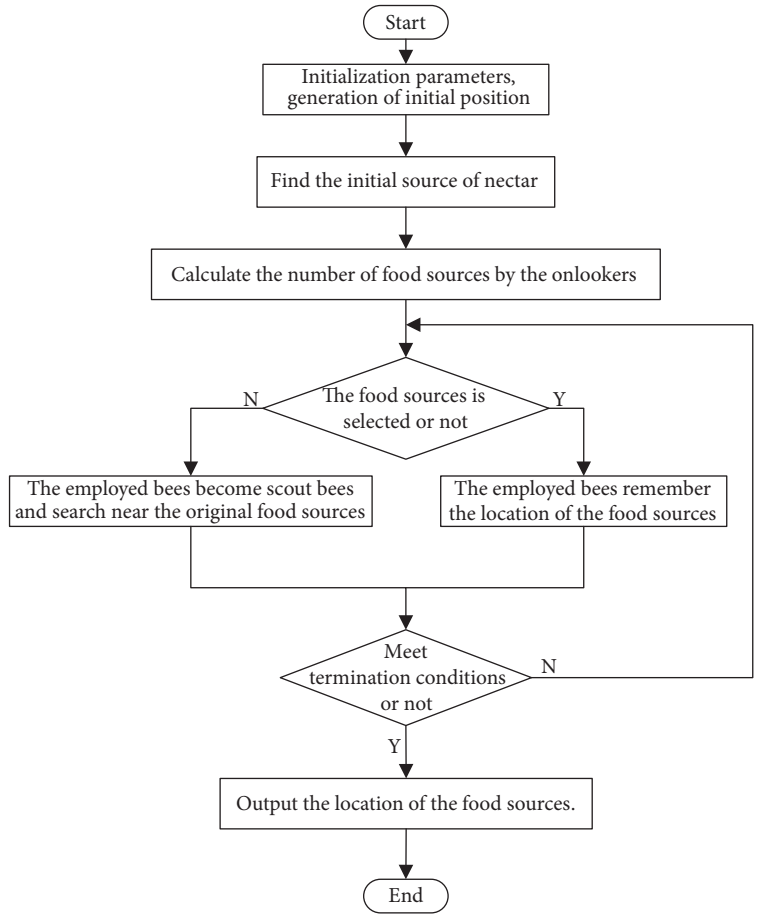

Figure 1: Flowchart of artificial bee colony algorithm.

location of the new source; otherwise, they will choose to remember the location of the original source. As soon as all employed bees have completed the search, they share the nectar source information and location with the onlooker bees, which will select the most possible food source as the optimal solution through calculating the number of nectar. The probability of choosing a nectar source is calculated by

$$p_i = \frac{\text{fit}_i}{\sum_{i=1}^{SN} \text{fit}_i} \tag{2}$$

where p_i is the probability of choosing nectar source, SN is the number of food sources which is equal to the number of employed bees, and fit_i is the fitness of the solution given in (1).

In order to produce a candidate food position from the old one in memory, the new source is obtained by

$$v_{ij} = z_{ij} + \varphi_{ij}\left(z_{ij} - z_{kj}\right) \tag{3}$$

where v_{ij} represents the candidate food position, z_{ij} is the original resource location, and z_{kj} is a generated resource location through choosing the indexes k ($k \in \{1, 2, ...SN\}$) and j ($j \in \{1, 2, ...D\}$) randomly. Although k is determined randomly, it has to be different from i and ϕ_{ij} is randomly generated in $[-1, 1]$.

If the location of the nectar source cannot be updated by the previous 'limit' of bees, the location of the nectar source z_i is discarded and the employed bees are turned into scout bees. It is assumed that the location of the abandoned nectar source is z_i and $j \in \{1, 2, ...D\}$, and then the scouts found a new food source of food to replace z_i. The above steps can be expressed by

$$z_{ij} = z_{\min}^{j} + \text{rand}\,(0, 1)\left(z_{\max}^{j} - z_{\min}^{j}\right) \tag{4}$$

where $z_{\max}^{j}$ and $z_{\min}^{j}$ are the upper and lower limits of the jth component of all solutions. If the new solution is better than the original, the scout bee will become an employed bee again. All employed bees, onlooker bees, and scout bees repeat the above steps until the termination criteria are met. The flow of the artificial bee colony algorithm is shown in Figure 1 and the search result of ABC algorithm is shown in Figure 2. At last, the fake code is given in Algorithm 1.

3. Threshold Segmentation Model Construction

According to the analysis above, the artificial bee colony algorithm can obtain the optimal solution of a certain problem by searching the location of the optimal nectar

Input: MCN Number of iterations for optimization,
SN Number of food sources equal to the number of employed bees.
(1) Initialize parameters and generate initial position.
(2) Find the initial source of nectar
(3) **While** Stopping criteria not met **do**
(4) calculate the number of food sources by the onlookers
(5) **if** the food sources is selected
(6) the employed bees remember the location of the food sources
(7) **else**
(8) the employed bees become scout bees and search near the original food sources
(9) **end if**
(10) **end while**
Output: The location of the food sources.

Algorithm 1: Algorithm ABC.

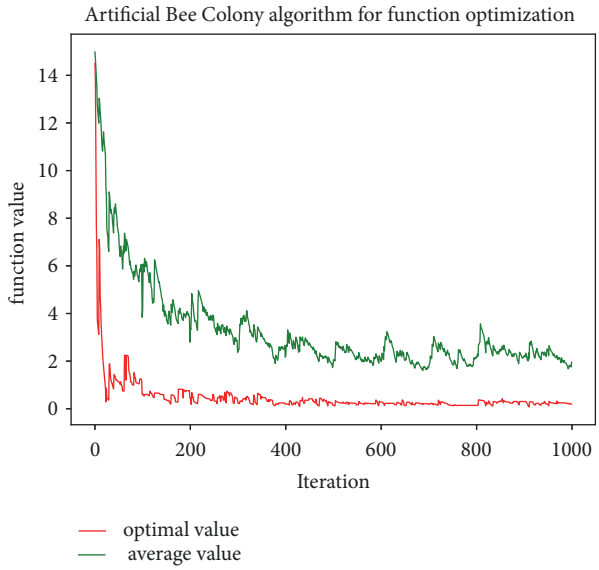

Figure 2: Search for the optimal value using the ABC algorithm.

source. In this section, we first converted an image undirected weight map based on graph spectra theory. On this basis, we constructed a cost function suitable for multithreshold segmentation. Finally, the artificial bee colony algorithm is used to search the minimum cut of undirected weight map to achieve threshold segmentation of the image.

3.1. Construction of Undirected Weight Map Based on Gray Value. According to undirected graph theory, the point sets of any feature space can be represented by $G = (V, E)$, where V represents the set of points and E represents the set of connecting edges between points. In undirected weighted map, there is only one connecting edge between two points. Weight $w\ (u,\ v)$ is given to the edge, which indicates the similarity between points u and v. In summary, the smaller the value, the less likely points u and v belong to the same set.

When constructing an undirected weighted map of an image, considering that the larger the distance between pixels, the less likely they belong to the same set, the weight function constructed is required to have a fast descent gradient, which means when the denominator of the weight function increases, the weight value decreases rapidly, implying the possibility that two pixel points belong to the same set quickly decreases. At the same time, the weight value w represents the probability, which is nonnegative. To sum up, the edge weight between pixel point u and pixel point v is as follows:

$$w(u,v) = \begin{cases} \dfrac{1}{d_t \left\| F(u) - F(v) \right\|_2^2 + d_X \left\| X(u) - X(v) \right\|_2^2} & \left\| X(u) - X(v) \right\|_2 < r \\ 0 & \text{others} \end{cases} \tag{5}$$

where $F(\bullet)$ is the gray value of pixel points, $X(\bullet)$ is the spatial position of pixel points, $\| \bullet \|_2^2$ is the two norm, d_i and d_X are positive scale factors, and r is positive integer, representing the range of pixel points involved in calculating the weight. The weight function is visualized for several values of $r \in [1, 5]$ in Figure 3.

In this paper, $r = 2, d_i = 1/25$, and $d_X = 1/4$ is taken as an example to test the effectiveness of the algorithm. Meanwhile, the weight function constructed in this paper is lower than the original function in time complexity, and its analysis is put in Section 4.

3.2. The Cost Function Construct Based on the Undirected Weight Map. For any threshold $t = \{t_1, t_2, \ldots t_n\}, 0 < t_1 < t_2, \ldots t_n < T$, where T is dynamic which depends on the bits per pixel occupied (T=255 if 8 bits per pixel while T=65535 if 16 bits per pixel). We can get a multithreshold partition $V = \{H_1, H_2, \ldots H_n\}$ of the corresponding undirected weighted

map $G = (V, E)$ of the image, which can be expressed as

$$\begin{aligned} H_1 &= \bigcup_{k=0}^{t_1-1} V_k \\ H_2 &= \bigcup_{k=t_1}^{t_2} V_k \\ &\bullet\bullet\bullet \\ H_n &= \bigcup_{k=t_n}^{255} V_k \\ k &\in L \end{aligned} \tag{6}$$

where V represents the collection of pixels,E represents the collection of edges between pixels, and H_n represents a pixel collection belonging to class$n$. According to the graph cuts theory, when the image is segmented by multithresholds, the difference between pixels belonging to different divisions is the largest while the difference between pixels belonging to the same division is the smallest. The cut between H_1 and H_2 is defined as

$$\begin{aligned} \operatorname{cut}(H_1, H_2) &= \sum_{u\in H_1, v\in H_2} w(u,v) = \sum_{u\in H_1}\left[\sum_{v\in H_2} w(u,v)\right] \\ &= \sum_{i=0}^{t_1}\sum_{u\in V_i}\left[\sum_{j=t_1+1}^{t_2}\sum_{v\in V_j} w(u,v)\right] \\ &= \sum_{i=0}^{t_1}\sum_{j=t_1+1}^{t_2}\left[\sum_{u\in V_i, v\in V_j} w(u,v)\right] \end{aligned} \tag{7}$$

Similarly, the cut between $H_1$and H_1 is defined as

$$\begin{aligned} \operatorname{asso}(H_1, H_1) &= \sum_{u\in H_1, v\in H_1} w(u,v) \\ &= \sum_{i=0}^{t_1}\sum_{j=i}^{t_1}\left[\sum_{u\in V_i, v\in V_j} w(u,v)\right] \end{aligned} \tag{8}$$

In the same way

$$\begin{aligned} \operatorname{asso}(H_2, H_2) &= \sum_{u\in H_2, v\in H_2} w(u,v) \\ &= \sum_{i=t_1}^{t_2}\sum_{j=i}^{t_2}\left[\sum_{u\in V_i, v\in V_j} w(u,v)\right] \end{aligned} \tag{9}$$

For image multithreshold segmentation, it is to find $t = \{t_1, t_2, \ldots t_n\}, 0 < t_1 < t_2, \ldots t_n < T$ making the value of $\operatorname{cut}(H_1, H_2) + \operatorname{cut}(H_1, H_3) + \ldots + \operatorname{cut}(H_1, H_n) + \operatorname{cut}(H_2, H_3) + \ldots \operatorname{cut}(H_{n-1}, H_n)$ minimum while the value of $\operatorname{asso}(H_1, H_1) + \operatorname{asso}(H_2, H_2) + \ldots + \operatorname{asso}(H_n, H_n)$ maximum.

In order to facilitate further analysis, we define

$$\operatorname{cut}(V_i, V_j) = \sum_{u\in V_i, v\in V_j} w(u,v) \tag{10}$$

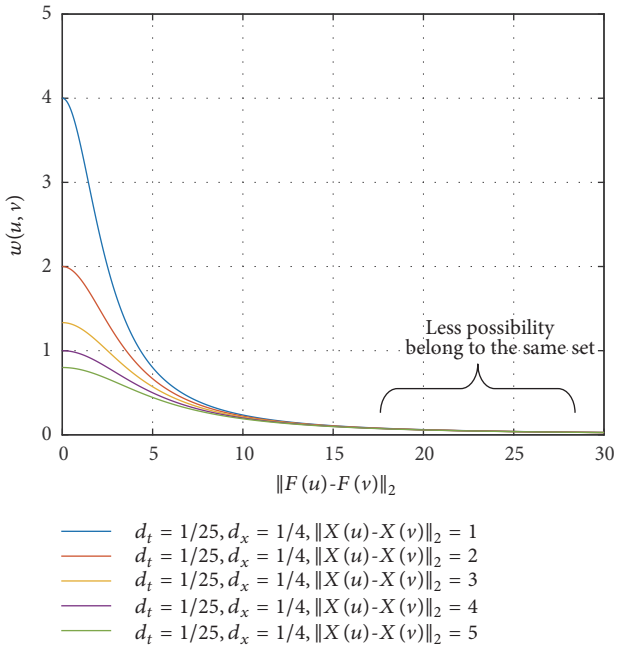

Figure 3: Graphical display of the weight function results constructed in this paper.

Then (7) can be transformed into

$$\operatorname{cut}(H_1, H_2) = \sum_{i=0}^{t_1}\sum_{j=t_1+1}^{t_2} \operatorname{cut}(V_i, V_j) \tag{11}$$

Similarly,

$$\operatorname{cut}(H_1, H_3) = \sum_{i=0}^{t_1}\sum_{j=t_2}^{t_3} \operatorname{cut}(V_i, V_j) \tag{12}$$

Therefor (13) is established:

$$\begin{aligned} &\operatorname{cut}(H_1, H_2) + \operatorname{cut}(H_1, H_3) \\ &= \sum_{i=0}^{t_1}\sum_{j=t_1+1}^{t_2} \operatorname{cut}(V_i, V_j) + \sum_{i=0}^{t_1}\sum_{j=t_2+1}^{t_3} \operatorname{cut}(V_i, V_j) \\ &= \sum_{i=0}^{t_1}\sum_{j=t_1+1}^{t_3} \operatorname{cut}(V_i, V_j) = \operatorname{cut}(H_1, H_2 + H_3) \end{aligned} \tag{13}$$

By parity of reasoning, (14) also holds.

$$\begin{aligned} &\operatorname{cut}(H_1, H_2) + \operatorname{cut}(H_1, H_3) + \cdots + \operatorname{cut}(H_1, H_n) \\ &\quad + \operatorname{cut}(H_2, H_3) + \cdots \operatorname{cut}(H_{n-1}, H_n) \\ &= \operatorname{cut}(H_1, H_2) + \operatorname{cut}(H_1 + H_2, H_3) + \cdots \\ &\quad + \operatorname{cut}(H_1 + H_2 + \cdots + H_{n-1}, H_n) \end{aligned} \tag{14}$$

Similarly, in order to overcome the problem of isolated points in segmentation, Normalized Cuts (Ncut) is adopted

Input: An image with gray value,
Segmented threshold level.
(1) Calculate the edge weight between pixels with the new constructed function
(2) Generate segmentation threshold randomly
(3) Calculate the value of Ncut under the threshold
(4) Calculate the value of new cost function (fit_i) based on Ncut
(5) **While** Stopping criteria not met **do**
(6) Search new threshold near the original threshold using artificial bee colony algorithm
(7) Recalculate the value of Ncut under the new threshold
(8) Recalculate he value of new cost function (fit_i) based on Ncut
(9) **if** the cost function becomes smaller
(10) Continue searching new threshold near the original threshold
(11) **else**
(12) **Break**
(13) **end if**
(14) **end while**
Output: Multi-level image segmentation threshold

Algorithm 2: Multithreshold image segmentation algorithm based on graph cuts.

to describe the degree of separation between the two classes [38], which is defined as follows:

$$\mathrm{Ncut}(A,B) = \frac{\mathrm{cut}(A,B)}{\mathrm{asso}(A,V)} + \frac{\mathrm{cut}(A,B)}{\mathrm{asso}(B,V)} \tag{15}$$

Equation (16) can be obtained by transforming (15).

$$\begin{aligned}\mathrm{Ncut}(H_1,H_2) &= \frac{\mathrm{cut}(H_1,H_2)}{\mathrm{asso}(H_1,H_1+H_2)} \\ &\quad + \frac{\mathrm{cut}(H_1,H_2)}{\mathrm{asso}(H_2,H_1+H_2)} \\ &= \frac{\mathrm{cut}(H_1,H_2)}{\mathrm{asso}(H_1,H_1)+\mathrm{cut}(H_1,H_2)} \\ &\quad + \frac{\mathrm{cut}(H_1,H_2)}{\mathrm{asso}(H_2,H_2)+\mathrm{cut}(H_1,H_2)}\end{aligned} \tag{16}$$

After normalization, (17) can be obtained from (14).

$$\begin{aligned}&\mathrm{Ncut}(H_1,H_2) + \mathrm{Ncut}(H_1+H_2,H_3) + \cdots \\ &\quad + \mathrm{Ncut}(H_1+H_2+\cdots+H_{n-1},H_n) \\ &= \frac{\mathrm{cut}(H_1,H_2)}{\mathrm{asso}(H_1,H_1+H_2)} + \frac{\mathrm{cut}(H_1,H_2)}{\mathrm{asso}(H_2,H_1+H_2)} \\ &\quad + \frac{\mathrm{cut}(H_1+H_2,H_3)}{\mathrm{asso}(H_1+H_2,H_1+H_2+H_3)} \\ &\quad + \frac{\mathrm{cut}(H_1+H_2,H_3)}{\mathrm{asso}(H_3,H_1+H_2+H_3)} + \cdots \\ &\quad + \frac{\mathrm{cut}(H_1+H_2+\cdots+H_{n-1},H_n)}{\mathrm{asso}(H_1+H_2+\cdots+H_{n-1},H_1+H_2+\cdots+H_n)} \\ &\quad + \frac{\mathrm{cut}(H_1+H_2+\cdots+H_{n-1},H_n)}{\mathrm{asso}(H_n,H_1+H_2+\cdots+H_n)}\end{aligned} \tag{17}$$

Therefore, the key of image segmentation is to find a set of N thresholds, so that (18) can achieve the smallest value, which is mentioned in (1).

$$\begin{aligned}&f([t_1,t_2,\ldots t_n]) \\ &\quad = \mathrm{Ncut}(H_1,H_2) + \mathrm{Ncut}(H_1+H_2,H_3) + \cdots \\ &\quad + \mathrm{Ncut}(H_1+H_2+\cdots+H_{n-1},H_n)\end{aligned} \tag{18}$$

The flowchart of image segmentation algorithm proposed in this paper is shown in Figure 4 and the fake code is given in Algorithm 2.

4. Experiments

In this section, we will evaluate the performance of the algorithm proposed in this paper comprehensively. Firstly, public dataset for segmentation and widely used images are evaluated separately using the proposed algorithm in this paper to segment each image into two, three, four, and five levels of threshold. And at the same time, quantitative methods are used to demonstrate the advantages of the proposed algorithm through comparing the Information Entropy (IE), Root Mean Squared Error (RMSE), Peak Signal to Noise Ratio (PSNR), and Structural Similarity Index (SSIM) of images with other widely used algorithms such as BA (Bat Algorithm) [35], IBA (Improved Bat Algorithm) [39], MMSA (Meta-heuristic Moth Swarm Algorithm) [40], and OTUS [41] algorithms. Finally, the time advantage of the algorithm is confirmed via analyzing the time complexity of the algorithm.

4.1. Qualitative Comparison and Analysis of Different Algorithms. In this part, we will compare the performance of our algorithm using the new weight and cost function with other algorithms. Firstly, we selected five images commonly used in the image field to verify the effectiveness of the algorithm proposed in this paper. Figures 5, 6, 7, and 8 are

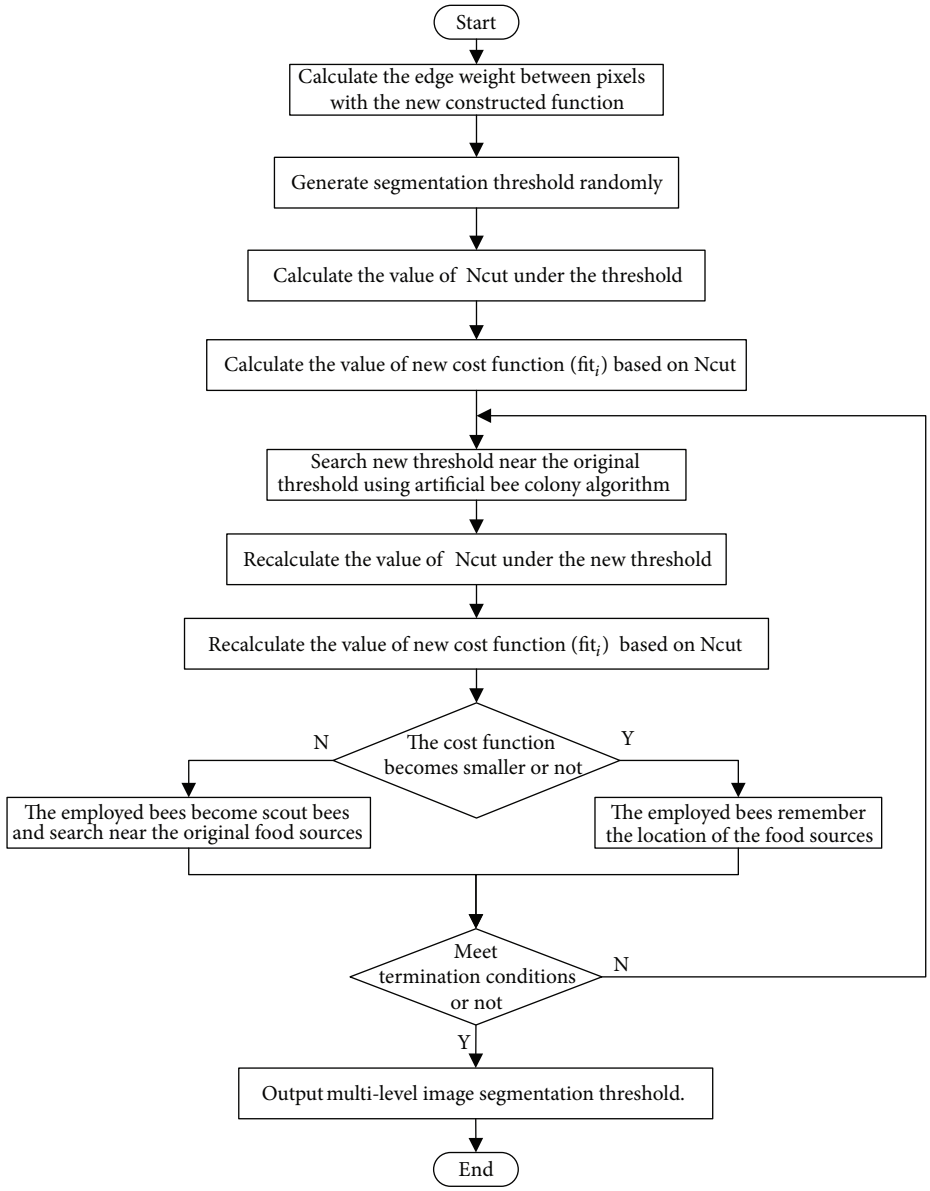

Figure 4: Flowchart of image segmentation algorithm proposed in this paper.

the segmentation results obtained by different algorithms. Figure 5 shows the two-level segmentation results, Figure 6 shows the three-level segmentation results, Figure 7 shows the four-level segmentation results, and Figure 8 shows the five-level segmentation results.

The specific segmentation thresholds of different algorithms are given in Table 1. As can be seen from Table 1, the segmentation results given by our method are slightly different from those of other methods.

Further, we selected ten images from the data set [42] to justify the superiority of proposed method. Figures 9, 10, 11, and 12 are the segmentation results obtained by different algorithms. Figure 9 shows the two-level segmentation results, Figure 10 shows the three-level segmentation results, Figure 11 shows the four-level segmentation results, and Figure 12 shows the five-level segmentation results.

The specific segmentation thresholds of different algorithms are given in Table 2, and qualitative analysis of the advantages and disadvantages of those algorithms is put in Section 4.2.

4.2. Quantitative Comparison and Analysis of Different Algorithms. In this part, we will evaluate the performance of the algorithm quantitatively by calculating Information Entropy (IE), Root Mean Squared Error (RMSE), Peak Signal to Noise Ratio (PSNR), and Structural Similarity Index (SSIM) of images. Generally speaking, the entropy of an image represents the information contained in the image. According to the theory of Information Entropy (IE), the better segmentation results are, the greater the value of information entropy is. The entropy of an image can be calculated as follows:

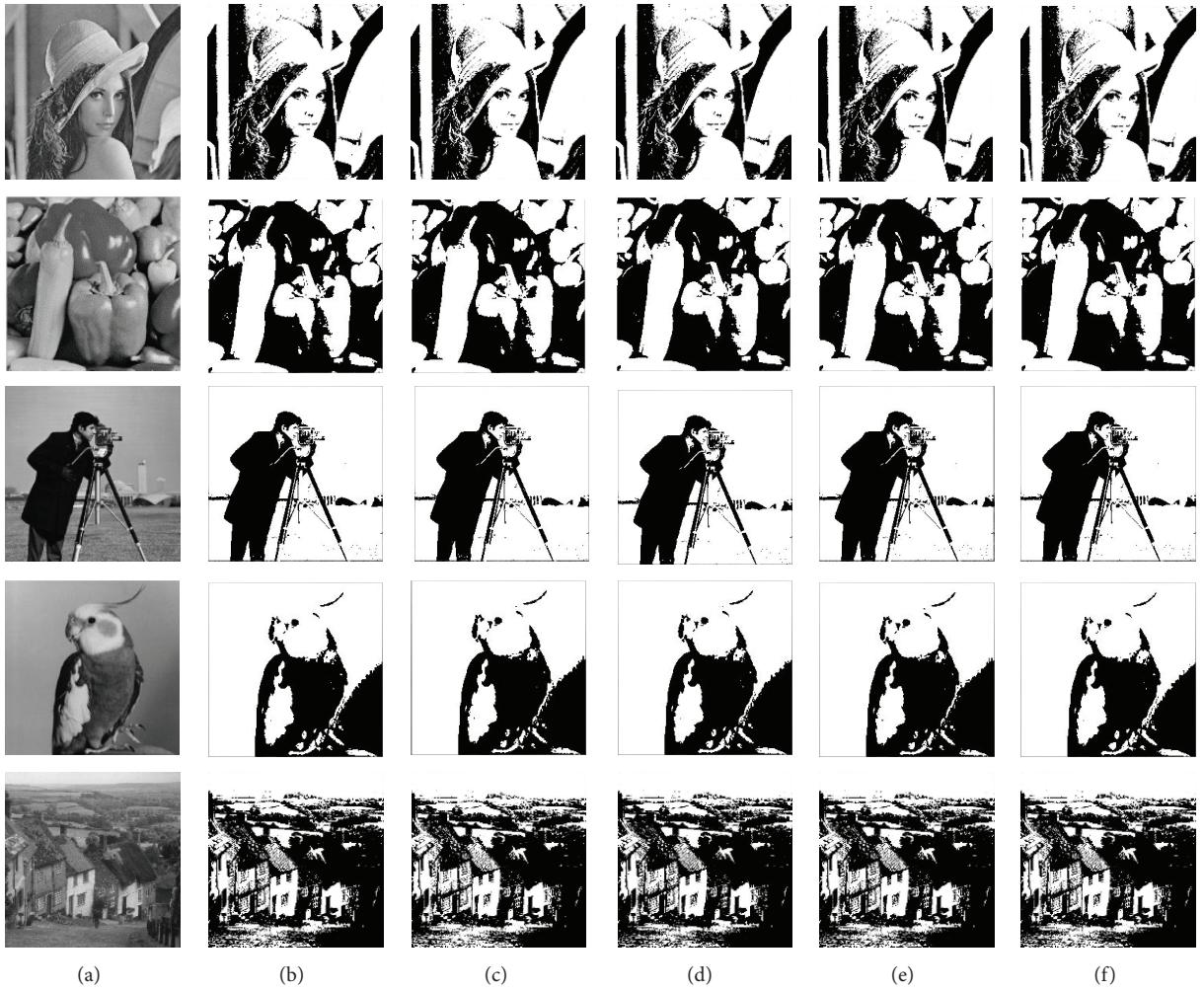

Figure 5: Two-level segmentation results of different algorithms: (a) origin image, (b) segmented image using our method, (c) segmented image using BA algorithms, (d) segmented image using MMSA algorithms, (e) segmented image using IBA algorithms, and (f) segmented image using OTSU algorithms.

$$\mathrm{H} = -\sum_{i=o}^{M} p(k) \log_2 p(k) \tag{19}$$

where $p(k)$ is the probability density of pixel value k and M is largest pixel value. For the convenience of reading, we put the results of the two data sets in one table. The Information Entropy (IE) of different segmented images using various algorithms is given in Table 3.

As shown in Table 3, the image segmented by the algorithm proposed in this paper can obtain a larger information entropy (IE), which means the algorithm proposed in this paper has the best segmentation effect compared with other algorithms mentioned above. What is more the value of information entropy (IE) of multithreshold segmented image is greater than that of two-level threshold segmentation, meaning the more threading levels there are, the less information lost is.

Root Mean Squared Error (RMSE) is a mathematical model established based on the visual system of human eyes, which determines the degree of distortion of the image by calculating the mean square value of the pixel difference between the original image and the processed image. The entropy of an image can be calculated as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{M \times N} \sum_{0 \le i < N} \sum_{0 \le j < M} \left(f_{ij} - f_{ij}{}' \right)^2} \tag{20}$$

where M and N represents the length and width of the image, f_{ij} represents the gray value of the point (i, j) in the original image, and $f_{ij}{}'$ represents the pixel value of the point (i, j) in the image after segmentation. We put the results of the two datasets in one table. The Root Mean Squared Error (RMSE) of different segmented images using various algorithms is given in Table 4.

As shown in Table 4, the image segmented by the algorithm proposed in this paper can obtain smaller Root Mean Squared Error (RMSE), which means the proposed algorithm has the least degree of distortion compared with other algorithms. The value of The Root Mean Squared Error (RMSE) of the multithreshold segmentation image is greater

(a) (b) (c) (d) (e) (f)

Figure 6: Three-level segmentation results of different algorithms: (a) origin image, (b) segmented image using our method, (c) segmented image using BA algorithms, (d) segmented image using MMSA algorithms, (e) segmented image using IBA algorithms, and (f) segmented image using OTSU algorithms.

Table 1: Specific segmentation thresholds of different algorithms.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
LENA (512×512)	2	121	118	120	121	119
	3	78,146	80,150	80,150	80,150	77,145
	4	58,108,160	60,109,160	60,109,160	60,109,160	56,106,159
	5	60,105,144,180	56,100,144,182	56,100,144,182	56,100,144,182	74,112,144,179
PEPPERS (512×512)	2	128	128	130	129	126
	3	70,140	74,146	74,146	74,146	67,134
	4	61,115,164	61,112,164	72,135,193	61,112,164	61,117,165
	5	55,104,150,194	57,104,148,194	58,105,148,194	57,104,148,194	46,85,125,168
BIRD (256×256)	2	105	108	106	109	104
	3	70,125	71,138	71,138	71,138	68,124
	4	67,128,177	70,129,177	70,129,177	70,130,177	65,116,159
	5	55,96,139,177	51,96,139,177	51,94,138,177	51,96,140,177	58,96,131,163
CAMERA (256×256)	2	76	75	76	77	76
	3	71,150	128,193	128,193	128,193	69,143
	4	58,120,156	44,104,193	44,104,193	44,104,193	58,118,155
	5	40,95,146,195	44,97,146,197	44,97,146,197	44,97,146,197	41,94,139,169
GOLDHILL (256×256)	2	115	113	116	115	114
	3	92,158	90,157	90,157	90,157	93,160
	4	80,126,178	79,132,178	79,132,178	79,132,178	82,125,178
	5	70,108,150,192	67,108,151,191	66,107,151,191	65,104,144,186	69,102,137,185

Figure 7: Four-level segmentation results of different algorithms: (a) origin image, (b) segmented image using our method, (c) segmented image using BA algorithms, (d) segmented image using MMSA algorithms, (e) segmented image using IBA algorithms, and (f) segmented image using OTSU algorithms.

than that of two-threshold segmentation, meaning that the more threading level there are, the less degree of distortion is.

Peak Signal to Noise Ratio (PSNR) is another important indicator to measure image quality. It is based on communication theory which represents the ratio of maximum semaphore to noise intensity. Since digital images represent image pixels in discrete numbers, the maximum pixel value of the image is used instead of the maximum semaphore. The specific formula is as follows:

$$\mathrm{PSNR} = 10 \times \lg \frac{L \times L}{\mathrm{MSE}} \tag{21}$$

where L is the maximum gray value of the pixels in the image, generally 255, and *MSE* is the square of *RMSE*. We also only used one table to present the results. The Peak Signal to Noise Ratio (PSNR) of different segmented images using various algorithms is given in Table 5.

As shown in Table 5, the image segmented by the algorithm proposed in this paper can obtain a higher Peak Signal to Noise Ratio (PSNR), which means the algorithm proposed in this paper has the best background noise filtering compared with other algorithms whether it is in two-level threshold segmentation or multithreshold segmentation

Structural Similarity Index (SSIM) is an indicator that measures the similarity of two images. The method was first proposed by the University of Texas at Austin's Laboratory for Image and Video Engineering. If the two images are one after segmentation and the other before segmentation, SSIM algorithm can be used to evaluate the segmentation effect. The calculation formula is as follows:

$$\begin{aligned}\mathrm{SSIM}\left(I_O, I_S\right) &= \mathrm{SSIM}\left(I_S, I_O\right) \\ &= \frac{\left(2\mu_{I_O}\mu_{I_S} + c_1\right)\left(2\sigma_{I_O}\sigma_{I_S} + c_2\right)}{\left(\mu_{I_O}^2 + \mu_{I_S}^2 + c_1\right)\left(\sigma_{I_O}^2 + \sigma_{I_S}^2 + c_2\right)}\end{aligned} \tag{22}$$

where I_O represents the original image and I_S represents the segmented image. μ_{I_O} and μ_{I_S}, respectively, represent the mean values of images I_O and I_S, σ_{I_O} and σ_{I_S} represent the standard deviations of images I_O and I_S, respectively, and $\mu_{I_O}^2$ and $\mu_{I_S}^2$ are the square of μ_{I_O} and μ_{I_S}.$\sigma_{I_O}^2$ and $\sigma_{I_S}^2$ represent the variance of the images I_O and I_S, and c_1 and c_2 are constants to maintain stability in order to avoid the

Figure 8: Five-level segmentation results of different algorithms: (a) origin image, (b) segmented image using our method, (c) segmented image using BA algorithms, (d) segmented image using MMSA algorithms, (e) segmented image using IBA algorithms, and (f) segmented image using OTSU algorithms.

denominator being zero. Normally, $c_1 = (K_1 * L)^2$ and $c_2 = (K_2 * L)^2$, where $K_1 = 0.01$ and $K_2 = 0.03$. L is the dynamic range of pixel values, generally taken as 255. We put the results of two datasets together, and the Structural Similarity Index (SSIM) of different segmented images using various algorithms is given in Table 6.

As shown in Table 6, the image segmented by the algorithm proposed in this paper can obtain a higher Structural Similarity Index (SSIM), which means the algorithm proposed in this paper has the highest similarity to the original image compared with other algorithms. The value of Structural Similarity Index (SSIM) of multithreshold segmented image is higher than that of two-threshold segmentation, meaning the more the threading levels there are, the higher the similarity is.

4.3. Time Complexity Analysis of Different Algorithm. In this part, we show the time advantage of the algorithm by analyzing the time complexity of the algorithm.

The computing of the algorithm proposed in this paper can be divided into two parts: the first part is the computational time T_1 needed to construct the undirected weight map based on gray level, and the second part is the time needed to search the optimal solution using artificial bee colony algorithm according to the undirected weight map. The analysis of the time complexity of the second part has been given in literature [29]; therefore it will not be involved in the essay. For the first part, the computation of structuring the undirected weight map depends on the parameter r. With the increase of r, there are more edges connecting the points in weight map **G**, and the corresponding calculation increases as well. Obviously, in (4), r=1 is meaningless while r=2 means, for every pixel, we must calculate the weight value between this pixel and every other pixel in its 3∗3 neighborhood. The total amount of calculation frequency needed to calculate all pixels in undirected weight map **G** is (8 ∗N)/2=4 ∗N, where N represents the total number of pixels. Division by '2' is because the weight between pixel point v and pixel point u is repeatedly calculated twice when pixel point v and pixel point u are, respectively, centered.

Generally speaking, when r>1, every pixel has $[2(r-1)+1]^2-1$ neighborhood pixels except the pixels on the boundary of an image. Therefore, the number of weights needed to calculate in the undirected weight map is

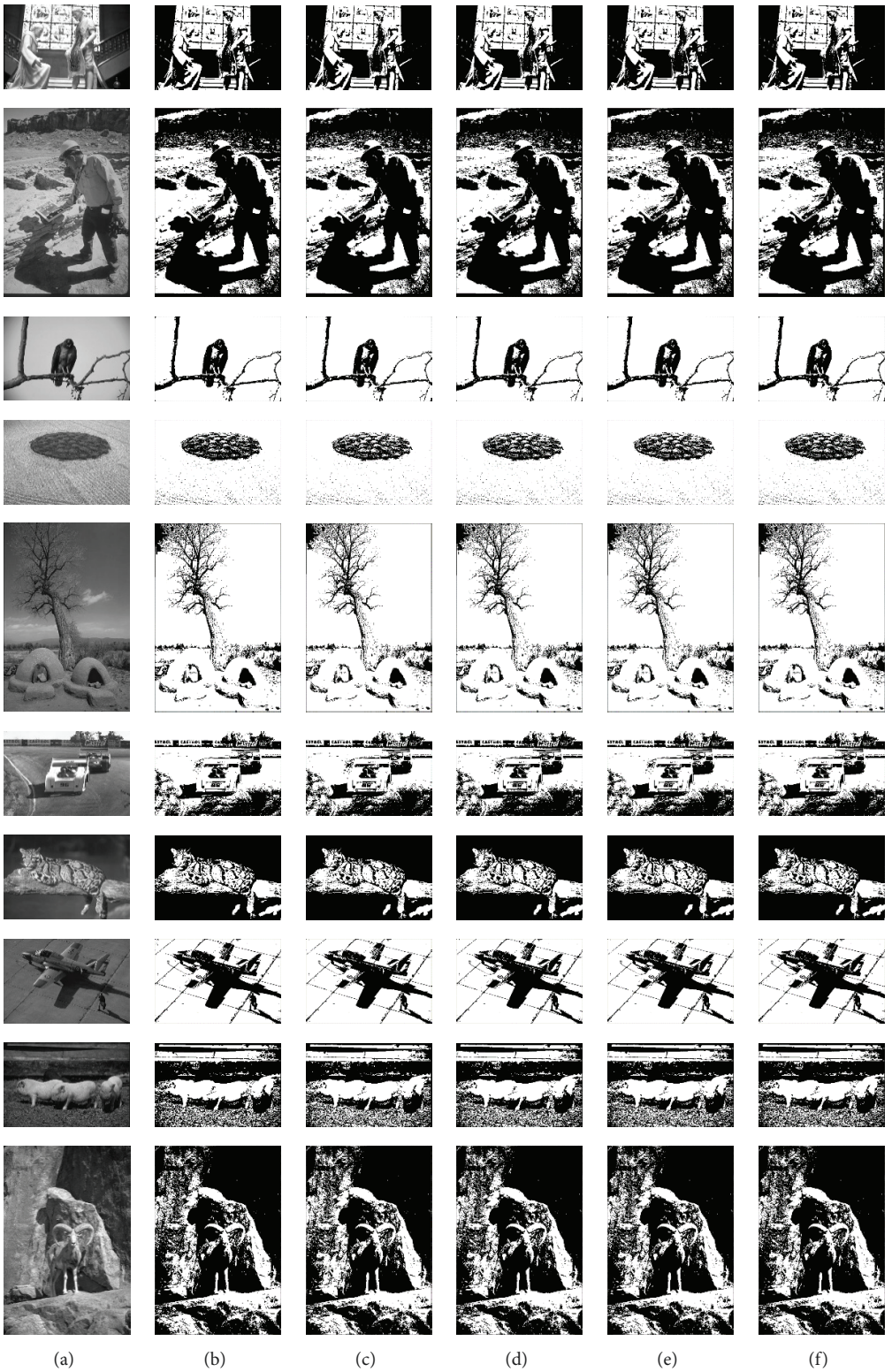

Figure 9: Two-level segmentation results of different algorithms: (a) origin image, (b) segmented image using our method, (c) segmented image using BA algorithms, (d) segmented image using MMSA algorithms, (e) segmented image using IBA algorithms, and (f) segmented image using OTSU algorithms.

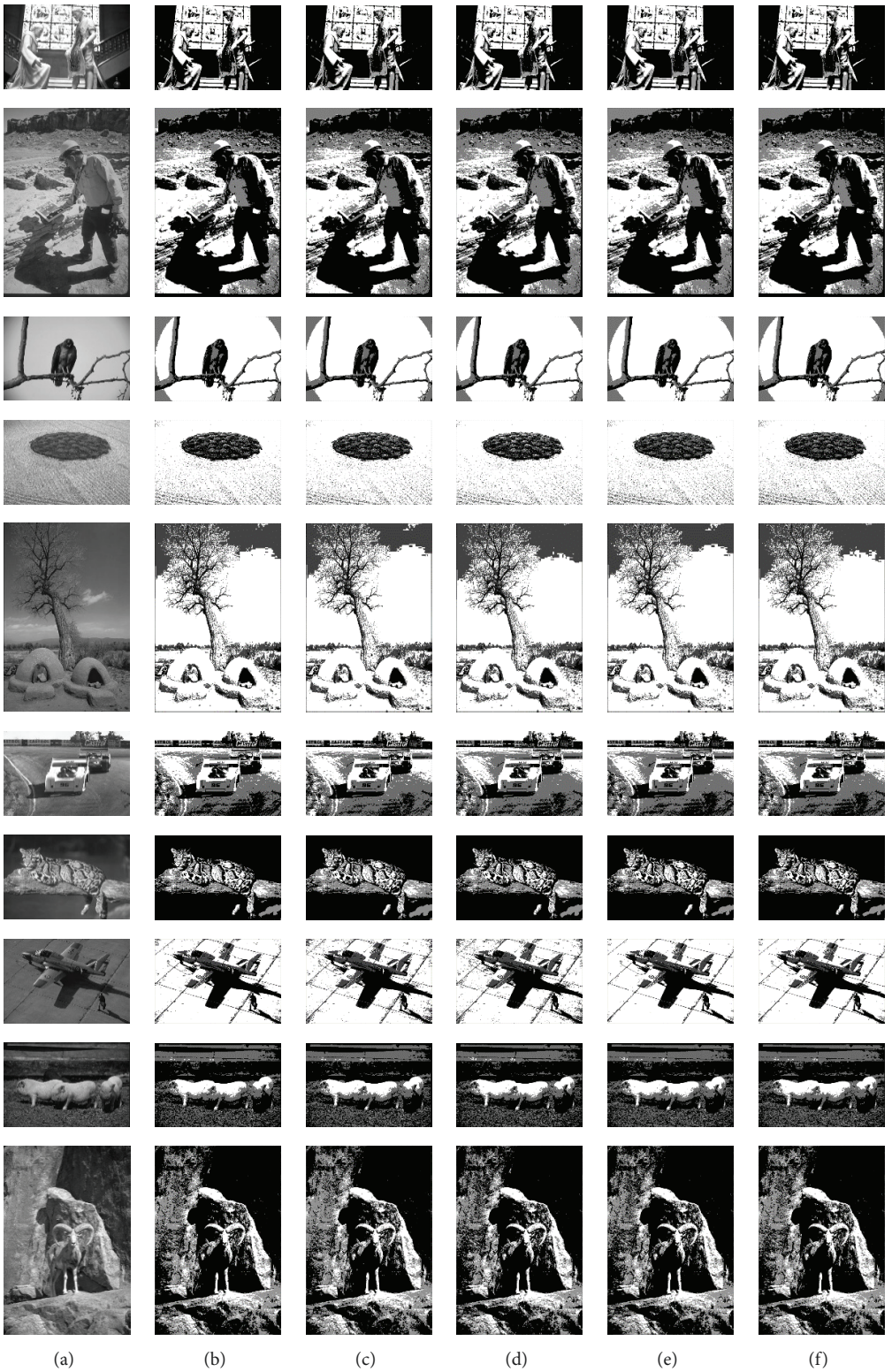

Figure 10: Three-level segmentation results of different algorithms: (a) origin image, (b) segmented image using our method, (c) segmented image using BA algorithms, (d) segmented image using MMSA algorithms, (e) segmented image using IBA algorithms, and (f) segmented image using OTSU algorithms.

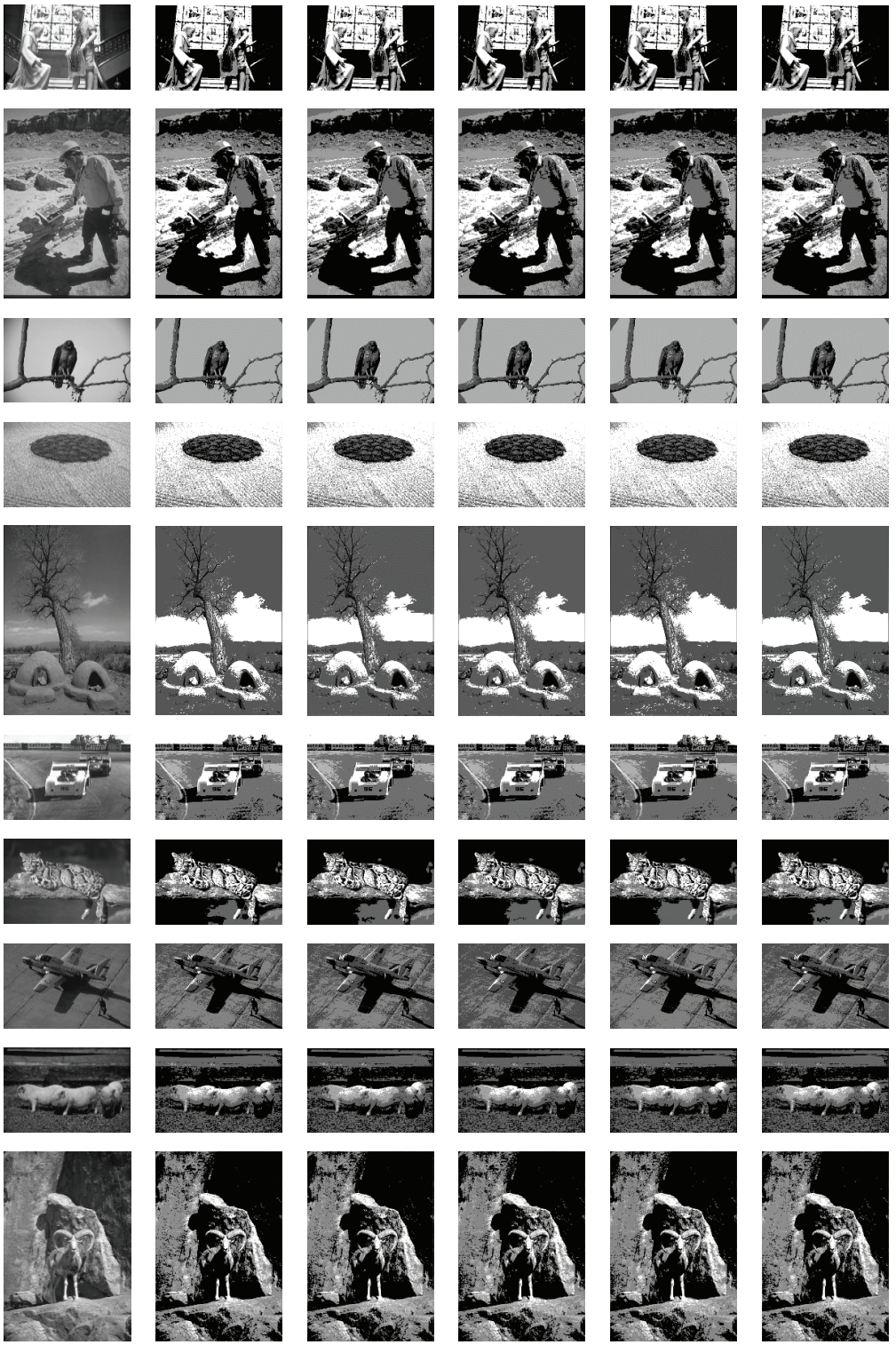

Figure 11: Four-level segmentation results of different algorithms: (a) origin image, (b) segmented image using our method, (c) segmented image using BA algorithms, (d) segmented image using MMSA algorithms, (e) segmented image using IBA algorithms, and (f) segmented image using OTSU algorithms.

Figure 12: Five-level segmentation results of different algorithms: (a) origin image, (b) segmented image using our method, (c) segmented image using BA algorithms, (d) segmented image using MMSA algorithms, (e) segmented image using IBA algorithms, and (f) segmented image using OTSU algorithms.

Table 2: Specific segmentation thresholds of different algorithms.

Image ID	level	Our algorithm	BA	MMSA	IBA	OTSU
24077	2	141	136	138	142	145
(481×321)	3	135,186	131,190	133,181	129,184	134,181
	4	121,167,201	117,160,196	119,162,203	123,171,204	118,164,205
	5	89,141,181,230	84,136,179,224	86,137,183,239	91,145,183,235	88,139,183,226
89072	2	134	131	132	134	132
(321×481)	3	104,154	101,152	99,157	103,159	105,156
	4	97,137,181	91,132,180	92,135,183	94,134,184	93,136,186
	5	84,143,175,236	81,137,171,232	79,136,172,231	82,135,177,239	80,140,172,234
42049	2	88	84	85	86	85
(481×321)	3	66,165	62,161	63,162	64,167	64,163
	4	41,124,221	42,128,218	45,130,215	39,129,224	42,126,225
	5	45,91,161,235	40,93,167,228	41,86,158,229	48,96,168,231	42,89,159,231
86016	2	95	91	92	91	93
(481×321)	3	87,121	83,119	84,116	82,123,	85,125
	4	82,124,152	81,126,151	84,123,147	79,119,143	81,126,151
	5	71,129,152,201	73,132,156,205	69,128,153,206	74,131,156,206	76,132,149,204
54082	2	73	71	72	72	73
(321×481)	3	48,87	45,86	46,89	45,86	47,88
	4	35,52,121	32,54,125	36,56,124	33,51,118	32,50,124
	5	50,93,130,192	48,89,126,188	46,91,135,196	47,96,126,190	48,91,128,195
21077	2	102	104	105	103	103
(481×321)	3	95,130	91,129	92,134	94,135	93,133
	4	84,142,216	81,143,219	82,145,214	80,139,215	82,145,219
	5	80,117,143,225	78,114,147,227	77,115,141,221	76,115,149,220	78,119,141,224
160068	2	119	116	117	117	118
(481×321)	3	117,169	112,170	115,172,	120,168,	119,171
	4	82,120,158	80,124,160	79,121,162	78,123,159	80,124,160
	5	90,121,150,200	91,125,153,204	92,123,154,203	87,118,151,197	88,120,154,201
37073	2	86	84,	85	84	86
(481×321)	3	54,91	51,94	52,95	50,88	51,89
	4	56,102,162,	54,103,165	52,106,159	53,99,159	57,104,165
	5	51,86,147,201	49,84,144,199	53,84,150,197	51,89,149,204	53,83,145,200
66053	2	75	72,	73	73,	74
(481×321)	3	71,148	70,146	73,145	69,145	68,146
	4	62,143,186	61,141,189	59,140,185,	60,147,188	60,142,189
	5	60,112,177,205	63,115,179,201	61,116,174,208	58,114,180,201	61,115,179,201
304074	2	136	134	135	135	136
(321×481)	3	127,173	124,170	130,171	129,173	127,173
	4	115,164,199	114,161,201	113,159,197	111,161,195	113,168,201
	5	104,153,190,229	105,149,187,225	106,151,188,226	103,151,190,227	104,156,187,228

$$t_1 = \frac{[2(r-1)+1]^2 - 1}{2} \times N = 2r(r-1)N \qquad (23)$$

The time complexity of t_1 is $O(r^2 N)$, and the time cost of various algorithm compared with our method is given in Table 7.

As shown in Table 7, the image segmented by the algorithm proposed in this paper can reduce the computation load, which means the algorithm proposed in this paper has the shortest computation time compared with other algorithms.

5. Conclusion

In this paper, we have proposed an improved segmentation algorithm based on graph cut theory using artificial bee colony. This approach uses a new weight function based on gray level and the location of pixels to calculate the probability that each pixel belongs to the same region. Then the optimal threshold of the image is obtained through searching for the minimum value of the cost function which is constructed based on the weight function using artificial bee colony algorithm. Experiment results show that

Table 3: The Information Entropy (IE) of different segmented images using various algorithm.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
LENA (512×512)	2	2.63	2.61	2.61	2.63	2.62
	3	4.48	4.39	4.39	4.39	4.45
	4	5.57	5.45	5.45	5.45	5.51
	5	6.22	6.05	6.05	6.05	6.14
PEPPERS (512×512)	2	2.84	2.84	2.83	2.83	2.83
	3	4.50	4.43	4.43	4.43	4.46
	4	5.55	5.53	5.51	5.53	5.55
	5	6.15	6.10	6.08	6.10	6.07
BIRD (256×256)	2	2.33	2.31	2.34	2.34	2.33
	3	3.62	3.62	3.62	3.62	3.59
	4	4.42	4.38	4.38	4.38	4.41
	5	4.81	4.78	4.77	4.78	4.75
CAMERA (256×256)	2	2.27	2.23	2.24	2.27	2.26
	3	3.85	3.81	3.81	3.81	3.84
	4	4.61	4.45	4.45	4.45	4.60
	5	4.69	4.68	4.68	4.68	4.69
GOLDHILL (256×256)	2	3.16	3.18	3.15	3.15	3.16
	3	4.71	4.68	4.68	4.68	4.70
	4	5.70	5.62	5.62	5.62	5.64
	5	6.24	6.17	6.17	6.18	6.16
24077 (481×321)	2	2.87	2.85	2.84	2.85	2.87
	3	3.91	3.88	3.88	2.88	2.89
	4	4.64	4.59	4.58	4.58	4.61
	5	5.12	5.04	5.05	5.05	5.09
89072 (321×481)	2	3.02	3.01	3.01	3.01	3.02
	3	4.36	4.33	4.34	4.33	4.34
	4	5.23	5.20	5.21	5.21	5.23
	5	5.98	5.91	5.92	5.92	5.93
42049 (481×321)	2	2.19	2.18	2.17	2.17	2.18
	3	3.24	3.21	3.22	3.22	3.22
	4	3.98	3.94	3.95	3.95	3.96
	5	4.45	4.39	4.38	4.39	4.41
86016 (481×321)	2	2.23	2.22	2.21	2.22	2.23
	3	3.41	3.39	3.38	3.38	3.40
	4	4.12	4.08	4.08	4.09	4.10
	5	4.67	4.62	4.63	4.62	4.62
54082 (321×481)	2	2.35	2.34	2.34	2.34	2.35
	3	4.23	4.21	4.21	4.22	4.23
	4	5.52	5.48	5.49	5.49	5.51
	5	6.31	6.25	6.24	6.25	6.28
21077 (481×321)	2	2.76	2.75	2.75	2.75	2.76
	3	3.63	3.61	3.62	3.62	3.62
	4	4.32	4.28	4.29	4.28	4.30
	5	4.86	4.81	4.80	4.80	4.79
160068 (481×321)	2	2.31	2.30	2.31	2.30	2.30
	3	3.83	3.81	3.80	3.81	3.80
	4	4.76	4.71	4.70	4.71	4.72
	5	5.56	5.49	5.51	5.49	5.50

Table 3: Continued.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
37073	2	2.64	2.62	2.63	2.59	2.64
(481×321)	3	4.30	4.26	4.25	4.26	4.28
	4	4.95	4.93	4.92	4.91	4.90
	5	5.28	5.25	5.23	5.18	5.24
66053	2	3.16	3.15	3.14	3.15	3.15
(481×321)	3	4.56	4.51	4.51	4.52	4.51
	4	5.43	5.38	5.37	5.39	5.39
	5	6.19	6.12	6.13	6.12	6.14
304074	2	3.13	3.12	3.10	3.14	3.13
(321×481)	3	4.09	4.07	4.06	4.08	4.09
	4	4.76	4.70	4.71	4.73	4.73
	5	5.21	5.16	5.15	5.15	5.18

Table 4: Root Mean Squared Error (RMSE) of different segmented images using various algorithm.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
LENA (512×512)	2	17.96	17.97	18.07	18.03	18.01
	3	17.21	17.53	17.67	17.34	17.30
	4	16.47	16.76	16.83	16.61	16.52
	5	15.67	16.07	16.18	16.16	15.89
PEPPERS (512×512)	2	18.14	18.16	18.15	18.14	18.15
	3	17.41	17.58	17.60	17.57	17.52
	4	16.51	16.68	16.70	16.66	16.57
	5	16.11	16.18	16.21	16.20	16.14
BIRD (256×256)	2	18.16	18.17	18.22	18.17	18.19
	3	17.63	17.75	17.73	17.74	17.71
	4	16.07	16.16	16.24	16.21	16.15
	5	15.26	15.47	15.41	15.38	15.32
CAMERA (256×256)	2	18.06	18.08	18.07	18.07	18.06
	3	16.61	16.73	16.84	16.76	16.64
	4	16.04	16.22	16.22	16.21	16.10
	5	15.63	15.74	15.86	15.81	15.76
GOLDHILL (256×256)	2	17.70	17.78	17.80	17.76	17.72
	3	17.19	17.27	17.31	17.29	17.23
	4	16.78	16.84	16.86	16.80	16.78
	5	15.83	16.03	16.06	16.01	15.96
24077 (481×321)	2	17.21	17.24	17.25	17.21	17.22
	3	16.92	17.05	17.10	16.96	16.95
	4	16.82	16.94	16.99	16.87	16.85
	5	16.56	16.63	16.79	16.66	16.63
89072 (321×481)	2	18.10	18.11	18.12	18.10	18.11
	3	18.02	18.00	18.08	18.01	18.03
	4	17.34	17.38	17.43	17.40	17.34
	5	16.96	17.12	17.19	17.17	17.05
42049 (481×321)	2	18.03	18.09	18.13	18.11	18.05
	3	17.72	17.79	17.82	17.76	17.73
	4	16.72	16.71	16.72	16.73	16.73
	5	15.90	15.92	15.92	15.93	15.92

Table 4: Continued.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
86016	2	18.00	18.00	18.02	18.00	18.02
(481×321)	3	17.45	17.57	17.58	17.56	17.46
	4	17.21	17.31	17.31	17.25	17.22
	5	15.69	15.81	15.78	15.83	15.74
54082	2	17.92	17.95	18.00	18.00	17.92
(321×481)	3	17.11	17.14	17.16	17.14	17.13
	4	15.33	15.49	15.67	15.51	15.33
	5	14.76	14.80	14.96	14.89	14.80
21077	2	17.06	17.17	17.22	17.12	17.12
(481×321)	3	16.35	16.53	16.56	16.48	16.41
	4	16.12	16.17	16.22	16.19	16.12
	5	15.15	15.21	15.29	15.22	15.17
160068	2	18.21	18.23	18.24	18.24	18.22
(481×321)	3	17.89	17.95	17.98	19.96	17.89
	4	16.61	16.74	16.79	16.78	16.71
	5	15.59	15.62	15.81	15.65	15.59
37073	2	18.50	18.53	18.55	18.53	18.50
(481×321)	3	18.09	18.16	18.21	18.12	18.09
	4	16.27	16.38	16.48	16.37	16.27
	5	15.41	15.81	15.92	15.70	15.40
66053	2	18.47	18.50	18.52	18.50	18.49
(481×321)	3	18.01	18.07	18.05	18.06	18.02
	4	16.89	16.97	17.13	17.03	16.94
	5	16.31	16.42	16.51	16.46	16.33
304074	2	17.72	17.74	17.76	17.73	17.72
(321×481)	3	17.28	17.35	17.47	17.31	17.28
	4	16.30	16.38	16.46	16.44	16.35
	5	16.01	16.04	16.09	16.04	16.02

Table 5: Peak Signal to Noise Ratio (PSNR) of different segmented images using various algorithm.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
LENA (512×512)	2	23.06	23.04	22.99	23.01	23.02
	3	23.60	23.26	23.19	23.35	23.37
	4	23.89	23.65	23.61	23.72	23.77
	5	24.10	24.01	23.95	23.96	24.11
PEPPERS (512×512)	2	23.23	22.95	22.95	22.96	22.95
	3	23.67	23.23	23.22	23.24	23.26
	4	23.92	23.69	23.68	23.70	23.74
	5	24.17	23.95	23.94	23.94	23.97
BIRD (256×256)	2	23.18	22.94	22.92	22.94	22.93
	3	23.57	23.15	23.16	23.15	23.17
	4	23.92	23.96	23.92	23.94	23.97
	5	24.15	24.34	24.37	24.39	24.43
CAMERA (256×256)	2	23.01	22.99	22.99	22.99	23.00
	3	23.42	23.66	23.60	23.65	23.71
	4	23.75	23.93	23.93	23.94	23.99
	5	23.92	24.19	24.12	24.15	24.18

Table 5: Continued.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
	2	22.66	23.13	23.12	23.14	23.16
GOLDHILL (256×256)	3	22.87	23.38	23.36	23.37	23.41
	4	23.08	23.60	23.59	23.62	23.63
	5	23.20	24.03	24.02	24.04	24.07
24077	2	23.01	23.40	23.40	23.42	23.41
(481×321)	3	23.44	23.50	23.47	23.54	23.55
	4	23.75	23.55	23.53	23.59	23.60
	5	23.99	23.71	23.63	23.70	23.71
89072	2	22.98	22.97	22.97	22.98	22.97
(321×481)	3	23.34	23.03	22.99	23.02	23.01
	4	23.57	23.33	23.30	23.32	23.35
	5	23.67	23.46	23.43	23.44	23.50
42049	2	23.90	22.98	22.96	22.97	23.00
(481×321)	3	24.34	23.13	23.11	23.14	23.16
	4	24.46	23.67	23.67	23.66	23.66
	5	24.58	24.09	24.09	24.09	24.09
86016	2	22.68	23.03	23.02	23.03	23.02
(481×321)	3	23.02	23.24	23.23	23.24	23.29
	4	23.33	23.36	23.36	23.40	23.41
	5	23.48	24.15	24.17	24.14	24.19
54082	2	23.23	23.05	23.03	23.03	23.06
(321×481)	3	23.42	23.45	23.44	23.45	23.46
	4	23.58	24.33	24.23	24.32	24.42
	5	23.65	24.73	24.63	24.67	24.73
21077	2	22.77	23.44	23.41	23.46	23.46
(481×321)	3	23.01	23.77	23.75	23.79	23.83
	4	23.24	23.96	23.93	23.95	23.98
	5	23.32	24.49	24.44	24.48	24.51
160068	2	22.74	22.92	22.91	22.91	22.92
(481×321)	3	23.08	23.05	23.04	22.13	23.08
	4	23.30	23.66	23.63	23.63	23.67
	5	23.37	24.26	24.15	24.24	24.27
37073	2	23.41	22.77	22.76	22.77	22.79
(481×321)	3	23.68	22.95	22.92	22.97	22.98
	4	23.82	23.84	23.79	23.85	23.90
	5	23.87	24.15	24.09	24.21	24.38
66053	2	22.67	22.79	22.78	22.79	22.79
(481×321)	3	22.99	22.99	23.00	23.00	23.02
	4	23.25	23.54	23.46	23.51	23.55
	5	23.28	23.82	23.78	23.80	23.87
304074	2	22.97	23.15	23.14	23.16	23.16
(321×481)	3	23.15	23.34	23.28	23.36	23.38
	4	23.30	23.84	23.80	23.81	23.86
	5	23.35	24.03	24.00	24.03	24.04

Table 6: Structural Similarity Index (SSIM) of different segmented images using various algorithm.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
LENA (512×512)	2	0.56	0.54	0.55	0.54	0.55
	3	0.74	0.72	0.73	0.72	0.74
	4	0.81	0.78	0.79	0.78	0.80
	5	0.90	0.87	0.88	0.87	0.89
PEPPERS (512×512)	2	0.63	0.61	0.61	0.62	0.63
	3	0.78	0.75	0.75	0.77	0.78
	4	0.85	0.82	0.82	0.83	0.84
	5	0.94	0.90	0.91	0.92	0.93
BIRD (256×256)	2	0.57	0.56	0.55	0.56	0.57
	3	0.68	0.66	0.65	0.66	0.68
	4	0.87	0.84	0.83	0.85	0.86
	5	0.91	0.88	0.87	0.89	0.90
CAMERA (256×256)	2	0.71	0.69	0.70	0.69	0.70
	3	0.81	0.77	0.78	0.77	0.79
	4	0.94	0.91	0.92	0.91	0.93
	5	0.97	0.94	0.93	0.94	0.96
GOLDHILL (256×256)	2	0.55	0.54	0.54	0.53	0.55
	3	0.78	0.76	0.76	0.75	0.77
	4	0.85	0.83	0.84	0.82	0.84
	5	0.91	0.88	0.89	0.88	0.90
24077 (481×321)	2	0.79	0.79	0.78	0.78	0.78
	3	0.84	0.83	0.83	0.84	0.82
	4	0.88	0.86	0.86	0.85	0.86
	5	0.94	0.93	0.93	0.92	0.93
89072 (321×481)	2	0.62	0.62	0.62	0.62	0.62
	3	0.76	0.75	0.74	0.75	0.74
	4	0.82	0.81	0.80	0.81	0.82
	5	0.88	0.87	0.86	0.87	0.87
42049 (481×321)	2	0.76	0.75	0.76	0.76	0.76
	3	0.83	0.82	0.81	0.82	0.83
	4	0.92	0.92	0.91	0.92	0.92
	5	0.97	0.95	0.96	0.96	0.97
86016 (481×321)	2	0.54	0.54	0.54	0.54	0.54
	3	0.66	0.65	0.64	0.64	0.66
	4	0.73	0.73	0.71	0.71	0.73
	5	0.92	0.90	0.91	0.90	0.89
54082 (321×481)	2	0.32	0.31	0.31	0.31	0.30
	3	0.38	0.37	0.37	0.37	0.36
	4	0.60	0.59	0.59	0.58	0.56
	5	0.92	0.91	0.91	0.91	0.90
21077 (481×321)	2	0.48	0.48	0.48	0.47	0.47
	3	0.73	0.72	0.71	0.71	0.71
	4	0.89	0.88	0.88	0.87	0.87
	5	0.93	0.92	0.92	0.91	0.91
160068 (481×321)	2	0.61	0.60	0.59	0.59	0.60
	3	0.67	0.65	0.64	0.64	0.64
	4	0.72	0.69	0.69	0.69	0.68
	5	0.77	0.74	0.73	0.73	0.73

Table 6: Continued.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
37073	2	0.33	0.32	0.32	0.31	0.32
(481×321)	3	0.66	0.65	0.64	0.64	0.65
	4	0.82	0.80	0.80	0.80	0.81
	5	0.89	0.87	0.87	0.87	0.88
66053	2	0.47	0.44	0.44	0.44	0.45
(481×321)	3	0.79	0.77	0.77	0.77	0.78
	4	0.84	0.82	0.82	0.81	0.83
	5	0.88	0.86	0.86	0.85	0.87
304074	2	0.55	0.54	0.55	0.54	0.54
(321×481)	3	0.69	0.65	0.66	0.65	0.66
	4	0.78	0.75	0.77	0.76	0.77
	5	0.86	0.82	0.83	0.82	0.82

Table 7: Time cost of diffident methods (unit: millisecond).

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
LENA (512×512)	2	1.38	1.68	1.54	1.62	1.49
	3	2.39	2.68	2.57	2.64	2.51
	4	30.94	34.78	33.14	33.57	31.89
	5	204.39	241.38	234.25	238.14	224.64
PEPPERS (512×512)	2	1.46	1.82	1.67	1.74	1.54
	3	2.79	3.12	2.97	3.04	2.84
	4	31.64	35.78	33.79	34.51	32.59
	5	242.41	284.12	271.25	279.49	261.41
BIRD (256×256)	2	1.12	1.43	1.37	1.41	1.24
	3	1.98	2.24	2.14	2.19	2.07
	4	18.64	21.74	20.91	21.42	19.75
	5	134.25	152.24	145.98	148.62	140.91
CAMERA (256×256)	2	1.19	1.49	1.41	1.45	1.26
	3	2.14	2.38	2.27	2.31	2.21
	4	28.56	32.85	31.71	32.24	29.83
	5	184.25	217.39	207.95	211.24	199.27
GOLDHILL (256×256)	2	1.42	1.78	1.63	1.70	1.49
	3	2.46	2.98	2.61	2.89	2.61
	4	34.25	39.41	37.81	38.45	35.78
	5	221.35	251.47	237.18	246.91	230.43
24077 (481×321)	2	1.51	1.83	1.69	1.77	1.59
	3	2.43	2.90	2.73	2.79	2.63
	4	29.43	34.12	31.43	32.64	30.57
	5	176.89	193.19	186.27	182.57	179.95
89072 (321×481)	2	1.46	1.81	1.64	1.76	1.55
	3	2.41	2.84	2.69	2.76	2.61
	4	29.67	34.12	31.49	32.68	30.41
	5	166.84	183.96	176.66	179.91	173.09
42049 (481×321)	2	1.25	1.53	1.42	1.47	1.35
	3	2.16	2.47	2.31	2.38	2.23
	4	23.14	27.06	25.42	26.71	24.58
	5	154.12	168.22	162.59	164.36	158.71

Table 7: Continued.

Image	level	Our algorithm	BA	MMSA	IBA	OTSU
86016	2	1.34	1.61	1.48	1.51	1.40
(481×321)	3	2.21	2.51	2.37	2.43	2.31
	4	25.16	29.45	27.64	28.85	26.74
	5	159.54	176.21	167.66	171.65	164.39
54082	2	1.41	1.76	1.61	1.68	1.52
(321×481)	3	2.34	2.79	2.63	2.70	2.52
	4	28.62	32.55	30.06	31.43	29.59
	5	164.59	180.64	174.01	177.39	170.42
21077	2	1.55	1.89	1.71	1.83	1.62
(481×321)	3	2.49	2.99	2.76	2.83	2.69
	4	30.09	35.07	32.16	33.18	31.55
	5	167.46	185.03	178.14	182.67	173.24
160068	2	1.49	1.79	1.63	1.72	1.54
(481×321)	3	2.38	2.81	2.66	2.72	2.58
	4	29.14	33.45	30.94	32.09	30.17
	5	166.01	182.73	176.10	178.93	172.23
37073	2	1.39	1.71	1.58	1.64	1.44
(481×321)	3	2.36	2.80	2.61	2.73	2.55
	4	29.04	33.01	30.58	31.91	29.91
	5	165.04	182.36	175.10	178.27	171.34
66053	2	1.58	1.93	1.76	1.88	1.65
(481×321)	3	2.53	3.08	2.81	2.92	2.75
	4	30.59	35.89	32.77	33.81	31.98
	5	198.47	227.55	211.13	221.34	206.23
304074	2	1.62	2.03	1.82	1.96	2.74
(321×481)	3	2.61	3.18	2.93	3.06	2.82
	4	33.18	38.69	35.18	36.87	34.18
	5	231.64	264.31	252.34	249.23	240.17

the algorithm proposed in this paper can achieve larger Information Entropy (IE), higher Peak Signal to Noise Ratio (PSNR), higher Structural Similarity Index (SSIM), smaller Root Mean Squared Error (RMSE), and shorter time than other image segmentation algorithms.

Data Availability

The data used to support the research findings of this study have been deposited in "https://pan.baidu.com/s/1UhHjhFnvfqS2Po0QUPIxzA" and "https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html/dataset/images.html".

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by Subproject of Key Project of Beijing, China (Nos. Z181100000618006 and D161100004916002), Beijing Natural Science Foundation (No. 4192042), and National Natural Science Foundation of China (No. 61627816).

Supplementary Materials

Test all the 100 pictures in the test dataset of Berkeley Segmentation Dataset to justify the superiority of the proposed approach. *(Supplementary Materials)*

References

[1] T. Wang, J. Yang, Z. Ji, and Q. Sun, "Probabilistic diffusion for interactive image segmentation," *IEEE Transactions on Image Processing*, vol. 28, no. 1, pp. 330–342, 2019.

[2] Y. Zhou and H. Q. Zhu, "Image segmentation using a trimmed likelihood estimator in the asymmetric mixture model based on generalized gamma and gaussian distributions," *Mathematical Problems in Engineering*, vol. 2018, Article ID 3468967, 17 pages, 2018.

[3] S. Kotte, R. K. Pullakura, and S. K. Injeti, "Optimal multilevel thresholding selection for brain MRI image segmentation based

on adaptive wind driven optimization," *Measurement*, vol. 130, pp. 340–361, 2018.

[4] M. A. Hossam, H. M. Ebied, M. H. Abdel-Aziz, and M. F. Tolba, "Accelerated hyperspectral image recursive hierarchical segmentation using GPUs, multicore CPUs, and hybrid CPU/GPU cluster," *Journal of Real-Time Image Processing*, vol. 14, no. 2, pp. 413–432, 2018.

[5] Z. Li and G. Zhang, "Fracture segmentation method based on contour evolution and gradient direction consistency in sequence of coal rock CT images," *Mathematical Problems in Engineering*, vol. 2019, Article ID 2980747, 8 pages, 2019.

[6] M. Sharif, M. A. Khan, Z. Iqbal, M. F. Azam, M. I. Lali, and M. Y. Javed, "Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection," *Computers and Electronics in Agriculture*, vol. 150, pp. 220–234, 2018.

[7] V. P. Ananthi, P. Balasubramanian, and P. Raveendran, "A thresholding method based on interval-valued intuitionistic fuzzy sets: an application to image segmentation," *PAA. Pattern Analysis and Applications*, vol. 21, no. 4, pp. 1039–1051, 2018.

[8] M. I. Daoud, A. A. Atallah, and F. Awwad, "Automatic superpixel-based segmentation method for breast ultrasound images," *Expert Systems with Applications*, vol. 121, pp. 78–96, 2019.

[9] Z. Fan, J. Lu, C. Wei, H. Huang, X. Cai, and X. Chen, "A hierarchical image matting model for blood vessel segmentation in fundus images," *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2367–2377, 2019.

[10] J. Olveres, D. E. Carbaajal, R. B. Escalante et al., "Deformable models for segmentation based on local analysis," *Mathematical Problems in Engineering*, vol. 2017, Article ID 1646720, 13 pages, 2017.

[11] B. Han and Y. Wu, "Active contours driven by global and local weighted signed pressure force for image segmentation," *Pattern Recognition*, vol. 88, pp. 715–728, 2019.

[12] R. Panda, S. Agrawal, L. Samantaray et al., "An evolutionary gray gradient algorithm for multilevel thresholding of brain MR images using soft computing techniques," *Applied Soft Computing*, vol. 50, pp. 94–108, 2017.

[13] A. K. Jumaat and K. Chen, "A reformulated convex and selective variational image segmentation model and its fast multilevel algorithm," *Numerical Mathematics: Theory, Methods and Applications*, vol. 12, no. 2, pp. 403–437, 2019.

[14] E. Essa and X. Xie, "Automatic segmentation of cross-sectional coronary arterial images," *Computer Vision and Image Understanding*, vol. 165, pp. 97–110, 2017.

[15] H. Liang, H. Jia, Z. Xing, J. Ma, and X. Peng, "Modified grasshopper algorithm-based multilevel thresholding for color image segmentation," *IEEE Access*, vol. 7, pp. 11258–11295, 2019.

[16] Y. T. Chen, "Medical image segmentation using independent component analysis-based kernelized fuzzy c -means clustering," *Mathematical Problems in Engineering*, vol. 2017, Article ID 5892039, 21 pages, 2017.

[17] A. R. J. Fredo, R. S. Abilash, and C. Suresh Kumar, "Segmentation and analysis of damages in composite images using multilevel threshold methods and geometrical features," *Measurement*, vol. 100, pp. 270–278, 2017.

[18] W. William, A. Ware, A. H. Basaza-Ejiri, and J. Obungoloch, "A review of image analysis and machine learning techniques for automated cervical cancer screening from pap-smear images," *Computer Methods and Programs in Biomedicine*, vol. 164, pp. 15–22, 2018.

[19] T. Pun, "A new method for grey-level picture thresholding using the entropy of the histogram," *Signal Processing*, vol. 2, no. 3, pp. 223–237, 1980.

[20] K. Chowdhury, D. Chaudhuri, and A. K. Pal, "A new image segmentation technique using bi-entropy function minimization," *Multimedia Tools and Applications*, vol. 77, no. 16, pp. 20889–20915, 2018.

[21] S. Hinojosa, K. G. Dhal, M. A. Elaziz, D. Oliva, and E. Cuevas, "Entropy-based imagery segmentation for breast histology using the stochastic fractal search," *Neurocomputing*, vol. 321, pp. 201–215, 2018.

[22] S. Mishra and M. Panda, "Bat algorithm for multilevel colour image segmentation using entropy-based thresholding," *Arabian Journal for Science and Engineering*, vol. 43, no. 12, pp. 7285–7314, 2018.

[23] S. Pare, A. Kumar, V. Bajaj, and G. K. Singh, "An efficient method for multilevel color image thresholding using cuckoo search algorithm based on minimum cross entropy," *Applied Soft Computing*, vol. 61, pp. 570–592, 2017.

[24] J. W. Long, X. Feng, X. F. Zhu, J. Zhang, and G. Gou, "Efficient superpixel-guided interactive image segmentation based on graph theory," *Symmetry-Basel*, vol. 10, no. 5, p. 169, 2018.

[25] Z. M. Lu, F. C. Zhu, X. Y. Gao, B. C. Chen, and Z. G. Gao, "In-situ particle segmentation approach based on average background modeling and graph-cut for the monitoring of L-glutamic acid crystallization," *Chemometrics and Intelligent Laboratory Systems*, vol. 178, pp. 11–23, 2018.

[26] C. D. Jimenez, P. D. Bermejo, and P. Nardelli, "A graph-cut approach for pulmonary artery-vein segmentation in noncontrast CT images," *Medical Image Analysis*, vol. 52, pp. 144–159, 2019.

[27] H. Zhu, Z. Zhuang, J. Zhou et al., "Improved graph-cut segmentation for ultrasound liver cyst image," *Multimedia Tools and Applications*, vol. 9, pp. 1–19, 2018.

[28] X. Deng, Y. Zheng, Y. Xu, X. Xi, N. Li, and Y. Yin, "Graph cut based automatic aorta segmentation with an adaptive smoothness constraint in 3D abdominal CT images," *Neurocomputing*, vol. 310, pp. 46–58, 2018.

[29] S. G. A. Usha and S. Vasuki, "Improved segmentation and change detection of multi-spectral satellite imagery using graph cut based clustering and multiclass SVM," *Multimedia Tools and Applications*, vol. 77, no. 12, pp. 15353–15383, 2018.

[30] Y. H. Guo, Y. M. Akbulut, A. Şengür et al., "An efficient image segmentation algorithm using neutrosophic graph cut," *Symmetry*, vol. 9, no. 9, p. 185, 2017.

[31] M. A. Diaz-Cortes, S. N. Ortega, S. Hinojosa et al., "A multi-level thresholding method for breast thermo grams analysis using dragonfly algorithm," *Infrared Physics & Technology*, vol. 93, pp. 346–361, 2018.

[32] J. C. Bansal, A. Gopal, and A. K. Nagar, "Stability analysis of artificial bee colony optimization algorithm," *Swarm and Evolutionary Computation*, vol. 41, pp. 9–19, 2018.

[33] L. B. Ma, X. W. Wang, H. Shen et al., "A novel artificial bee colony optimiser with dynamic population size for multi-level threshold image segmentation," *International Journal of Bio-Inspired Computation*, vol. 13, no. 1, pp. 32–44, 2019.

[34] H. Gao, Z. Fu, and C. M. Pun, "A multi-level thresholding image segmentation based on an improved artificial bee colony algorithm," *Computers and Electrical Engineering*, vol. 70, pp. 931–938, 2018.

[35] S. Mishra and M. Panda, "Bat algorithm for multilevel colour image segmentation using entropy-based thresholding," *Arabian Journal for Science and Engineering*, vol. 6, pp. 1–30, 2018.

[36] M. Q. Li, L. P. Xu, N. Xu, T. Huang, and B. Yan, "SAR image segmentation based on improved grey wolf optimization algorithm and fuzzy c-means," *Mathematical Problems in Engineering*, vol. 2018, Article ID 4576015, 11 pages, 2018.

[37] S. Zhang, W. Jiang, and S. Satoh, "Multilevel thresholding color image segmentation using a modified artificial bee colony algorithm," *IEICE Transaction on Information and Systems*, vol. E101D, no. 8, pp. 2064–2071, 2018.

[38] Y. Zhong, R. Gao, and L. Zhang, "Multiscale and multifeature normalized cut segmentation for high spatial resolution remote sensing imagery," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 54, no. 10, pp. 6061–6075, 2016.

[39] A. Alihodzic and M. Tuba, "Improved bat algorithm applied to multilevel image thresholding," *The Scientific World Journal*, vol. 2014, Article ID 176718, 16 pages, 2014.

[40] Y. Zhou, X. Yang, Y. Ling, and J. Zhang, "Meta-heuristic moth swarm algorithm for multilevel thresholding image segmentation," *Multimedia Tools and Applications*, vol. 77, no. 18, pp. 23699–23727, 2018.

[41] S. C. Satapathy, N. S. M. Raja, V. Rajinikanth et al., "Multilevel image thresholding using Otsu and chaotic bat algorithm," *Neural Computing and Applications*, vol. 29, no. 12, pp. 1285–1307, 2018.

[42] https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/BSDS300/html/dataset/images.