Applied Soft Computing 48 (2016) 151–159

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

A multilevel image thresholding segmentation algorithm based on two-dimensional K–L divergence and modified particle swarm optimization

Xiaoli Zhao [a,c,*], Matthew Turk [b], Wei Li [d], Kuo-chin Lien [b], Guozhong Wang [a]

[a] School of Communication and Information Engineering, Shanghai University, Shanghai 200444, China
[b] Department of Computer Science, University of California, Santa Barbara, CA 93106, USA
[c] School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China
[d] Department of Computer Science and Technology, Zhejiang University, Hangzhou 310027, China

ARTICLE INFO

Article history:
Received 30 September 2015
Received in revised form 16 June 2016
Accepted 8 July 2016
Available online 15 July 2016

Keywords:
2D K–L divergence
Modified PSO
Multilevel image thresholding segmentation

ABSTRACT

Multilevel image segmentation is a technique that divides images into multiple homogeneous regions. In order to improve the effectiveness and efficiency of multilevel image thresholding segmentation, we propose a segmentation algorithm based on two-dimensional (2D) Kullback–Leibler(K–L) divergence and modified Particle Swarm Optimization (MPSO). This approach calculates the 2D K–L divergence between an image and its segmented result by adopting 2D histogram as the distribution function, then employs the sum of divergences of different regions as the fitness function of MPSO to seek the optimal thresholds. The proposed 2D K–L divergence improves the accuracy of image segmentation; the MPSO overcomes the drawback of premature convergence of PSO by improving the location update formulation and the global best position of particles, and reduces drastically the time complexity of multilevel thresholding segmentation. Experiments were conducted extensively on the Berkeley Segmentation Dataset and Benchmark (BSDS300), and four performance indices of image segmentation – BDE, PRI, GCE and VOI – were tested. The results show the robustness and effectiveness of the proposed algorithm.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

Image segmentation is the process of partitioning an image into non-overlapping, homogeneous regions containing similar objects, which is widely used in computer vision applications such as object recognition, content-based retrieval, and object-based video coding. Though numerous algorithms have been proposed in recent years, image segmentation is still far from being an easy problem to solve.

At present, commonly used segmentation algorithms include graph cut [1,2], contour detection [3], and thresholding segmentation [4,5]. Thresholding segmentation has been widely adopted due to its simplicity, which consists of bi-level and multilevel segmentation. Bi-level segmentation splits an image into object and background. However, with the advent of multi-object technology such as multi-object optimization and multi-object tracking, dividing an image into just two regions cannot meet the requirements of most pattern recognition and machine vision applications. To overcome the limitations of bi-level segmentation, multilevel segmentation approaches that split an image into multiple objects and background have been developed. However, conventional multilevel image thresholding segmentation methods [6,7] are costly in time since they search exhaustively for the optimal multiple threshold values of the image. Moreover, the time complexity grows exponentially with the number of thresholds.

Metaheuristics provide a very popular way to yield near optimal solutions for a wide variety of complex optimization problems [8]. They combine rules and randomness to imitate some natural phenomena, and have grown significantly in usage in the past few decades. If we treat multiple threshold values as space dimensions of metaheuristics, their parallelism provides an efficient means to address the problem of computational cost of multilevel image thresholding segmentation. The PSO algorithm [9], introduced by Eberhart and Kennedy in 1995, is a swarm-based metaheuristic technique that models the social behavior of bird flocking. It is well adapted to the optimization of nonlinear functions in multidimensional space. PSO is easy to implement

* Corresponding author at: School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China.
E-mail address: evawhy@163.com (X. Zhao).

http://dx.doi.org/10.1016/j.asoc.2016.07.016
1568-4946/© 2016 Elsevier B.V. All rights reserved.

and may outperform other evolutionary algorithms [10,11]; However, it is liable to trap local optimization and cause premature convergence.

The fitness function of metaheuristics is a criterion for selecting the optimal solution, and in recent years, using entropy as a fitness function has drawn the attention of researchers [8,12]. Many entropy-based segmentation algorithms adopt one-dimensional (1D) entropy to calculate the summation of regional entropy as an objective function of image segmentation [13,14]. However, 1D entropy only takes account of gray values of the image with no spatial correlation between pixels, and thus the performance is usually unsatisfactory. Abutaleb [15] extended 1D entropy to 2D, regarding the 2D histogram as a distribution function to calculate entropy and achieving more accurate segmentation results than those based on 1D entropy. We also utilize the idea of 2D entropy in our segmentation algorithm proposed in this paper.

Kullback–Leibler(K–L) divergence was proposed by Kullback [16] under the name of directed divergence, which is a relative entropy to measure the information theoretical distance between two distributions P and Q. It was also studied by *Rényi* [17], who pointed out that the K–L divergence can be interpreted as the expectation of the change in the information content when substituting Q for P. The less the information changes, the smaller the divergence is, and the divergence will be zero if two distributions are the same. According to these characteristics of K–L divergence, when we regard image region and its corresponding segmentation result as P and Q respectively, minimizing the sum of K–L divergence of different regions in an image is to search for the optimal image segmentation. Because the searching process is time-consuming, PSO is introduced to reduce computational cost by using its parallelism in our work.

In this paper, we propose an unsupervised algorithm for multilevel image thresholding segmentation which combines 2D K–L divergence and modified PSO (2DKLMPSO). 2D histogram of an image is denoted as the distribution function to calculate the proposed 2D K–L divergence, which is then considered as the fitness function of MPSO to improve the accuracy of multilevel image segmentation. To reduce the time complexity of seeking the optimal threshold values of multilevel thresholding segmentation, we propose MPSO to manage the convergence and diversity of particles to conquer the drawback of premature convergence of PSO. Extensive experiments conducted on the BSDS300 illustrate that the proposed algorithm not only achieves better segmentation results, but also has lower time complexity.

The main contributions of this paper are as follows:

(1) We propose the 2D K–L divergence to be applied to multilevel image segmentation, and derive the formulation of 2D K–L divergence as an objective function of multilevel image segmentation to improve the accuracy of segmentation;
(2) We propose MPSO that modifies the location update formula and the global best position of particles to overcome the defect of premature convergence of PSO;
(3) We propose a scheme for multilevel image thresholding segmentation that denotes 2D K–L divergence as the fitness function of MPSO, which improves the effectiveness of the segmentation and reduces the time complexity.

The rest of this paper is organized as follows. We summarize the development of multilevel image thresholding segmentation in Section 2, followed by a brief introduction of K–L divergence, 2D histogram concept and PSO in Section 3. Section 4 describes the proposed algorithm. Section 5 illustrates our experimental results. Conclusions are drawn in Section 6.

2. Related works

Roughly speaking, the development of multilevel image thresholding segmentation algorithm has gone through two stages. Early stage approaches, such as between-class variance [6] and entropy [7], which are unsupervised and nonparametric, explore exhaustively the optimal threshold values to optimize the objective function. Although those segmentation approaches can produce accurate results, they are extremely time-consuming. To reduce the computational complexity, iterative schemes [18,19] and dichotomization techniques [20] were developed, which significantly reduced the computing time while producing results of comparable quality.

In recent years, many researchers have focused on metaheuristics in multilevel image thresholding segmentation domain to reduce computational cost. The commonly used metaheuristic algorithms include Genetic Algorithm (GA) [21,22], Particle Swarm Optimization (PSO) [23,24] and Differential Evolution (DE) [25]. PSO is characterized as simple in concept, easy to implement, and computationally efficient when compared with other heuristic techniques, so PSO has been widely discussed in many domains in recent years [23,26]. However, since the traditional PSO method tends to get stuck in local optima, many subsequent approaches based on PSO have been proposed. Wu et al. [27] developed guidelines for parameter settings both for PSO and discrete PSO to produce better results. Quantum inspired PSO [28] was proposed to reduce computational complexity for multilevel image segmentation. Hu et al. [29] studied an intelligent selection mechanism to trigger appropriate search methods. Liu et al. [11] adopted adaptive inertia and adaptive population to improve the performance of PSO.

Bhandari et al. [30] studied Cukoo search algorithm and wind driven optimization. Tuba et al. [31] compared evolutionary and swarm-based computational techniques. In order to consider spatial contextual information of the image, energy function combined GA [32] has been used. Yin et al. [33] employed the fuzzy c-partition entropy and ABC to select thresholding, and used graph cut instead of thresholing for each pixel to oversegment the image into small regions. Akay [12] used Kapur's entropy and Otsu as fitness function of PSO and artificial bee colony (ABC) to compare segmentation results, and pointed out Kapur's entropy was better than Otsu for multilevel image thresholding segmentation. Cross entropy combined DE algorithm was studied in color image by Sarkar et al. [8]. These methods use evolution algorithms or swarm-based algorithms to seek the optimal multilevel thresholding. Different objective functions, such as an energy function, Otsu and entropy, are adopted as the fitness function of metaheuristic algorithm. Entropy-based approaches, have drawn the attention of many researchers [34] because they have a solid foundation in information theory. The above-mentioned entropy approaches all employed 1D entropy as the objective function of image segmentation.

To involve spatial correlation between pixels in image segmentation, some researchers adopted 2D entropy to achieve better segmentation performance. Qi [35] combined adaptive PSO and 2D Shannon exponential entropy to pursue the optimal threshold values of image segmentation. However, he only divided the image into object and background, and did not implement multilevel thresholding segmentation. Sarkar et al. [36] proposed an automatic multilevel image thresholding scheme based on 2D Tsallis entropy and DE algorithm. They analyzed the Tsallia entropy and focused on how to calculate the maximum entropy of multilevel images. Nie [37] introduced 2D Tsallis cross-entropy to determine the optimal threshold value, which shows it as a new criterion of image thresholding, but it belongs to bi-level, not multilevel, segmentation.

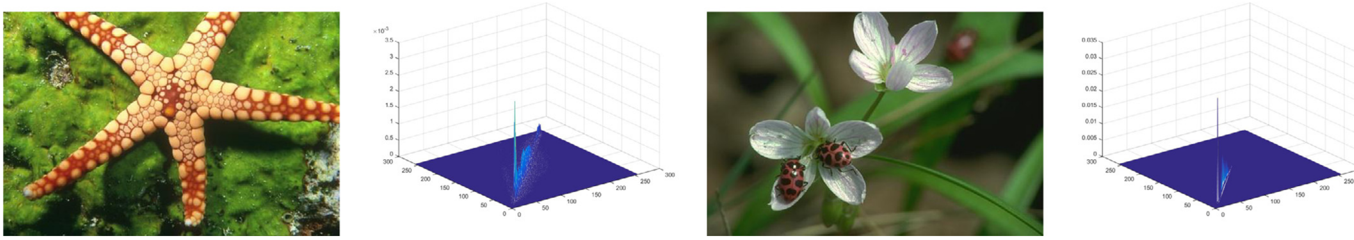

Fig. 1. Images and their 2D histograms.

In this paper, we propose a multilevel image thresholding segmentation algorithm, which adopts the MPSO to reduce the time complexity of algorithm, and regards the 2D K–L divergence as the fitness function of MPSO to seek the optimal threshold values of an image.

3. Background material

In this section, we will introduce the mathematical foundation for the proposed multilevel image thresholding segmentation algorithm.

3.1. K–L divergence [38]

K–L divergence is a measure of the difference between two probability distribution P and Q. It is not symmetric in P and Q. In applications, P typically represents the true distribution of data, while Q represents the approximation of P. It is the amount of information lost when Q is used to approximate P.

Consider $\alpha \in [0, 1) \cup (1, \infty)$ and let $P : \chi \to [0, 1]$ and $Q : \chi \to [0, 1]$ be two distributions, then

$$D_\alpha(P\|Q) = \log\left[\left(\sum_x \frac{P(x)^\alpha}{Q(x)^{\alpha-1}}\right)^{\frac{1}{\alpha-1}}\right], \tag{1}$$

where $D_\alpha(P\|Q)$ is *Rényi* divergence, which is understood as distance measure between two distributions. The more similar P and Q are, the smaller $D_\alpha(P\|Q)$ is.

If $\alpha \to 1$, the limit of *Rényi* divergence $D_1(P\|Q) = \lim_{\alpha\to 1} D_\alpha(P\|Q)$ coincides with

$$D(P\|Q) = \sum_x P(x)\log\frac{P(x)}{Q(x)}. \tag{2}$$

Eq. (2) describes the Kullback–Leibler divergence, which requires P and Q to have the same dimension, and $\sum_{x_i} p(x_i) = \sum_{x_i} q(x_i)$. In image segmentation, let P be the original image and Q be the corresponding segmented result.

3.2. Two-dimensional histogram

Let $g(x, y)$ be the average image of $f(x, y)$ by spatial filtering with a 3×3 window; then the 2D histogram $p_{i,j}$ is constructed as:

$$p_{i,j} = \frac{num(f(x, y) = i \cap g(x, y) = j)}{m \times n}, \quad i, j \in \{0, 1, 2, \cdots, 255\} \tag{3}$$

where $m \times n$ is the image size and num denotes the number of pixels whose gray value equals i and average value equals j.

Fig. 1 shows two images from the BSDS and their histograms. As is evident in figure, the 2D histogram is primarily located along the diagonal of the 2D histogram plane because the gray values of the average image are close to those of the original image.

In terms of above characteristics of the 2D histogram, multilevel threshold values are selected according to Fig. 2, which shows a 2D histogram plane with 4-level thresholding. Similarly, n-level thresholding can be extended. $t1$, $t2$, $t3$ are threshold values of the original image, and $s1$, $s2$, $s3$ are those of the average image. The 2D histogram in regions 1, 6, 11, and 16 come from objects and background, while other regions show edges and noise that are negligible as their frequency is small [36]. When calculating the objective function of image segmentation, only regions 1, 6, 11, and 16 are considered.

3.3. Particle swarm optimization (PSO)

PSO seeks iteratively the optimal solution in n-dimensional space with K particles. Every particle owns two features – location and velocity – when it is flying in space. The j-th velocity and location of the i-th particle [39] can be updated as follows:

$$\begin{aligned} v_{ij}(t+1) = {} & w \times v_{ij}(t) + c_1 \times rand_1 \times (pBest_j(t) - x_{ij}(t)) \\ & + c_2 \times rand_2 \times (gBest_j(t) - x_{ij}(t)) \end{aligned} \tag{4}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{5}$$

where t is the current iteration number, $i = 1, 2, \ldots, K$ denotes the index of a particle, and $j = 1, 2, \ldots, n$ denotes the space dimension. c_1 and c_2 are positive constants, which are called cognitive and social parameters, respectively. $rand_1$ and $rand_2$ are random numbers which meet normal distribution within the range [0, 1]. $pBest_j$ represents the individual best location of the jth dimension of particle i, and $gBest_j(t)$ denotes the global best location of the jth dimension of particle swarm. The inertial weight w controls the impact of previous velocity on current velocity. In this paper, we adopt adaptive inertia weight [10] $w = w_{\max} - (w_{\max} - w_{\min})t/I$

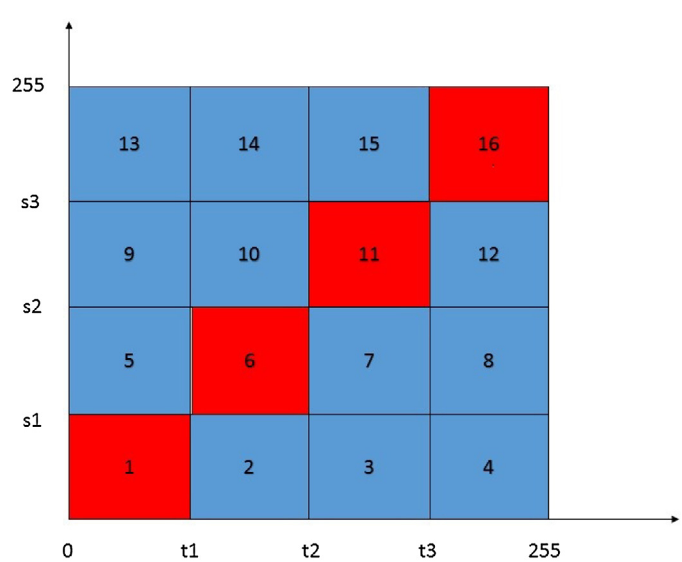

Fig. 2. Two-dimensional histogram plane with 4-level segmentation.

to update the velocity. $w_{\max}$ and $w_{\min}$ are set as 0.9 and 0.4 respectively. I is the total iteration number. In order to prevent the swarm from explosion, the velocity should range between $[v_{\min}\ \ v_{\max}]$. The particle location is similarly restricted to the range of $[x_{min}\ x_{max}]$.

4. The proposed algorithm

In this section, we firstly analyze the relationship between divergence and image segmentation, and explain why we adopt the minimum divergence as the objective function of image segmentation. Then we derive in detail how to calculate the objective function of multilevel thresholding by 2D K–L divergence. We also modify the classical PSO to balance the convergence and diversity of particles. By using 2D K–L divergence as the fitness function of MPSO to seek the optimal threshold values, we build the bridge between them.

4.1. Relationship of K–L divergence and image segmentation

For the purpose of clarity, we begin our discussion with the 1D histogram where only the gray value of each pixel in the image is considered, and then extend to the 2D histogram which further takes account of the spatial correlation between pixels. The segmentation process can be posed as one of reconstruction of the image distribution, which means that the task of image segmentation is to make the segmented image similar to the original image; that is, the information difference between the original image and the segmentation result should be as small as possible. Recall that Eq. (2) calculates the divergence between P and Q. Now let

$$P = \{p_0, p_1, \ldots, p_{255}\} \in \{0, 1, \ldots, N\}$$

and

$$Q = \{q_{v_1}, q_{v_2}, \ldots, q_{v_l}, \ldots, q_{v_L}\} \in \{0, 1, \ldots, N\}$$

be the number of pixels that have that gray level in the original image and its corresponding multilevel segmented result, respectively, where N denotes the number of pixels of an image, L represents the segmentation levels, and v_l, $l \in \{0, 1, \ldots, L\}$ is the gray value in the segmented image. Thus the divergence computed according to Eq. (2) represents the information change between the original image and the segmented output. The K–L divergence, namely the objective function for multilevel image segmentation, can be represented as:

$$D(P\|Q) = \sum_{l=1}^{L} \sum_{i=t_{l-1}}^{t_l} ip_i \log \frac{ip_i}{v_l}, \tag{6}$$

where $t_0 < t_1 < \cdots < t_{L-1} < t_L$ are the threshold values with $t_0 = 0$ and $t_L = 255$. According to the constraint of K–L divergence, the sum of all the pixel values in each region of the segmented image should equal to that of the original image, which means:

$$v_l = \frac{\sum_{i=t_{l-1}}^{t_l} ip_i}{\sum_{i=t_{l-1}}^{t_l} p_i}. \tag{7}$$

In image segmentation, we expect that the information between original and segmented images changes as little as possible; that is, the goal of the image segmentation is to seek the optimal threshold values that minimize the objective function:

$$\{t_1^*, \cdots, t_{L-1}^*\} = arg \min_{\{t_1, \cdots, t_{L-1}\}} D(P\|Q). \tag{8}$$

4.2. Image segmentation algorithm based on 2D K–L divergence

The 1D histogram only takes account of grayscale information of an image, with no spatial correlation between pixels. However, each pixel in an image is not isolated, and takes on correlation with its neighbor pixels. For the reason that 2D histogram considers the gray value of each pixel and spatial correlation of pixels, we propose 2D K–L divergence that employs 2D histogram as the distribution function of divergence to make image segmentation more accurate. To extend 1D divergence to 2D, based on the discussion of 2D histogram in Section 3.2, we redefine P as:

$$P = \left\{ \sum_{j=0}^{255} p_{0,j}, \sum_{j=0}^{255} p_{1,j}, \ldots, \sum_{j=0}^{255} p_{i,j}, \ldots, \sum_{j=0}^{255} p_{255,j} \right\}.$$

where $p_{i,j}$ represents the 2D histogram of an image.

As we have shown in Fig. 2, the 2D histogram is primarily located along the diagonal of the 2D histogram plane, so the proposed 2D K–L divergence of an image can be derived from Eq. (6):

$$D(P\|Q) = \sum_{l=1}^{L} \sum_{i=t_{l-1}}^{t_l} \sum_{j=s_{l-1}}^{s_l} ip_{i,j} \log \frac{ip_{i,j}}{v_l}, \tag{9}$$

where $t_0 < t_1 < \cdots < t_{L-1} < t_L$ and $s_0 < s_1 < \cdots < s_{L-1} < s_L$ are the threshold values of an image and its average image respectively, t_0, $s_0 = 0$ and t_L, $s_L = 255$ are the minimum and maximum in the image. $v_l = \frac{\sum_{i=t_{l-1}}^{t_l} \sum_{j=s_{l-1}}^{s_l} ip_{i,j}}{\sum_{i=t_{l-1}}^{t_l} \sum_{j=s_{l-1}}^{s_l} p_{i,j}}$.

For simplicity of computation, Eq. (9) can be reformulated as:

$$D(P\|Q) = \sum_{l=1}^{L} \sum_{i=t_{l-1}}^{t_l} \sum_{j=s_{l-1}}^{s_l} ip_{i,j} \log(ip_{i,j}) - \sum_{l=1}^{L} \sum_{i=t_{l-1}}^{t_l} \sum_{j=s_{l-1}}^{s_l} ip_{i,j} \log v_l. \tag{10}$$

Given an image, since the first part in Eq. (10) is constant, the objective function can be simplified as:

$$R(P\|Q) = -\sum_{l=1}^{L} \sum_{i=t_{l-1}}^{t_l} \sum_{j=s_{l-1}}^{s_l} ip_{i,j} \log v_l. \tag{11}$$

Then the optimal threshold values of multilevel image segmentation can be set according to the following optimization equation:

$$\{t_1^*, \ldots, t_{L-1}^*, s_1^*, \ldots, s_{L-1}^*\} = arg \min_{\{t_1, \ldots, t_{L-1}, s_1, \ldots, s_{L-1}\}} R(P\|Q). \tag{12}$$

The computational complexity of Eq. (12) by exhaustive search grows dramatically with the increasing number of thresholds. To reduce the computational cost, we propose a modified particle swarm optimization algorithm which will be detailed in the next section.

4.3. Modified particle swarm optimization (MPSO)

In the classic PSO algorithm (referred to as CPSO), the particle swarm is initialized randomly in feasible solution space. The velocity and location of particles are updated by the individual best position and the global best position, which make particles move to the optimal solution. the major drawback of CPSO is premature convergence, which incurs a rapid loss of diversity during the evolutionary processing. Therefore, to address this problem, we propose a modified PSO(named MPSO), which adopts an adaptive factor fa to prevent premature convergence and a perturbation operator ga help particles escape from local optima.

Adaptive factor is defined as follows:

$$fa = rand \times (1 - t/I) \tag{13}$$

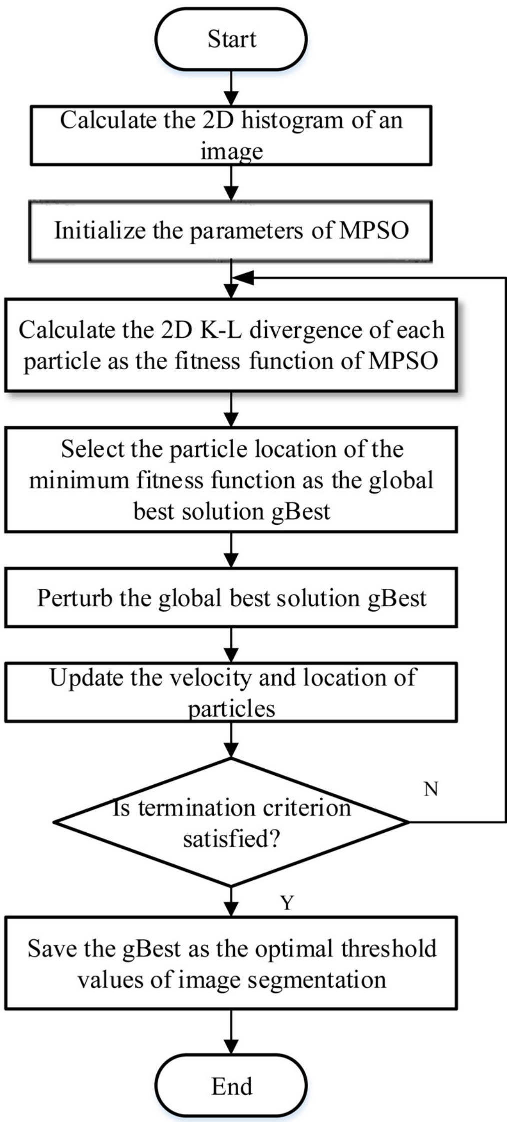

Fig. 3. The flowchart of the proposed algorithm.

where t is the current iteration, I denotes the total iterations, and *rand* meets normal distribution within the range $[0, 1]$. We referred to PSO with adaptive factor as APSO. Formula (13) means that the adaptive factor *fa* tends to decrease with the iterations t. By introducing *fa*, the location of particle is modified as follows:

$$x_{ij}(t+1) = x_{ij}(t) + fa \times v_{ij}(t+1) = x_{ij}(t) + rand \times \left(1 - t/I\right) \times v_{ij}(t+1) \quad (14)$$

It can be seen in Formula (14) that in the early stage of the iteration, *fa* is large, and the algorithm has strong ability to explore new areas continuously to overcome premature convergence, then as the number of iteration increases, *fa* decreases, thus the algorithm converges gradually and can search finely around the optimal solution. The adaptive factor *fa* balances the global and local search capability.

Using adaptive factor alone can't keep good diversity and convergence, as illustrated in Fig. 4. Fast convergence may be harmful to PSO if no action is taken to assist in escaping from a local optima. In PSO, the global best location *gBest* steers the flight direction for a particle, so it would be more effective to modify *gBest* to maintain particles diversity for jumping out of the potential local optima by using a perturbation operator *ga*. However, a higher perturbation rate will lead to a larger loss of fine local search. In order to better manage diversity and convergence, the perturbation range of *gBest* is limited to [0 *max(x)*–*min(x)*]. *max(x)*–*min(x)* denotes the difference of maximum and minimum location of particles in different iteration. At the beginning of the iteration, *max(x)*–*min(x)* is large, and the extensive perturbation range keeps the diversity of particles. Then as the number of iteration increases, *max(x)*–*min(x)* decreases gradually, the lower perturbation rate can converges gradually and search finely around the optimal solution. *gBest* with perturbation operator *ga* is defined as:

$$gBest_j = gBest_j + ga = gBest_j + (\max(x_j) - \min(x_j)) \times rand \quad (15)$$

The pseudo-code of the modified particle swarm optimization is described in Algorithm 1.

Algorithm 1. The modified particle swarm optimization

Input:
Initial parameters: the total iterations *maxIter*; the population size K; the space dimension n; inertial weight w; cognitive parameter c_1; social parameter c_2;
Output:
The global best location **gBest**;
Proceeding:
Generate randomly initial n-dimensional velocity **v** and location **x** of particle swarm;
for $ite = 1 : maxIter$ **do**
  Update the inertial weight w;
  for $pu = 1 : K$ **do**
    Calculate the minimum fitness function of each particle as its individual best location *pBest*;
    Update the velocity and location of each particle by Eqs. (4) and (14);
  end for
  Seek the minimum individual best location as the global best location *gBest*;
  if $ite < maxIter$ **then**
    Perturb the global best location to take new *gBest* as the the global best location by using Eq. (15);
  end if
end for
Return *gBest*.

4.4. The combination of 2D K–L divergence and MPSO

Searching for the optimal solution of MPSO is equivalent to seeking the optimal threshold values in image segmentation. We regard the n-dimensional space of MPSO as the multilevel thresholding space of an image, and the objective function of image segmentation, 2D K–L divergence, is treated as the fitness function of MPSO to seek the optimal solution. The final global best position *gBest* corresponds to the optimal thresholding of an image. The flow chart of the proposed multilevel image thresholding segmentation algorithm is shown in Fig. 3.

5. Experiments

The experimental images come from the Berkeley Segmentation Dataset and Benchmark (BSDS300). The experiments were conducted using Matlab R2014b with 10G memory and a Celeron 1.8 GHz processor. Parametric settings for the swarm size, c_1 and c_2 are 20, 0.7 and 1.43 respectively. The proposed algorithm converges gradually to a constant after 200 iterations according to Fig. 4, so we denote 200 as the total iterations. The number of levels in image segmentation is decided according to practical problems. Segmentation results in this paper are on grayscale images. For the sake of visual comparison, a color map 'jet' from Matlab is used to point out different layers in a better way.

Fig. 4 illustrates the convergence performance of CPSO, APSO and MPSO, which were evaluated on the mean of fitness function over BSDS300 images. We can see in Fig. 4 that MPSO converges

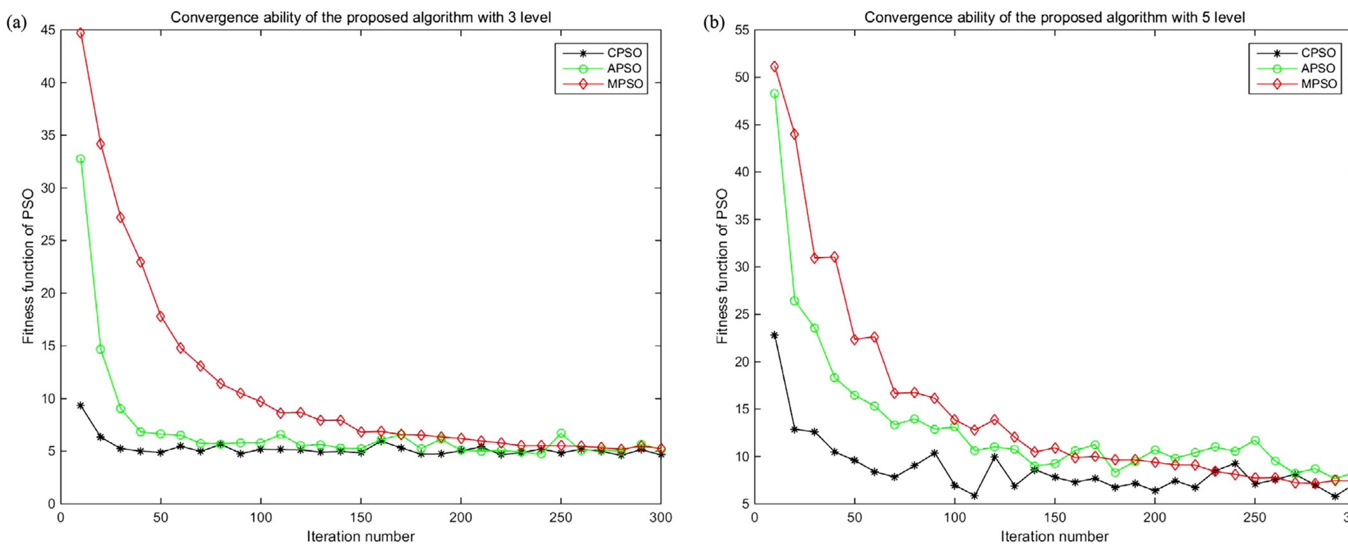

Fig. 4. The convergence performance of the three different PSOs. The red lines describe MPSO, green lines show APSO and black lines represent CPSO. (a) 3-Level segmentation and (b)5-level segmentation. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of the article.)

gradually at the beginning of the iterative process, which indicates that MPSO is seeking for the optimal solution within a large range. This actually helps MPSO to avoid premature convergence and keep better diversity than APSO and CPSO. When the iterations increase, MPSO is able to converge quickly to a constant. APSO offers a higher convergence speed at first, which maybe traps the local optima and can't jump out. With respect to CPSO, it does not seek the solution within a large range, then fluctuates and does not converge to an optimal area. Fig. 4 exhibits that the convergence performance of the proposed MPSO outperforms those of APSO and CPSO.

Fig. 5 exhibits the segmentation results of four images from BSDS with the proposed algorithm (2DKLMPSO), 1D K–L divergence combined MPSO (1DKLMPSO), 1D *Rênyi* entropy combined MPSO (1DRYMPSO) and 2D K–L divergence combined CPSO(2DKLCPSO). As is seen in Fig. 5, as 2D K–L divergence considers spatial correlation, 2DKLMPSO algorithm is capable of better segmenting objects with clearer edges compared with 1DKLMPSO and 1DRYMPSO; e.g., 1DKLMPSO fails to segment leaves and stems in the second image, pyramids in the third image, and trees in the fourth image. 1DRYMPSO does not give clear edges to the head of the horse in the first image, flowers in the second image and pyramids in the third image. 2DKLCPSO is not able to segment the pyramids in the third image because of the premature drawback of CPSO. In conclusion, our proposed algorithm outperforms other three approaches.

Fig. 6 shows the multilevel image segmentation results for another three images with the proposed algorithm 2DKLMPSO,

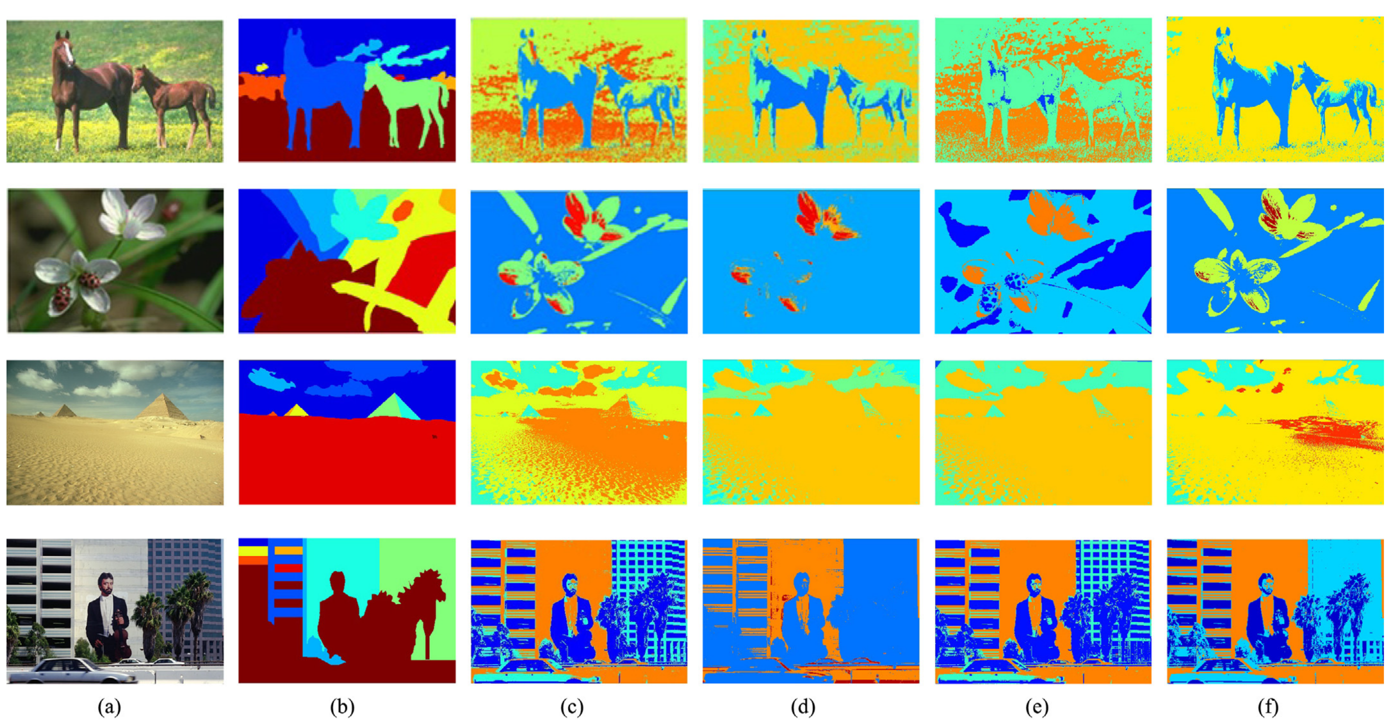

Fig. 5. Comparison of multilevel image segmentation using other entropy-based image segmentation approaches or classical PSO with 3-level segmentation. (a) Original image, (b) human segmentation, (c) 2DKLMPSO, (d) 1DKLMPSO, (e) 1DRYMPSO, (f)2DKLCPSO.

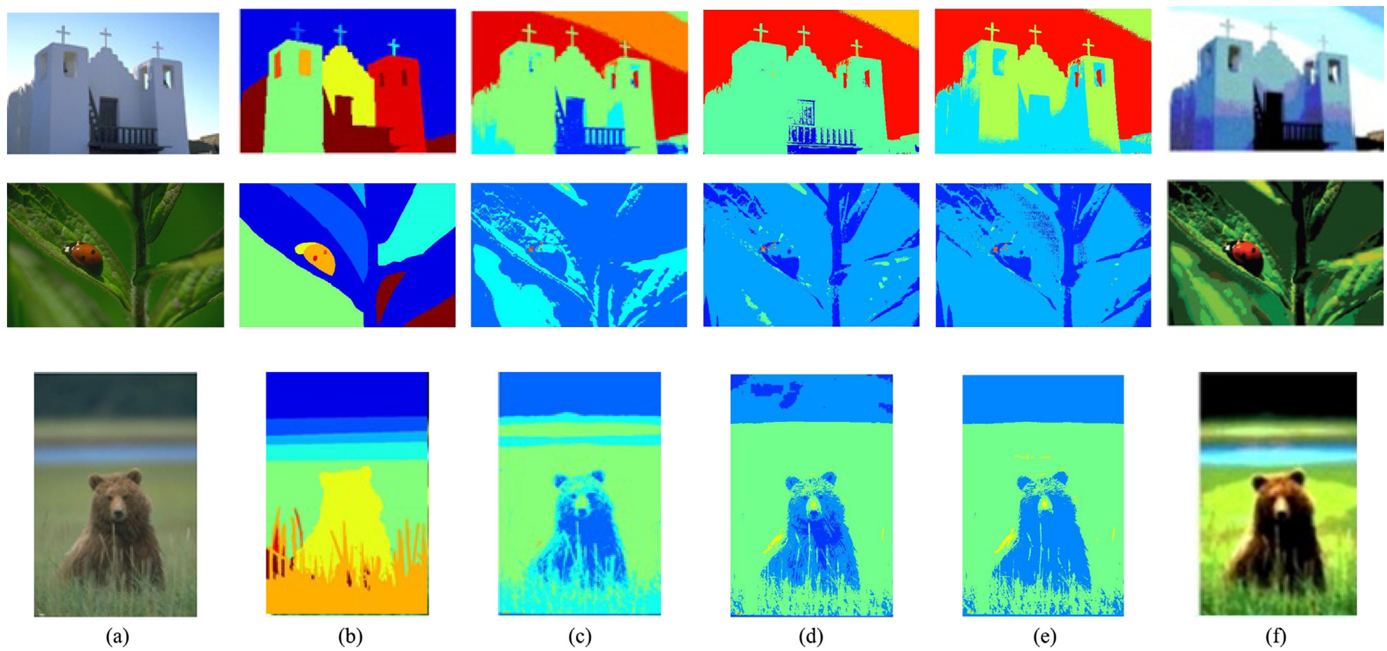

Fig. 6. Comparison of multilevel image segmentation algorithms with 5-level segmentation. (a) Original image, (b) human segmentation, (c) 2DKLMPSO, (d) Ref. [36], (e) Ref. [31], and (f) Ref. [8].

Refs. [36] [31] [8]. The three references make use of DE combined entropy to achieve multilevel image. In order to make fair comparison, Ref. [8] does not include small region mergence. As shown in Fig. 6, our algorithm is much closer to the human segmentation results and achieves clearer edges compared with other three methods, such as balcony in the first image, the contour of beetle in the second image and skyline in the third image.

The above-mentioned Figs. 6 and 5 evaluate subjectively the segmentation results of the proposed algorithm. To assess objectively the proposed algorithm, we also compare our algorithm with other segmentation approaches in terms of four performance indices on 300 images from BSDS dataset. All the images is the size of 481 × 321. For each image, a set of benchmark images compiled by the human observers are provided. All the images are normalized to have the longest side equal to 320 pixels. The four performance indices proposed in [40] are adopted in our experiments, which are:

Boundary displacement error (BDE): This measures the average displacement error of boundary pixels between two segmented images by defining the error of one boundary pixel as the distance between it and the closest pixel in the other boundary image.

Probability rand index (PRI): This counts the fraction of pairs of pixels whose labelings are consistent between the computed segmentation and the ground truth. PRI can be found by averaging the result across all human segmentations (ground truths) of a given image.

Variation of information (VOI): This measures the amount of randomness in one segmentation by calculating the distance between two segmentations as the average conditional entropy of one segmentation given the other.

Global consistency error (GCE): This measures the extent to which one segmentation can be viewed as a refinement of the other. Segmentations which are related in this manner are considered to be consistent, since they could represent the same natural image segmentation at different scales.

The higher the value of PRI, the better the segmentation, whereas for the remaining indices, lower values are better. The four performance indices for each algorithm shown in Tables 1 and 2 are the mean of calculating on 300 test images. Tables 1 and 2 show the four performance indices of image segmentation results with 3-level and 5-level segmentation, respectively, which reveal that the proposed algorithm outperforms the other algorithms in terms of BDE and PRI, and is also comparable to another 2D histogram-based approach [36] in GCE metric. The only shortcoming of the proposed algorithm occurs in VOI index. The possible reason is that the proposed algorithm has no small region merging technique, which will be tested in our future work. Another interesting observation is that although our 2D K–L divergence-based algorithm does not perform well in terms of VOI and GCE, the 1D K–L divergence segmentation method gives almost the best result among all the approaches involved. According to the theorem of "no free lunch" [41], one algorithm cannot offer the better performance than all the others on every aspect or on every kind of problem, so we think

Table 1
Average performance indices of different algorithms over BSDS300 with 3-level segmentation (unit:second).

Algorithm	BDE	PRI	VOI	GCE
Ground Truth	4.9940	0.8754	1.1040	0.0797
2DKLMPSO	**11.5203**	**0.5975**	4.2763	0.4012
1DKLMPSO	13.1636	0.4264	3.8654	**0.2736**
1DRYMPSO	13.4009	0.5814	3.833	0.2748
2DKLCPSO	23.6005	0.4823	3.5123	0.2992
Ref. [8]	13.3576	0.5502	**3.4163**	0.2821
Ref. [31]	11.7327	0.5877	4.6498	0.4152
Ref. [36]	13.0269	0.4863	4.3318	0.3056

The bolds mean the best result among all the approaches.

Table 2
Average performance indices of different algorithms over BSDS300 with 5-level segmentation (unit:second).

Algorithm	BDE	PRI	VOI	GCE
Ground Truth	4.9940	0.8754	1.1040	0.0797
2DKLMPSO	**9.9382**	**0.7154**	5.6802	0.4173
1DKLMPSO	15.2669	0.6128	**2.8948**	**0.3284**
1DRYMPSO	13.5325	0.6825	3.0493	0.3674
2DKLCPSO	11.6437	0.6279	4.6179	0.4108
Ref. [8]	10.5687	0.7153	4.5486	0.3650
Ref. [31]	12.3371	0.5944	5.0030	0.4241
Ref. [36]	10.355	0.6498	4.8931	0.4460

The bolds mean the best result among all the approaches.

Table 3
Time complexity of different sized images with different levels (unit: second).

Image size	2-level	3-level	4-level	5-level
537 × 358	1.263	1.306	1.391	1.436
321 × 481	1.226	1.299	1.362	1.414
256 × 256	1.207	1.239	1.314	1.398

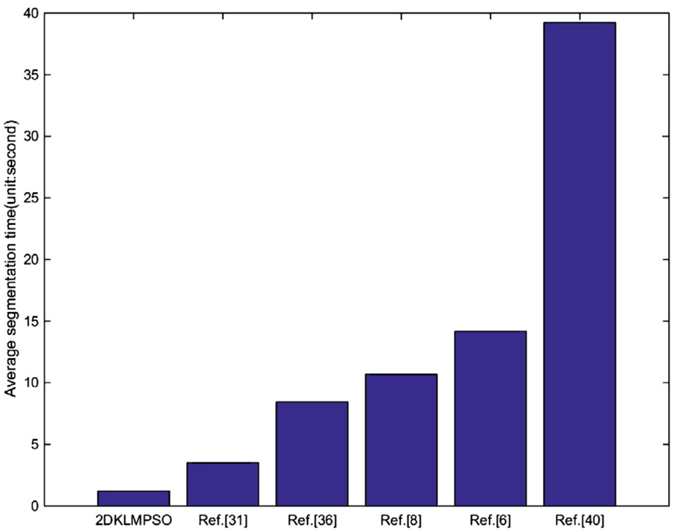

Fig. 7. Average computational cost of different algorithms.

that the proposed algorithm is an effective tool in multilevel image segmentation.

Table 3 shows the running time of different sized images with 2-level, 3-level, 4-level, and 5-level segmentation using the proposed algorithm, and reveals the relationship between the time complexity of our algorithm and the number of levels in multilevel segmentation. As is noticeable in Table 3, for a certain image, the computational time increases very little with the increase of levels. The major reason is that the n-dimensional space of MPSO is treated as multilevel thresholding space of the image in the proposed algorithm. The parallel capability of MPSO leads to parallel computing of multilevel image thresholding, not exhaustive search.

In order to evaluate the time complexity of image segmentation, we compare the processing time of metaheuristic algorithms (2DKLMPSO, Refs. [31,36,8]) and non-metaheuristic approaches(Refs. [6,40]). Exhaustive search [6] is a traditional image thresholding segmentation approach, which searches exhaustively the optimal multilevel thresholds and thus is quite costly in time, so we only compare the time consumption of 2-level image segmentation. The result of our algorithm shown in Fig. 7 is also the running time of 2-level image segmentation. It can be seen from Fig. 7 that our approach has the lowest computational cost. The likely reason is that MPSO has no mutation and crossover compared with DE. Although the time consuming of Ref. [6] with 2-level is smaller than that of Ref. [40], the time cost of the former is larger than the latter with 3 level or above. Fig. 7 shows that the proposed algorithm efficiently reduces the computational complexity of multilevel image thresholding segmentation.

6. Conclusion

In this paper, we propose a multilevel image thresholding segmentation algorithm based on 2D K–L divergence and MPSO. As 2D K–L divergence considers the spatial correlation between pixels, the proposed algorithm achieves clearer edges in segmentation. We also show experimentally that K–L divergence can be used as an effective tool for seeking the optimal thresholds of multilevel image segmentation. The MPSO that adopts adaptive factor and perturbation operator conquers the drawback of premature convergence of PSO, and its parallelism efficiently decreases the time complexity of multilevel image thresholding segmentation. The subjective and objective assessments conducted on BSDS300 show that the proposed algorithm is effective and robust. The limitation of our proposed algorithm is that we assign the number of segmentations in advance. In the future, We hope to select adaptively the number of levels according to the characteristics of an image.

References

[1] Y. Tian, J. Li, S. Yu, T. Huang, Learning complementary saliency priors for foreground object segmentation in complex scenes, Int. J. Comput. Vis. 111 (2) (2014) 153–170.
[2] F. Galasso, M. Keuper, T. Brox, B. Schiele, Spectral graph reduction for efficient image and streaming video segmentation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2014, pp. 49–56.
[3] P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011) 898–916.
[4] A. Dirami, K. Hammouche, M. Diaf, P. Siarry, Fast multilevel thresholding for image segmentation through a multiphase level set method, Signal Process. 93 (1) (2013) 139–153.
[5] X. Zhang, C. Xu, M. Li, X. Sun, Sparse and low-rank coupling image segmentation model via nonconvex regularization, Int. J. Pattern Recogn. Artif. Intell. 29 (02) (2015) 1555004.
[6] N. Otsu, A threshold selection method from gray-level histograms, Automatica 11 (285-296) (1975) 23–27.
[7] J.N. Kapur, P.K. Sahoo, A.K. Wong, A new method for gray-level picture thresholding using the entropy of the histogram, Comput. Vis. Graph. Image Process. 29 (3) (1985) 273–285.
[8] 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.
[9] J. Kennedy, Particle swarm optimization, in: Encyclopedia of Machine Learning, Springer, 2010, pp. 760–766.
[10] M. Hu, T.-F. Wu, J.D. Weir, An adaptive particle swarm optimization with multiple adaptive methods, IEEE Trans. Evolut. Comput. 17 (5) (2013) 705–720.
[11] Y. Liu, C. Mu, W. Kou, J. Liu, Modified particle swarm optimization-based multilevel thresholding for image segmentation, Soft Comput. 19 (5) (2014) 1311–1327.
[12] B. Akay, A study on particle swarm optimization and artificial bee colony algorithms for multilevel thresholding, Appl. Soft Comput. 13 (6) (2013) 3066–3091.
[13] M.P. de Albuquerque, I. Esquef, A.G. Mello, Image thresholding using Tsallis entropy, Pattern Recogn. Lett. 25 (9) (2004) 1059–1065.
[14] P.K. Sahoo, G. Arora, A thresholding method based on two-dimensional Renyi's entropy, Pattern Recogn. 37 (6) (2004) 1149–1161.
[15] A.S. Abutaleb, Automatic thresholding of gray-level pictures using two-dimensional entropy, Comput. Vis. Graph. Image Process. 47 (1) (1989) 22–32.
[16] S. Kullback, Information Theory and Statistics, Courier Corporation, 1968.
[17] A. Rényi, A diary on information theory, AMC 10 (1987) 12.
[18] P.-Y. Yin, L.-H. Chen, A fast iterative scheme for multilevel thresholding methods, Signal Process. 60 (3) (1997) 305–313.
[19] L. Dong, G. Yu, An efficient iterative optimization algorithm for image thresholding, in: Computational and Information Science, Springer, 2005, pp. 1079–1085.
[20] M. Sezgin, R. Taşaltın, A new dichotomization technique to multilevel thresholding devoted to inspection applications, Pattern Recogn. Lett. 21 (2) (2000) 151–161.
[21] J. Zhang, H. Li, Z. Tang, Q. Lu, X. Zheng, J. Zhou, An improved quantum-inspired genetic algorithm for image multilevel thresholding segmentation, Math. Prob. Eng. (2014) 12.
[22] J. Kennedy, J.F. Kennedy, R.C. Eberhart, Y. Shi, Swarm Intelligence, Morgan Kaufmann, 2001.
[23] Y. Li, L. Jiao, R. Shang, R. Stolkin, Dynamic-context cooperative quantum-behaved particle swarm optimization based on multilevel thresholding applied to medical image segmentation, Inf. Sci. 294 (2015) 408–422.
[24] A. Fu, X. Lei, Image segmentation based on two-dimensional histogram and the geese particle swarm optimization algorithm, in: 7th World Congress on Intelligent Control and Automation. WCICA 2008, IEEE, 2008, pp. 7045–7048.
[25] N. Chen, W.-N. Chen, J. Zhang, Fast detection of human using differential evolution, Signal Process. 110 (2015) 155–163.
[26] P. Ghamisi, J.A. Benediktsson, Feature selection based on hybridization of genetic algorithm and particle swarm optimization, IEEE Geosci. Remote Sens. Lett. 12 (2) (2015) 309–313.

[27] H. Wu, C. Nie, F.C. Kuo, H. Leung, C.J. Colbourn, A discrete particle swarm optimization for covering array generation, IEEE Trans. Evolut. Comput. 19 (4) (2015) 575–591.
[28] S. Dey, I. Saha, S. Bhattacharyya, U. Maulik, Multi-level thresholding using quantum inspired meta-heuristics, Knowl. Based Syst. 67 (2014) 373–400.
[29] M. Hu, T. Wu, J.D. Weir, An intelligent augmentation of particle swarm optimization with multiple adaptive methods, Inf. Sci. 213 (2012) 68–83.
[30] A.K. Bhandari, V.K. Singh, A. Kumar, G.K. Singh, Cuckoo search algorithm and wind driven optimization based study of satellite image segmentation for multilevel thresholding using kapur's entropy, Expert Syst. Appl. 41 (7) (2014) 3538–3560.
[31] T. Kurban, P. Civicioglu, R. Kurban, E. Besdok, Comparison of evolutionary and swarm based computational techniques for multilevel color image thresholding, Appl. Soft Comput. 23 (2014) 128–143.
[32] S. Patra, R. Gautam, A. Singla, A novel context sensitive multilevel thresholding for image segmentation, Appl. Soft Comput. 23 (2014) 122–127.
[33] S. Yin, X. Zhao, W. Wang, M. Gong, Efficient multilevel image segmentation through fuzzy entropy maximization and graph cut optimization, Pattern Recogn. 47 (9) (2014) 2894–2907.
[34] A. Bhandari, A. Kumar, G. Singh, Modified artificial bee colony based computationally efficient multilevel thresholding for satellite image segmentation using Kapur's, Otsu and Tsallis functions, Expert Syst. Appl. 42 (3) (2015) 1573–1601.
[35] C. Qi, Maximum entropy for image segmentation based on an adaptive particle swarm optimization, Appl. Math 8 (6) (2014) 3129–3135.
[36] S. Sarkar, S. Das, Multilevel image thresholding based on 2d histogram and maximum Tsallis entropya differential evolution approach, IEEE Trans. Image Process. 22 (12) (2013) 4788–4797.
[37] F. Nie, Tsallis cross-entropy based framework for image segmentation with histogram thresholding, J. Electron. Imaging 24 (1) (2015) 013002.
[38] S. Fehr, S. Berens, On the conditional Rényi entropy, IEEE Trans. Inf. Theory 60 (11) (2014) 6801–6810.
[39] W. Wang, Research on particle swarm optimization and its application, 2012.
[40] A.Y. Yang, J. Wright, Y. Ma, S.S. Sastry, Unsupervised segmentation of natural images via lossy data compression, Comput. Vis. Image Understand. 110 (2) (2008) 212–225.
[41] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, IEEE Trans. Evolut. Comput. 1 (1) (1997) 67–82.