

## Research Article

# Subpixel Mapping Method of Hyperspectral Images Based on Modified Binary Quantum Particle Swarm Optimization

Shuhan Chen,<sup>1</sup> Xiaorun Li,<sup>1</sup> and Liaoying Zhao<sup>2</sup>

<sup>1</sup>College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China

<sup>2</sup>Institute of Computer Application Technology, Hangzhou Dianzi University, Hangzhou 310018, China

Correspondence should be addressed to Xiaorun Li; [lxrly@zju.edu.cn](mailto:lxrly@zju.edu.cn)

Received 16 March 2017; Revised 31 May 2017; Accepted 3 July 2017; Published 6 August 2017

Academic Editor: Qunming Wang

Copyright © 2017 Shuhan Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Subpixel mapping technology can determine the specific location of different objects in the mixed pixel and effectively solve the uncertainty of the ground features spatial distribution in traditional classification technology. Existing methods based on linear optimization encounter the premature and local convergence of the optimization algorithm. This paper proposes a subpixel mapping method based on modified binary quantum particle swarm optimization (MBQPSO) to solve the above issues. The initial subpixel mapping imagery is obtained according to spectral unmixing results. We focus mainly on the discretization of QPSO, which is implemented by modifying the discrete update process of particle location, to minimize the objective function, which is formulated based on different connected regional perimeter calculating methods. To reduce time complexity, a target optimization strategy of global iteration combined with local iteration is performed. The MBQPSO is tested on standard test functions and results show that MBQPSO has the best performance on global optimization and convergent rate. Then, we analyze the proposed algorithm qualitatively and quantitatively by simulated and real experiment; results show that the method combined with MBQPSO and objective function, which is formulated based on the gap length between region and background, has the best performance in accuracy and efficiency.

## 1. Introduction

Hyperspectral images are composed of hundreds of bands with a very high spectral resolution, generally from the visible to the infrared region. For every recorded pixel, rich spectral information provides a complete spectral description and a better characterization of the observed surface, which results in a very powerful tool for materials discrimination and earth observation. However, a common drawback of hyperspectral sensors is the relatively low spatial resolution, which leads to the problem of mixed pixels (e.g., pixels containing mixture of different materials). Ground feature spatial detail in mixed pixels is extremely important for land cover mapping, coast-line extraction, change detection, and landscape index estimating. Mixed pixels cannot be correctly addressed by traditional hard classification methods. Subpixel mapping is an effective measure to solve this problem.

Subpixel mapping and pixels spatial dependence theory were initially proposed by Atkinson [1]. Based on this theory,

several subpixel mapping techniques have been proposed. Existing subpixel mapping methods contain the following typical algorithms: the neural network method, the geostatistics method, the Markov random field method, and the linear optimization method.

The first category is the neural network method. Among them, Hopfield's neural network (HNN) model combined with the constrained energy minimization principle of output neuron has been used in the subpixel mapping. In the HNN model, each pixel is a neuron that constructs an energy function with the correlation between subpixels and the neighboring subpixel, using the mixed pixels abundance as the constraint [2, 3]. The improved HNN method for small scale targets and small scale coexisting with large scale target had been proposed in [4]. A Back Propagation (BP) neural network is trained to learn the appropriate location of the subpixels, which belongs to the different classes inside the pixel [5]. At the same time, wavelet transform and genetic algorithms have been used to solve subpixel mapping

[6, 7]. The ARTMAP neural network [8] has been utilized to realize the learning model in [6]. Based on the subpixel shifted remote sensing images (SSRSI), component information obtained from SSRSI was put into the HNN model to increase ratio constraints and reduce uncertainty [9]. In order to eliminate isolated pixels, the results of the BP neural network model were postprocessed [10]. Multiple low spatial resolution subpixel shifted images were used to modify the BP neural network to reduce uncertainty and error in the BPNN model [11]. The general regression neural network (GRNN) has been proposed to achieve improved accuracy in subpixel mapping [12].

The second category is the geostatistics method. Existing approaches for subpixel mapping have been placed within an inverse problem framework; a geostatistical method was proposed for generating alternative synthetic land cover maps at the fine (target) spatial resolution [13, 14]. In [15], a model sequentially produced with local indicator variogram (SLIV) was proposed, and indicator variograms extracted from target-resolution classification were produced from a representative local area to develop an effective method.

The third category is the Markov random field method. The Markov random field (MRF) has been used in subpixel mapping [16]. Subsequently, scholars have done in-depth study on the MRF model [17, 18]. Considering the idea of the MRF, which is able to consider spatial and spectral information simultaneously, a novel sFCM-based (supervised fuzzy  $c$ -means) subpixel mapping model that incorporates the sFCM criterion for unmixing pixels into the objective function was proposed [19]. In terms of the energy minimization for MRF, one of the most commonly used traditional methods is simulated annealing (SA), but SA is very time-consuming. To overcome this limitation, graph cut is used in [20].

The fourth category is the linear optimization method. Through the decomposition of mixed pixels or fuzzy classification technique, the proportion of features was determined and then a mathematical model was defined for describing the spatial correlation to construct the objective function, converting the subpixel mapping into a linear optimization problem. Among linear optimization method, Verhoeve proposed the maximum spatial correlation between neighborhood pixels and solved the problem using simplex linear optimization technology [21]. Immune clonal selection algorithms and differential evolution algorithms were used to solve the mathematical model using Verhoeve's data [22, 23]. A new subpixel mapping method based on evolution agent was proposed [24]. Since subpixel/pixel spatial attraction model (SPSAM), which assumes that similar features have physical characteristics of mutual attraction, has effectively verified spatial correlation theory [25, 26], SPSAM ignores the correlation between subpixels. Wang put forward a modified SPSAM, which considers the correlation between and within the pixel simultaneously [27, 28]. By incorporating auxiliary datasets, a subpixel mapping framework based on a maximum a posteriori (MAP) model was proposed to utilize the complementary information of multiple shifted images [29]. Since the previous MAP-based subpixel mapping algorithm was obtained by downsampling a classification image without spectral unmixing errors, an adaptive subpixel

mapping method based on a MAP model and a winner-take-all class determination strategy (AMCDSM) was proposed [30]. The existing spectral-spatial based SPM algorithms only use the maximal spatial dependence principle as the spatial term to describe the local spatial distribution of different land cover features; a novel spectral-spatial based SPM algorithm with multiscale spatial dependence was proposed [31]. In [32], the spectral unmixing predictions (i.e., coarse land cover proportions used as input for SPM) were considered a convolution of not only subpixels within the coarse pixel, but also subpixels from neighboring coarse pixels. A new SPM method based on optimization is developed which recognizes the optimal solution as the one that, when convolved with the PSF, is the same as the input coarse land cover proportion.

The computation complexity of those above objective functions is relatively highly complex. In order to intuitively reflect the spatial correlation, based on the principle that similar features attract each other, Villa utilized various types of regional perimeter minimum as a target after subpixel mapping, using the simulated annealing (SA) and pixel swapping algorithm (PSA) to achieve iterative optimization [33, 34]. Compared with SA and other similar evolutionary techniques, PSO has some attractive characteristics and has proven to have superior computational efficiency. This is because PSO is not applicable to problems where the position of the particle should be discrete [35]. Ertürk used binary particle swarm optimization (BPSO) to optimize the objective function that had been proposed by Villa, and the optimization algorithm has been fit for parallel implementation [35]. The results of the BPSO algorithm are better than SA. Compared with BPSO algorithm, Sun proposed the binary quantum particle swarm optimization algorithm (QPSO) [36], which demonstrates many advantages such as a simple evolution equation, fewer control parameters, fast convergence rate, and simple operation. Therefore, BQPSO algorithm was proposed for subpixel mapping [37], but the discrete update process of particle location may lead to algorithm premature convergence and local convergence. The modified version of BQPSO was proposed in this article.

In this study, a subpixel mapping method of hyperspectral images based on MBQPSO is proposed. The initial subpixel mapping imagery is obtained according to the results of spectral unmixing. Using regional perimeter minimizing to depict spatial correlation is intuitive and requires low computational complexity. The objective function is formulated by different connected regional perimeter calculating methods, such as the gap length between region and background ( $f_{\text{gap}}$ ), the length of chain code ( $f_{\text{chain}}$ ), and the sum of border point number ( $f_{\text{point}}$ ). We focus mainly on the discretization of QPSO, which is implemented by modifying the discrete update process of particle location, to minimize the above three objective functions. In order to reduce time complexity, objective optimization strategy of global iteration combined with local iteration is performed.

The rest of this paper is organized as follows. Section 2 presents the basic methodology of the proposed subpixel mapping method based on MBQPSO. Section 3 provides the experimental results and analyses with simulated and real hyperspectral images. The conclusion is drawn in Section 4.

## 2. Subpixel Mapping Method Based on MBQPSO

The proposed method provides an accurate subpixel mapping method based on MBQPSO and local optimization strategy. In this paper, known endmembers and abundance are the premise of the subpixel mapping method. Specifically, the number of endmembers is estimated by VD [38] and extracted by VCA. Then, the abundance is obtained by FCLS in each pixel for each endmember. A pixel can be seen as a pure pixel only if the minimum difference value between the maximum abundance and the others in the pixel is greater than a certain threshold [33–35]. The initial subpixel mapping imagery is obtained according to spectral unmixing results.

Then the follow-up process of the proposed subpixel mapping algorithm based on MBQPSO consists of the following: (1) the objective function is formulated by different connected regional perimeter calculating methods; (2) the proposed MBQPSO is used to minimize the objective function; (3) optimization strategy of global iteration combined with local iteration is performed to reduce time complexity. The framework of the proposed subpixel mapping method is organized as follows: The objective function and adaptability analysis are presented in Section 2.1. The MBQPSO algorithm is introduced in Section 2.2. Two key problems of objective function optimization based on MBQPSO are described in Section 2.3 followed by the optimization strategy in Section 2.4.

**2.1. Modified Binary Quantum Particle Swarm Optimization Algorithm.** Before presenting the proposed MBQPSO, we first introduce QPSO method, containing the basic concept and evolution equation. The discrete update process of evolution equation is given in the form of pseudocode, and the specific details of the discrete update process of  $X_{ij}^{t+1}$  are given in the form of textual description.

**2.1.1. QPSO Method.** The QPSO algorithm [36] is theoretical a global convergence guaranteed algorithm, which has fewer parameters and better global searching ability than PSO. In QPSO algorithm, there are only the concepts of particle position and the distance between particles, while PSO includes the concept of speed and trajectory. In a QPSO, each particle only has position information  $X_i^t = (X_{i1}^t, X_{i2}^t, \dots, X_{id}^t)$  in a  $d$ -dimensional searching space.  $t$  is the current time and  $i$  is the  $i$ th particle in  $M$  particles. While time is  $t$ , the optimal position of the  $i$ th particle is  $P_i^t = (P_{i1}^t, P_{i2}^t, \dots, P_{id}^t)$  and the global optimal position of particle swarm is  $P_g^t = (P_{g1}^t, P_{g2}^t, \dots, P_{gd}^t)$ . The evolution equations are as follows:

$$mbest^t = \sum_{i=1}^M \frac{P_i^t}{M} \quad (1)$$

$$p_{ij}^t = \theta_j^t \cdot P_{ij}^t + [1 - \theta_j^t] \cdot P_{gj}^t \quad (2)$$

$$X_{ij}^{t+1} = p_{ij}^t \pm \beta \cdot |mbest_j^t - X_{ij}^t| \cdot \ln\left(\frac{1}{u_{ij}^t}\right), \quad (3)$$

```

Input:  $P_{ij}^t, P_{gj}^t$ 
Output:  $p_{ij}^t$ 
 $r = \text{rand}()$ 
 $p_{ij}^t = rP_{ij}^t + (1 - r)P_{gj}^t$ 
If  $p_{ij}^t > 0.5$  then  $p_{ij}^t = 1$ 
else  $p_{ij}^t = 0$ 
end

```

ALGORITHM 1: Get\_ $P$ :  $p_{ij}^t$  discretization.

```

Input:  $P_i^t$  ( $1 \leq i \leq n$ )
Output:  $mbest^t$ 
 $\bar{P} = \frac{1}{n} \sum_{i=1}^n P_i^t$ 
for each bit in  $\bar{P}$ 
if  $\bar{P}(i) > 0.5$  then  $mbest^t(i) = 1$ 
else if  $\bar{P}(i) < 0.5$  then  $mbest^t(i) = 0$ 
else if  $\text{rand}() < 0.5$  then  $mbest^t(i) = 0$ 
else  $mbest^t(i) = 1$ 
end

```

ALGORITHM 2: Get\_ $mbest$ :  $mbest^t$  discretization.

where  $j$  is a positive integer between 1 and  $d$ ;  $u_{ij}^t$  and  $\theta_j$  are random positive numbers between 0 and 1. If  $u_{ij}^t \geq 0.5$ , then the plus/minus sign in (3) takes a plus sign, or else it takes a minus [36].  $p_i^t$  is the  $i$ th particle attractor, represented by a random vector that is defined by (2).  $mbest^t$  is the average optimal position;  $\beta$  is contraction-expansion coefficient and determines the particle convergence speed. The expression is as follows:

$$\beta = (\alpha_0 - \alpha_1) \times \frac{\text{maxiteration} - \text{curiteration}}{\text{maxiteration}} + \alpha_0, \quad (4)$$

where curiteration is the current iteration number and maxiteration is the maximum iteration number.  $\alpha_0$  and  $\alpha_1$  are the initial and final value of the control parameter, respectively. In addition,  $\alpha_0$  is greater than  $\alpha_1$ , and  $\beta$  is a linear decreasing function, controlled by the parameters.

However, like PSO and other evolutionary algorithms, QPSO readily relapses into local optimum when solving high-dimensional complex optimization problems. In order to adapt the QPSO algorithm to the practical problems in discrete search space, we introduce the concept of binary encoding to QPSO, which is implemented by modifying the discrete update process of particle location.

**2.1.2. MBQPSO Method.** In BQPSO [37], particle position is represented as a binary string, and continuous evolution equations (1)–(3) are all discretized. The discrete update process of  $mbest^t$  and  $p_i^t$  is given in Algorithms 1 and 2.

```

Input:  $X_{ij}^t, p_{ij}^t, mbest^t$ 
Output:  $X_{ij}^{t+1}$ 
Calculate  $l_{ij}, b, p_r$ 
 $X_{ij}^{t+1} = p_{ij}^t$ 
if  $b \neq 0$ 
begin
  For each bit of  $X_{ij}^{t+1}$ , denote as  $X_{ij}^{t+1}(k), k > 1$ 
  if  $X_{ij}^{t+1} > 0$ 
    if  $p_r < \text{rand}()$  then  $X_{ij}^{t+1}(k) = 1$ 
    else  $X_{ij}^{t+1}(k) = 0$ 
  else
    if  $p_r < \text{rand}()$  then  $X_{ij}^{t+1}(k) = 0$ 
    else  $X_{ij}^{t+1}(k) = 1$ 
  end
end
end

```

ALGORITHM 3:  $\text{Trans}_X: X_{ij}^{t+1}$  discretization.

In order to discretize the particle updated location, deformation of (3) is as follows:

$$d_H(X_{ij}^{t+1}, p_{ij}^t) = [b]$$

$$b = \beta \cdot d_H(mbest_j^t, X_{ij}^t) \cdot \ln\left(\frac{1}{u_{ij}^t}\right), \quad (5)$$

where  $d_H(\cdot)$  represents the function of Hamming distance and  $[\cdot]$  represents integers. It should be noted that  $b \geq 0$  due to  $u_{ij}^t$  is random positive number between 0 and 1 and  $d_H(\cdot) \geq 0$

Substituting  $b$  and  $p_{ij}^t$  into (5),  $X_{ij}^{t+1}$  can be obtained in reverse. The time complexity of this procedure is  $O(l_{ij}b)$ . In order to reduce calculation amount, each bit of  $p_{ij}^t$  is mutated with mutation probability  $p_r$  to obtain a new  $X_{ij}^{t+1}$ :

$$p_r = \begin{cases} \frac{b}{l_{ij}}; \\ 1, & \text{if } \frac{b}{l_{ij}} > 1, \end{cases} \quad (6)$$

where  $l_{ij}$  is the length of  $X_{ij}^t$ .

It should be noted that “ $\pm$ ” in (3) is not considered in (5), which may result in premature convergence. In order to overcome this problem, the discretization procedure of  $X_{ij}^{t+1}$  is modified and called  $\text{Trans}_X$ , which is shown in Algorithm 3. The time complexity of  $\text{Trans}_X$  is  $O(l_{ij})$ . The modified algorithm is named MBQPSO to distinguish it from the BQPSO algorithm.

The algorithm's design work is described as follows.

First, set  $X_{ij}^{t+1} = p_{ij}^t$ ; then use mutation probability  $p_r$  to update  $X_{ij}^{t+1}$ . For  $X_{ij}^{t+1}$ , let  $n10$  be the number of bits, which changes from 1 to 0 and  $n01$  be the number of bits, which changes from 0 to 1. The detailed update process is divided into three cases.

(1) if  $b = 0$ , then  $X_{ij}^{t+1} = p_{ij}^t$ .

(2) According to the selection of plus or minus sign in (3), we set the following rule:

(2.1) If  $u_{ij}^t \geq 0.5$ , then  $X_{ij}^{t+1} = p_{ij}^t + [b]$ ; this may result in  $X_{ij}^{t+1} > p_{ij}^t$  due to  $b > 0$ . It is obvious that the bigger  $b$ , the bigger the difference of  $n01$  and  $n10$ .

(a) For the case of  $X_{ij}^{t+1} > 0$ , in order to obtain updated  $X_{ij}^{t+1} > p_{ij}^t$ , the result of mutation should satisfy  $n01 > n10$ . According to the definition of  $p_r$ , the bigger  $b$ , the smaller  $p_r$  and the more the chance for  $p_r < \text{rand}()$ ; that is,  $n01 > n10$  will be satisfied if set  $X_{ij}^{t+1}(k)=1$  when  $p_r < \text{rand}()$  and otherwise set  $X_{ij}^{t+1}(k) = 0$ .

(b) For the case of  $X_{ij}^{t+1} < 0$ , in order to obtain a new  $X_{ij}^{t+1} > p_{ij}^t$ , the result of mutation should satisfy  $n10 > n01$ , which will be satisfied if set  $X_{ij}^{t+1}(k) = 0$  when  $p_r < \text{rand}()$  and otherwise set  $X_{ij}^{t+1}(k) = 1$ .

(2.2) If  $u_{ij}^t < 0.5$ , then  $X_{ij}^{t+1} = p_{ij}^t - [b]$ ; this may result in  $X_{ij}^{t+1} < p_{ij}^t$  due to  $b > 0$ . In the same way, the same conclusion can be obtained:

(a) For the case of  $X_{ij}^{t+1} > 0$ , set  $X_{ij}^{t+1}(k) = 0$  when  $p_r < \text{rand}()$  and else set  $X_{ij}^{t+1}(k) = 1$ .

(b) For the case of  $X_{ij}^{t+1} < 0$ , set  $X_{ij}^{t+1}(k) = 1$  when  $p_r < \text{rand}()$  and otherwise set  $X_{ij}^{t+1}(k) = 0$ .

One final comment is noteworthy. If  $X_{ij}^{t+1}$  is unsigned, a new binary string  $[0, X_{ij}^{t+1}]$  will be constructed to complete the update of  $X_{ij}^{t+1}$  using the function  $\text{Trans}_X$ .

Using function analysis theory, the convergence of the BQPSO algorithm is proven [37]. Compared to BQPSO, the proposed MBQPSO algorithm simply modifies the discretization process of particle position. Therefore, it is easy to prove the convergence of the modified algorithm using the same method.

## 2.2. Objective Function and Adaptability Analysis

2.2.1. *Objective Function.* Relying on the spatial correlation tendency of features, we assume that each endmember within a pixel should be spatially close to the same endmembers in the surrounding pixels. Therefore, the cost function  $C$  to be minimized is chosen as the perimeter of the areas belonging to the same class:

$$C = \sum_{n=1}^{C_M} \sum_{m=1}^{C_n} P_i, \quad (7)$$

where  $C_M$  is the number of the classes,  $C_i$  is the number of connected components of the class  $n$ , and  $P_{ij}$  is the perimeter of the connected component  $j$ .

TABLE 1: The perimeter of Figure 1.

Method	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
$f\_gap$	42	86	48	68	58	108	56	64
$f\_chain$ (4-connected)	26	17.31	30.73	24.38	40.24	33.07	40.49	39.90
$f\_chain$ (8-connected)	26	45.6	30.73	24.38	40.24	67.01	40.49	39.90
$f\_point$	26	27	25	25	34	36	34	34

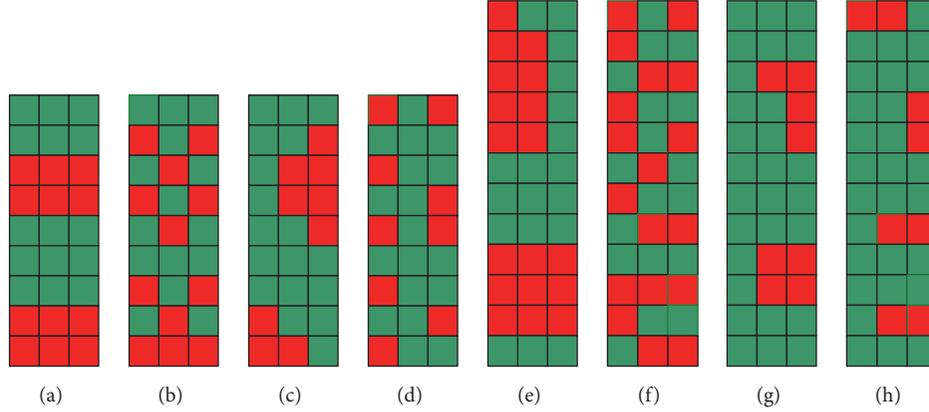


FIGURE 1: Special cases subpixel distribution.

In general, the perimeter calculation of the connected component has three methods: the gap length between region and background, the length of chain code (4-connected and 8-connected), and the sum of border points number [39]. For convenience, the objective functions corresponding to these three methods are named  $f\_gap$ ,  $f\_chain$ , and  $f\_point$ .

**2.2.2. Adaptability Analysis of Perimeter Calculation Methods.** Adaptability analysis of the three perimeter calculation methods ( $f\_gap$ ,  $f\_chain$ , and  $f\_point$ ) are shown in Figure 1. Table 1 gives the perimeters that are calculated by the different methods in Figure 1. The real features' subpixel distribution maps are Figures 1(a), 1(c), 1(e), and 1(g), respectively. Based on the same subpixel abundance, Figures 1(b), 1(d), 1(f), and 1(h) are randomly selected subpixel mapping results with special case.

**Special Case 1.** There may be only one subpixel in a connected area (isolated single point). The perimeter of an isolated single point is 0. The total perimeter, including an isolated point and a certain region, is bound to be bigger than the region's perimeter, as shown in Figures 1(b) and 1(d).

**Special Case 2.** There may be only two subpixels in a connected area (isolated two points), as shown in Figures 1(f) and 1(h).

From Figure 1, it can be seen that the feature original pixel distributions (a), (c), (e), and (g) are better than the results of the special case distributions (b), (d), (f), and (h).

Combination of Figure 1 and Table 1 shows the following:

(1)  $f\_gap$  method calculates connected components perimeter; isolated points or isolated two points will increase

the perimeter length. This method satisfies the requirement that the smaller the value of objective function, the better the subpixel mapping result.

(2) For cases (b) and (f), the number of boundary points is increased with the  $f\_point$  method, but for (d) and (h), the mapping results could not acquire (c) and (g), even with the same objective function values.

(3) The  $f\_chain$  (8-connected) method has an uncertain trend of changing calculation results. For cases (b) and (f), the smaller the target function is, the better the result is. Because cases (d) and (h) have isolated single points, for which 8-connected chain code perimeter is 0, this method cannot obtain mapping results of (c) and (g).

(4) For Figure 1, the  $f\_chain$  (4-connected) method cannot acquire the optimal results while the value of the objective function gets smaller.

Therefore, according to the analysis of these special cases,  $f\_gap$  is the best,  $f\_point$  takes second place, and  $f\_chain$  is the worst. While using chain code method to calculate perimeter, 4-connected segmentation leads to the generation of isolated single points. However, 8-connected segmentation may not necessarily generate an isolated point. So the latter's performance is better than the former. Therefore,  $f\_chain$  (8-connected) is used in the following experiments.

**2.3. Objective Function Optimization Based on MBQPSO.** In this paper, we use different connected regional perimeter to formulate the objective function and the proposed method is used to minimize the aforementioned objective function. Based on the MBQPSO algorithm, there are two problems that should be solved to optimize objective function (7). They are as follows.

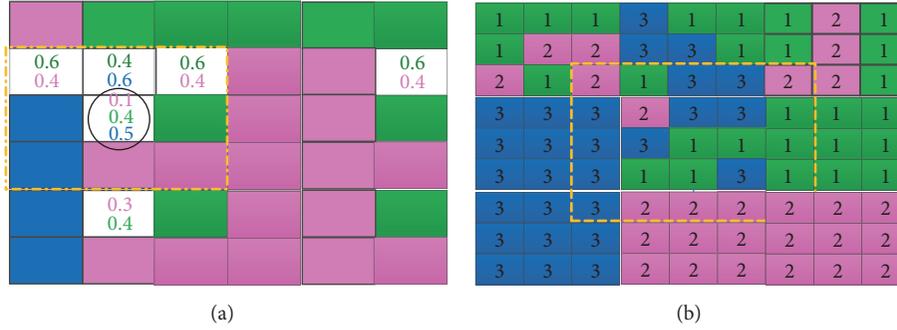


FIGURE 2: Pixel-by-pixel iterative optimization region: (a) original image; (b) spatial resolution enhancement image.

(a) *Corresponding Relations between Particle and Pixel.* After the endmembers are extracted and the abundances in each pixel for each endmember are determined, an abundance threshold value is used to determine whether each pixel may be considered pure. The remaining pixels are considered mixed. Then, for each mixed pixel, establish that each pixel corresponds to a subpixel window, which is composed of  $r^2$  subpixel. Supposing the  $i$ th pixel contains  $d$  pieces features, the abundance values  $\{s_{i1}, s_{i2}, \dots, s_{id}\}$  are ordered from low to high in accordance with  $|s_{ik} \times r^2 - \text{round}(s_{ik} \times r^2)|$ . The number of subpixels in the corresponding subpixel window is

$$n_{ik} = \text{round}(s_{ik} \times r^2), \quad k = 1, 2, \dots, d - 1$$

$$n_{id} = r^2 - \sum_{k=1}^{d-1} n_{ik}, \quad (8)$$

where  $d$  is the number of feature classes,  $i$  is the  $i$ th pixel, and  $k$  is the  $k$ th class feature.  $n_{ik}$  is the number of subpixels assigned to the  $k$ th endmember in the  $i$ th pixel and  $s_{ik}$  is the abundance of subpixels assigned to the  $k$ th endmember in the  $i$ th pixel.  $r$  is the rate of spatial resolution enhancement.

After the above step, the number of subpixel labels for each mixed pixel is known, whereas the subpixel position is not. In the subpixel mapping process of objective function optimization based on MBQPSO, a number of particles (particle population  $M$ ) are assigned for each mixed pixel. Each of these particles is a bit series and contains  $d - 1$  rows and  $r^2$  columns, where  $d$  is the number of endmembers that the pixel contains, and each column denotes a subpixel location for that pixel. These binary bit series initialize each particle  $X_i^t$  in particle swarm and set  $P_i^t = X_i^t$ .

The  $d$ th feature's distribution can be determined as long as the first  $d - 1$  kinds have been determined. The particle location of  $i$ th pixel can be expressed by  $(d - 1) \times r^2$  dimensional binary matrix. The position matrix of each pixel  $X_i$  must satisfy the needs of two constraints:

- (a) The  $k$ th row only has  $n_k$  elements that are 1.
- (b) Each column cannot have more than one element that is 1.

(b) *Abundance Full Additive Constraint While Particle Updates.* (a) The number of 1-bits in each row of a particle must be  $n_{ik}$ . However, the number of bits, taking 1 in each

particle, is not stable. Under this condition, the number of 1-bits in each row of each particle is checked, and randomly located deletion or addition is conducted to ensure that the number of bits that takes the value of 1 is equal to  $n_{ik}$ .

(b) Only a single 1-bit should be allowed in each column of each particle. When the number of bits takes 1 in a column more than one, this constraint is imposed by randomly moving the overlapping 1-bit for the row with the smaller abundance rate to a column that is empty for all rows [35]. Note that this constraint is enforced under the first constraint. Furthermore, an overlapping 1-bit means that there exists an extra empty column.

2.4. *Optimization Strategy.* After construction of objective function and presentation of modified optimization algorithm, the target optimization strategy of subpixel mapping is the process of global iteration combined with local iteration. Local iteration is a pixel-by-pixel iteration, whereas traversing over all pixels can be considered a global iteration. In order to reduce the time complexity, the optimization strategy is as follows [39].

*Pixel-by-Pixel Iteration.* to construct cost function for each iteration, only calculate the current pixel's corresponding subpixel window and the surrounding structure's local area perimeter.

Figure 2 shows the pixel-by-pixel iterative optimization area diagram. Figure 2(b) is a  $3 \times 3$  resolution enhancement image of the dotted area in Figure 2(a). Denote the pixel with a circle in Figure 2(a) and denote the dotted area in Figure 2(b) as region, which indicate the subpixel window and the surrounding structures local region for the pixel  $x$ . In each iteration, the subpixel mapping for pixel  $x$  is optimized by calculating the perimeter of region  $R_x$  as cost function.

### 3. Experimental Results and Discussion

Experiments are conducted to test the performance of the proposed modified BQPSO algorithm with standard test function, one simulated image, and one real image.

3.1. *The Standard Test Function Experiment.* In standard test function experiments, we use the same standard test function as done in [37]. Each test function implements three sets of

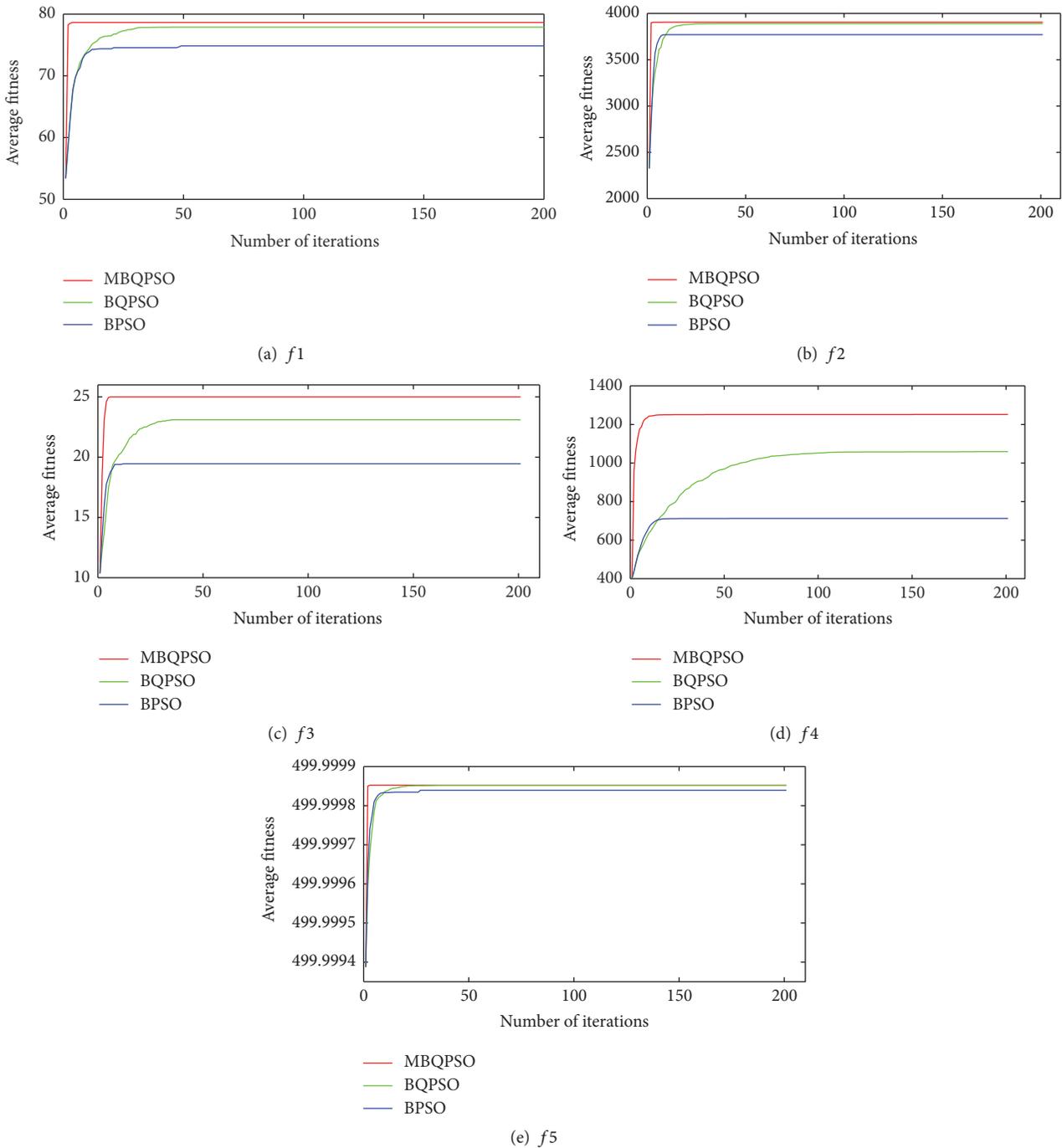


FIGURE 3: Comparison of the convergence curves of objective function values for different test functions with three optimization algorithms.

simulation test and the number of groups is 20, 40, and 80. The BPSO, BQPSO, and MBQPSO algorithm run 20 times [40] and the maximum iteration number is 200. Table 2 records the average optimal value of the three algorithms, the maximum value, the minimum value, and variance of the algorithm in the 20 times simulation.

From Table 2, we can see that the test results of the MBQPSO algorithm are better than those of the BQPSO and BPSO algorithm. In a low dimensional test function, the variance of the test results is greatly reduced. The average

optimal value is significantly improved. The convergence curve from Figure 3 shows that the MBQPSO has the best performance on global optimization ability and convergent rate. The BQPSO algorithm's performance takes the second place and is better than the BPSO in simulation. The curve's adjustment time is shorter and the final result is optimized.

### 3.2. The Subpixel Mapping Experiment of Different Algorithms.

In subpixel mapping experiments, consisting of simulated and real image, we compare the proposed six subpixel

TABLE 2: BPSO, BQPSO, and MBQPSO algorithm simulation results with five test functions.

Function	Number	Algorithm	Index parameter			
			Average optimal value	Variance	Maximum value	Minimum value
$f_1$	20	MBQPSO	<b>78.6</b>	<b>1.5e – 14</b>	<b>78.6</b>	<b>78.6</b>
		BQPSO	78.3	0.47	78.6	76.8
		BPSO	74.5	3.5	78.6	66.6
	40	MBQPSO	<b>78.6</b>	<b>1.5e – 14</b>	<b>78.6</b>	<b>78.6</b>
		BQPSO	78.6	0.11	78.6	78.2
		BPSO	76.9	2.6	77.7	67
	80	MBQPSO	<b>78.6</b>	<b>1.5e – 14</b>	<b>78.6</b>	<b>78.6</b>
		BQPSO	78.6	3.5e – 2	78.6	78.5
		BPSO	78.4	0.4	78.6	77
$f_2$	20	MBQPSO	<b>3905.9</b>	<b>4.7e – 13</b>	<b>3905.9</b>	<b>3905.9</b>
		BQPSO	3885.3	37.2	3905.9	3735.2
		BPSO	3736.6	296.4	3900	2603.5
	40	MBQPSO	<b>3905.9</b>	<b>4.7e – 13</b>	<b>3905.9</b>	<b>3905.9</b>
		BQPSO	3902.1	11	3905.9	3860.1
		BPSO	3871.2	58.9	3900	3639.5
	80	MBQPSO	<b>3905.9</b>	<b>4.7e – 13</b>	<b>3905.9</b>	<b>3905.9</b>
		BQPSO	3904.8	4.56	3905.9	3885.5
		BPSO	3893.6	20.2	3905.9	3821.5
$f_3$	20	MBQPSO	<b>25</b>	<b>0</b>	<b>25</b>	<b>25</b>
		BQPSO	23.1	1.119	25	21
		BPSO	19.6	1.932	22	15
	40	MBQPSO	<b>25</b>	<b>0</b>	<b>25</b>	<b>25</b>
		BQPSO	24	0.999	25	22
		BPSO	21.5	1.317	23	19
	80	MBQPSO	<b>25</b>	<b>0</b>	<b>25</b>	<b>25</b>
		BQPSO	24.8	0.44	25	24
		BPSO	22.3	1.129	25	21
$f_4$	20	MBQPSO	<b>1251.8</b>	<b>0.2</b>	<b>1252.3</b>	<b>1251.6</b>
		BQPSO	1046.7	36.7	1119.3	968.9
		BPSO	734.45	43.131	818.7	661.39
	40	MBQPSO	<b>1251.8</b>	<b>0.2</b>	<b>1252.3</b>	<b>1251.5</b>
		BQPSO	1114.4	32.6	1157.5	1039.5
		BPSO	804.1	35.847	881.3	722.36
	80	MBQPSO	<b>1252</b>	<b>0.3</b>	<b>1252.7</b>	<b>1251.7</b>
		BQPSO	1149.2	31	1190.9	1082.3
		BPSO	882.8	42.979	961.7	809.75
$f_5$	20	MBQPSO	<b>499.9</b>	<b>0</b>	<b>499.9</b>	<b>499.9</b>
		BQPSO	499.9	1e – 6	499.9	499.9
		BPSO	499.9	0.8e – 5	499.9	499.9
	40	MBQPSO	<b>499.9</b>	<b>0</b>	<b>499.9</b>	<b>499.9</b>
		BQPSO	499.9	0	499.9	499.9
		BPSO	499.9	0.5e – 5	499.9	499.9
	80	MBQPSO	<b>499.9</b>	<b>0</b>	<b>499.9</b>	<b>499.9</b>
		BQPSO	499.9	0	499.9	499.9
		BPSO	499.9	0.1e – 5	499.9	499.9

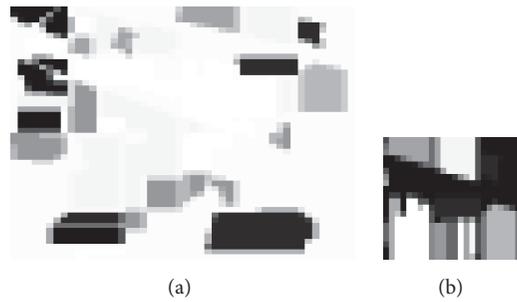


FIGURE 4: Simulated hyperspectral imagery (the thirtieth band): (a) image 1 ( $48 \times 48$ ); (b) image 2 ( $20 \times 25$ ).

mapping algorithms which are combined by three objective functions ( $f_{\text{point}}$ ,  $f_{\text{chain}}$ , and  $f_{\text{gap}}$ ) and two optimization algorithms (BQPSO and MBQPSO), named as PBQPSO, CBQPSO, GBQPSO, PMBQPSO, CMBQPSO, and GMBQPSO. In addition, we also compare the six proposed methods with pixel swapping algorithm (PSA) in simulated and real experiment. The results of the six proposed subpixel mappings are compared under the conditions of different objective functions and different optimization strategies. Since the subpixel mapping result is the same as the classification result, the Kappa coefficient and recognition rate are used to evaluate the performance of the different subpixel mapping algorithms.

In order to construct an accurate abundance map, based on the real features distribution of the Indiana AVIRIS Indian pines hyperspectral data (size is:  $145 \times 145$ ), the ground truth gives 17 kinds of different features (including the background), and then some of these classes are merged into nine categories. The simulated hyperspectral image is generated by nine kinds of spectral feature on the basis of category labels, which are selected from the USGS spectral library.

Two hyperspectral image pictures are constructed for the later experiments:

(1) Figure 4(a) is captured from the original image's [1:144, 1:144] region, which is then filtered by the  $3 \times 3$  mean filter and narrowly shown in image 1.

(2) Figure 4(b) is captured from the original image's [1:60, 70:144] region, which is then filtered by the  $3 \times 3$  mean filter and narrowly shown in image 2. So image 2 is in fact the subfigure of image 1.

This part has five experiments, and only the fourth experiment's spatial resolution is  $5 \times 5$ ; the others are  $3 \times 3$ . In order to avoid the influence of the initial value sensitivity of the particle swarm algorithm, the same initial particle swarm is used in each experiment and the results run for five times.

We compare the results of the above seven algorithms for Figure 4(b). The initial recognition rate (the ratio of the number of correct classification result label to the total number of category labels) of the seven experiments is 93.89%. For BQPSO-based and MBQPSO-based method, the particle number is 50; the number of maximum iteration steps is 30.

Table 3 gives the results of the subpixel location result, combining the different objective functions and the different

TABLE 3: Comparison of subpixel mapping results for Figure 4(b) with six kinds of algorithms.

Algorithm	recognition rates%	Kappa coefficient	time (s)
PSA	94.1	0.84	3.9
PMBQPSO	96.5	0.91	13
CMBQPSO	96.1	0.90	238
GMBQPSO	98.7	0.96	23
PBQPSO	96.3	0.90	4
CBQPSO	95	0.87	178
GBQPSO	97.9	0.94	4

optimization algorithms. As can be seen from Table 3, we have the following:

(1) Under the same objective function, recognition rate and Kappa coefficient of MBQPSO are greatly improved.

(2) Comparing the results of objective function calculation and subpixel mapping, we learn that the  $f_{\text{gap}}$  method is more suitable for constructing the target function of the subpixel position, which has the advantages of the shortest time and the best mapping results.

(3) Comparing the results of PSA with the six methods, based on BQPSO and MBQPSO, the proposed six methods can obtain higher recognition rates and Kappa coefficient.

As can be seen from Figure 5, MBQPSO had fewer isolated points than BQPSO and the result is better. From the iterative curves of Figure 6, all the results of BQPSO are only three times iterative convergence. The number of iteration steps is increased after optimization and the corresponding results are better. Compared with the above six, PSA had more isolated points and the recognition rates and Kappa coefficient are lower.

### 3.3. The Influence of Parameters

**3.3.1. The Number of Particles.** To investigate the influence of the number of particles, the following simulation parameters are used: set the maximum number of iterations to 30 and set the number of particles at 30, 40, 50, 60, and 100. The test results are shown in Figure 7.

The following information can be seen from Figure 7.

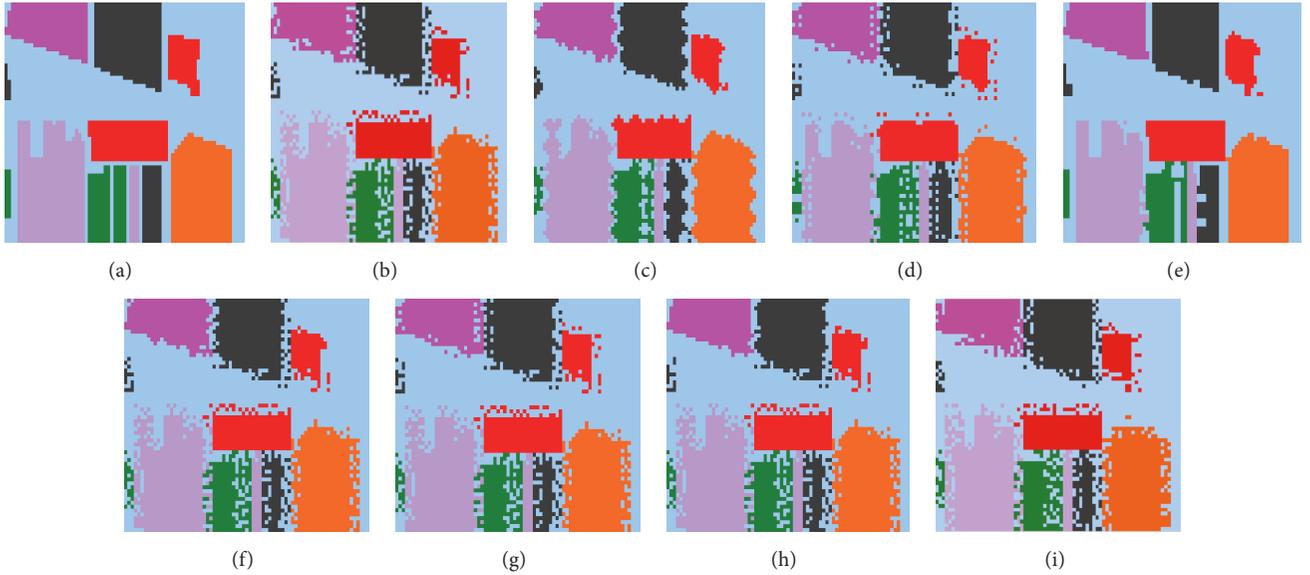


FIGURE 5: Subpixel mapping results with the six kinds of algorithms for Figure 4(b): (a) true distribution, (b) random distribution, (c) PMBQPSO, (d) CMBQPSO, (e) GMBQPSO, (f) PBQPSO, (g) CBQPSO, (h) GBQPSO, and (i) PSA.

(1) Under the same number of particles, the recognition rate and the Kappa coefficient are the best using MBQPSO combined with  $f\_gap$  method to calculate perimeter.

(2) With the increase of the particles number, running time generally has an upward trend. Among them, the calculation results of the  $f\_point$  method and MBQPSO present a rising trend on the whole, with an increasing number of particles. But at the middle of the function it slightly declines, which may be caused by the uncertainty of the two constraints during the iterative process.

(3) The running time of the  $f\_gap$  method becomes longer with the increase of the number of particles. The recognition rate and Kappa coefficient decrease at first and then increase. The two parameters of the  $f\_point$  and  $f\_chain$  methods are increased at first and then decreased. The recognition rate of GMBQPSO algorithm is the highest in 50 particles.

**3.3.2. Adaptability of Image Size.** This section verifies the applicability of image size with the modified algorithm. Based on the real feature distribution of the whole Indian pines image, we design a simulation experiment, as shown in Figure 8(a). Using image 1 in Figure 4, the results of the above six algorithms are compared. The number of particles is 50 and the maximum iteration number is 30. Figure 8 depicts the result of subpixel mapping and the evaluation index of the mapping results is given as follows.

Table 4 and Figure 8 show the improvement of the modified algorithm. The results of Table 4 quantitatively confirm that the GMBQPSO algorithm has the best applicability and that the Kappa coefficient and the recognition rate are the highest, but the operation time is longer. From Figure 8 it can be seen that there are many isolated points among the results of the BQPSO algorithm. Most boundaries between

TABLE 4: Comparison of whole image's subpixel mapping results of six algorithms.

Algorithm	Recognition rates%	Kappa coefficient	Time (s)
PMBQPSO	97.3	0.92	96
CMBQPSO	96.6	0.90	2267
GMBQPSO	98.9	0.97	181
PBQPSO	97	0.92	20
CBQPSO	95.5	0.87	796
GBQPSO	98	0.95	21

categories of MBQPSO running results are more obvious. Comparing Tables 4 and 3, it can be seen that the image size has no effect on subpixel mapping accuracy. The bigger the image is, the larger the mixed pixels' number is. After statistical analysis in Figure 4, the number of mixed pixels in image 1 is 5.5 times the number of pixels in image 2. Comparing the operating time of Table 4 with Table 3, considering the two constraints, the operating time is proportional to the number of mixed pixels in the image. In summary, the GMBQPSO is superior to other algorithms in both visual perception and quantitative evaluation results (the Kappa coefficient and the recognition rate).

**3.3.3. Influence of Spatial Resolution Enhancement Ratio.** In order to verify the subpixel mapping practicality of the optimization objective function on the image spatial resolution enhancement ratio, which is greater than  $3 \times 3$ , the original image is filtered by  $5 \times 5$  filtering and subpixel mapping is implemented by a spatial enhancement ratio  $5 \times 5$ .

Because the  $5 \times 5$  filtered image contained fewer pure pixels, it is necessary to adjust the abundance threshold. A pixel can be seen as a pure pixel only if the minimum

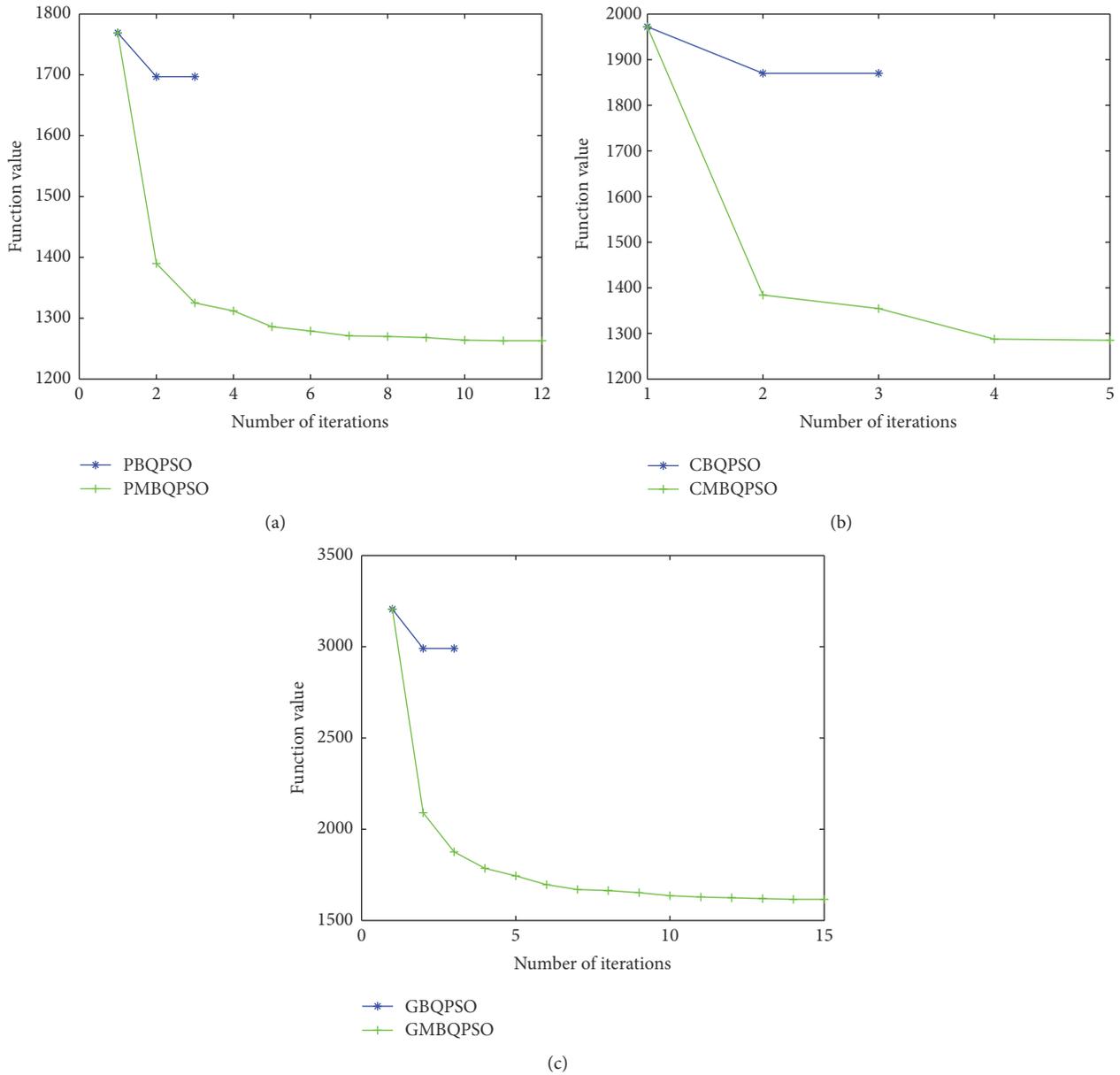


FIGURE 6: The iterations change curves with the space resolution enhancement ratio  $3 \times 3$ : (a)  $f_{\text{point}}$  method, (b)  $f_{\text{chain}}$  method, and (c)  $f_{\text{gap}}$  method.

difference value between the maximum abundance and the others in the pixel is greater than 0.9.

As can be seen from Table 5, we have the following:

(1) The improved MBQPSO results are obviously better than those of the BQPSO algorithm.

(2) In the case of 100 particles, the recognition rate and the Kappa coefficient are superior to the 50 particles, but the running time is longer. Considering time and accuracy, the choice of 50 particles is reasonable.

In Figure 9, the results show that MBQPSO combined with three kinds of perimeter calculation methods had fewer isolated points and the GMBQPSO results are optimal. But

there still exist other isolated cases in the original  $5 \times 5$  filtering image. In practical application, we should take further measures to achieve better results.

3.4. *The Real Image Subpixel Mapping Experiment.* The experiments on real images are conducted by considering two different hyperspectral pieces of data. The first experimental data uses a ROSIS hyperspectral remote sensing image, which is located in the north of Italy, Pavia University. The size of the original image is  $610 \times 340$  pixels and it contains nine classes of pixel number ranging from 947 to 18649 pixels. The band number is 103 and the geometric resolution is

TABLE 5: Results of subpixel mapping with spatial resolution enhancement ratio  $5 \times 5$ .

Algorithm	$n = 50$			$n = 100$		
	Recognition rates%	Kappa coefficient	Time (s)	Recognition rates%	Kappa coefficient	Time (s)
PMBQPSO	92.9	0.82	27	93.7	0.83	59
CMBQPSO	91.4	0.78	394	91.6	0.78	446
GMBQPSO	93.4	0.82	44	93.8	0.85	123
PBQPSO	91.2	0.78	3	92.7	0.82	17
CBQPSO	90.8	0.77	88	91.2	0.78	343
GBQPSO	91.5	0.79	3	93.3	0.83	22

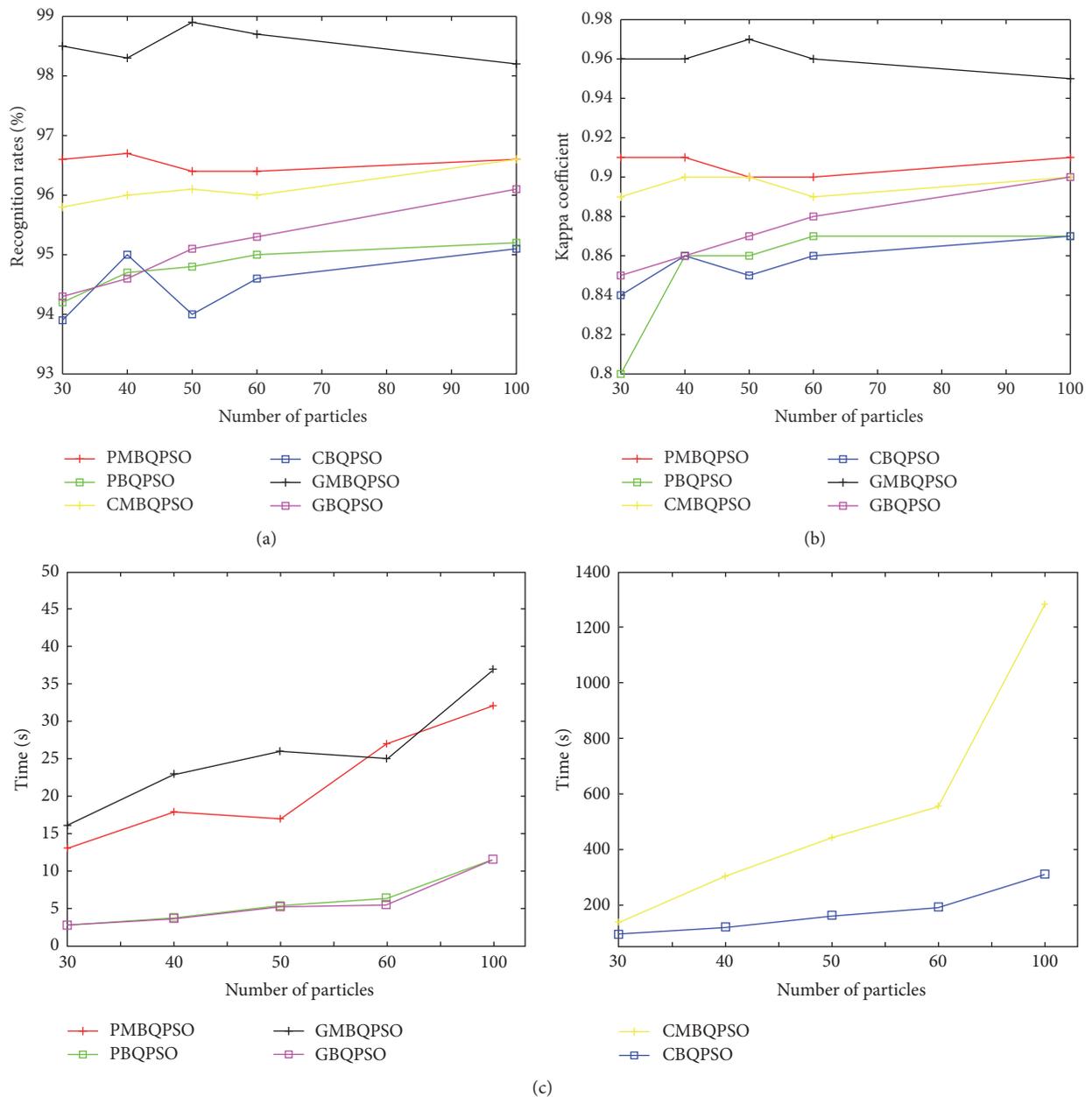


FIGURE 7: Comparison of different particle number: (a) recognition rates curve with particle number, (b) Kappa coefficient curve with particle number, and (c) iteration time curve with particle number.

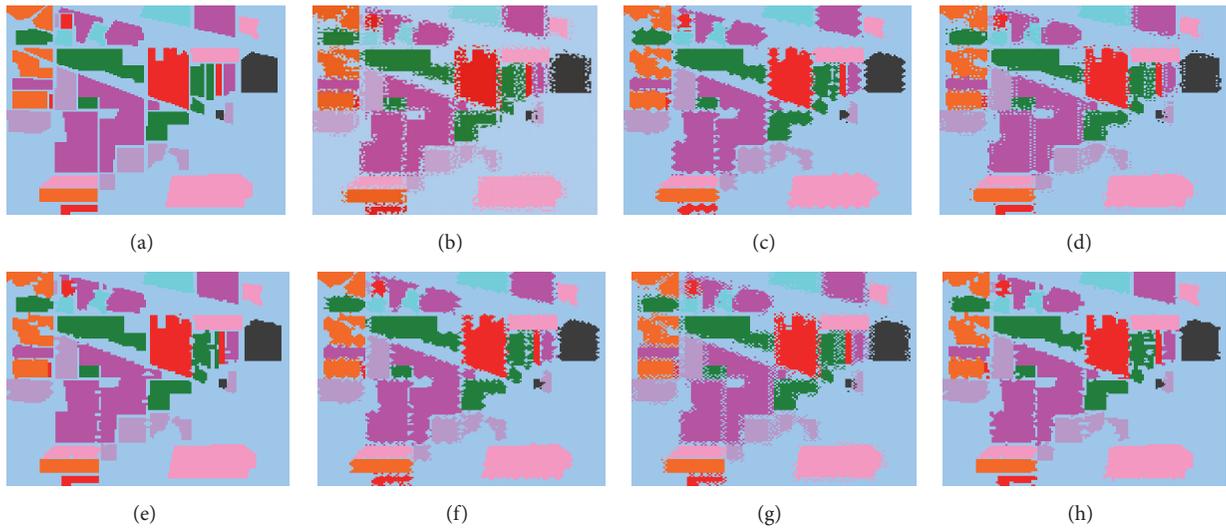


FIGURE 8: Subpixel positioning results of six algorithms for Figure 4(a): (a) true distribution, (b) random distribution, (c) PMBQPSO, (d) CMBQPSO, (e) GMBQPSO, (f) PBQPSO, (g) CBQPSO, and (h) GBQPSO.

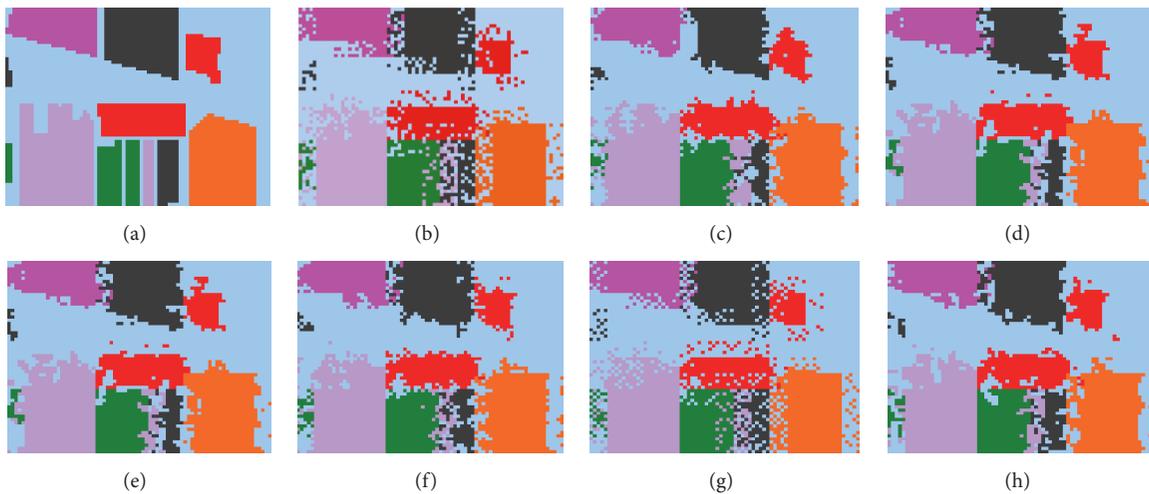


FIGURE 9: The subpixel mapping results of the  $5 \times 5$  filter image with the six algorithms: (a) true distribution, (b) random distribution, (c) PMBQPSO, (d) CMBQPSO, (e) GMBQPSO, (f) PBQPSO, (g) CBQPSO, and (h) GBQPSO.

1.3 meters. We sample a  $117 \times 75$  region where five kinds of features and the real surface features' distribution are shown in Figure 10(b). The  $3 \times 3$  mean filter is developed for the original image to obtain the 60th band gray image Figure 10(a). The number of endmembers is estimated as 5 by VD [38]. A pixel can be seen as a pure pixel only if the minimum difference value between the maximum abundance and the others in the pixel is greater than a certain threshold [33–35]. Because a metal area is obvious, we only perform a statistical analysis of the recognition rate on a metal area. Figure 10(c) shows the random mapping results. Figures 10(d)–10(i) represent the subpixel mapping results with different optimization algorithm, respectively. Table 6 gives a comparison of recognition rate.

The second image analyzed in our experiments is a Salinas dataset. The size of the original image is  $510 \times 216$  pixels and it contains 204 bands and contained 17 classes. The  $3 \times 3$  mean filter is developed for the original image to obtain the  $170 \times 72$  pixels downsampled image and we select three bands to obtain the pseudocolor image Figure 11(a). We combine class 1 and class 2 in one class, combine class 8 and class 15 in one class, and let class 16 be named class 2. The number of endmembers is estimated as 15 by VD and endmembers are extracted by calculating the mean of each category. A pixel can be seen as a pure pixel only if the minimum difference value between the maximum abundance and the others in the pixel is greater than a certain threshold. The background image is shown in

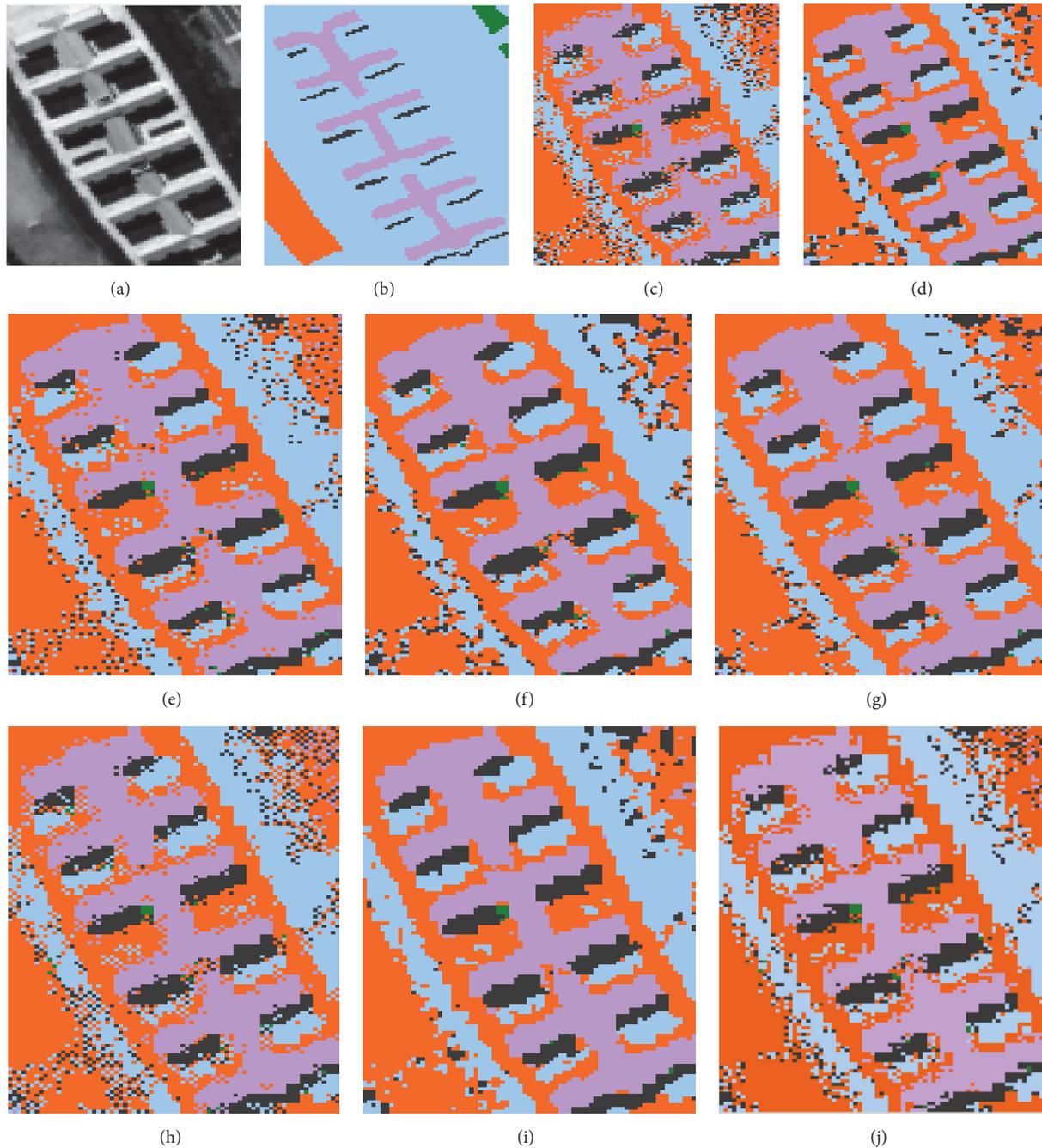


FIGURE 10: The subpixel mapping results of the PaviaU original subimages: (a) 60th band gray image, (b) true distribution, (c) random distribution, (d) PMBQPSO, (e) CMBQPSO, (f) GMBQPSO, (g) PBQPSO, (h) CBQPSO, (i) GBQPSO, and (j) PSA.

Figure 11(b). In this part, we only perform a statistical analysis of the recognition rate on the other 14 categories without background. Figure 11(c) shows the random mapping results. Figures 11(d)–11(f) represent the subpixel mapping results with different optimization algorithm, respectively. Table 7 gives a comparison of recognition rate.

Table 6 gives a comparison of the recognition rate and the running time of various algorithms with the abundance threshold at 0.4 and 0.5, respectively. Figure 10 gives the

results of the algorithm with the abundance threshold at 0.4 and 50 particles.

From Table 6, we have the following:

(1) With the same number of particles, the recognition rate of the MBQPSO is better than that of the BQPSO, and the recognition rate of GMBQPSO is the highest.

(2) The results of the abundance threshold at 0.4 are better than those at 0.5, and the recognition rate and Kappa coefficient are all improved.

TABLE 6: PaviaU subimage subpixel mapping results.

Algorithm	Particle number	0.5		0.4	
		Recognition rates%	Time (s)	Recognition rates%	Time (s)
PSA	—	92.7	7	94.3	6
PMBQPSO	50	97.4	88	97.7	98
	100	97.6	163	97.8	148
PBQPSO	50	97.1	12	97.4	11
	100	97.2	25	97.5	19
CMBQPSO	50	94.8	1036	96.2	400
	100	95.7	1089	95.9	793
CBQPSO	50	94.9	315	95.6	242
	100	95	625	95.5	473
GMBQPSO	50	97.7	199	98.2	115
	100	97.9	283	98.4	249
GBQPSO	50	97.5	13	97.4	11
	100	97.9	25	97.7	21

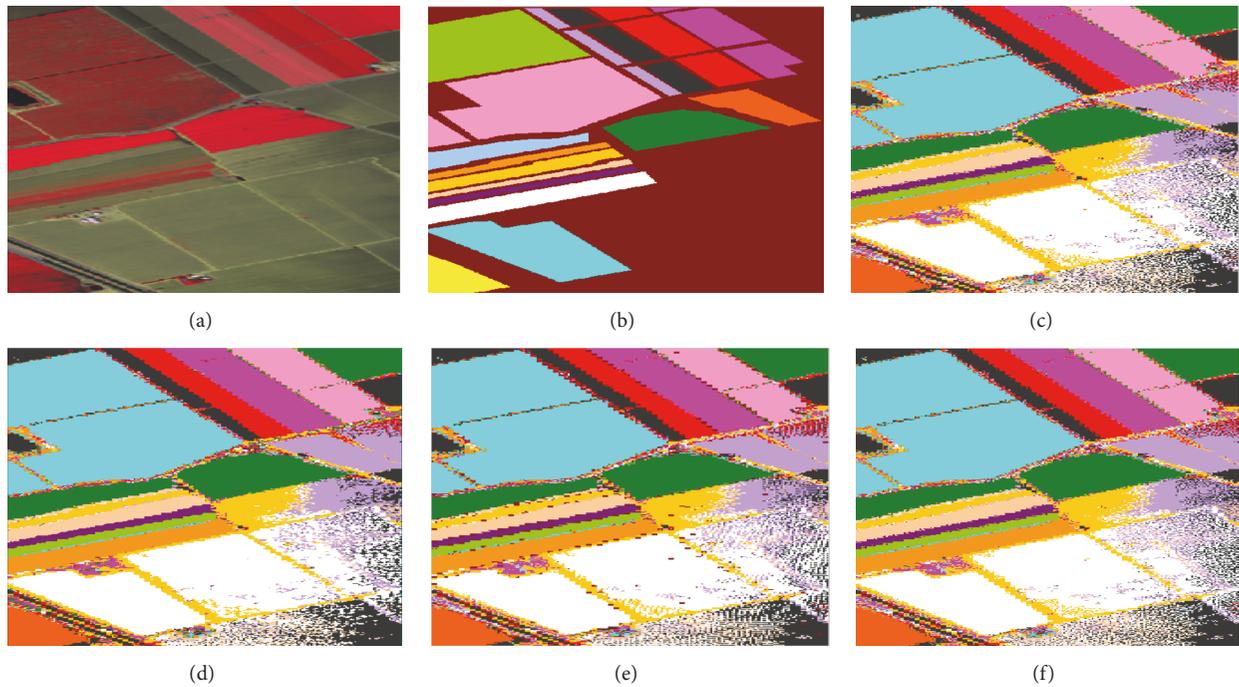


FIGURE 11: The subpixel mapping results of the PaviaU original subimages: (a) pseudocolor image, (b) true distribution, (c) random distribution, (d) GMBQPSO, (e) GBQPSO, and (f) PSA.

(3) From the point of view of the particle number, the Kappa coefficient and the recognition rate of the total particle number 100 can be improved, but time is longer.

(4) Compared with the above six, PSA had more isolated point, and the recognition rates and Kappa coefficient are lower.

As analysis in the first real experimental result, we present the comparison of the recognition rate and the running time of various algorithms with the abundance threshold at 0.4 in Table 7. Figure 11 gives the results of the algorithm with

the abundance threshold at 0.4 and 50 particles. Based on the subpixel mapping results in the Table 6, we only present the subpixel mapping results image of GMBQPSO, GBQPSO, and PSA in Figure 11. We can see that the marginal area of MBQPSO's mapping results is better than others. The recognition rate of the MBQPSO is better than that of the corresponding BQPSO, and the recognition rate of GMBQPSO is the highest. The running time of the corresponding BQPSO is shorter than MBQPSO, owing to premature convergence of BQPSO-based method. In addition, the running time is

TABLE 7: Salinas subpixel mapping results.

Algorithm	0.4	
	Recognition rates%	Time (s)
PSA	99.14	94
PMBQPSO	99.56	1089
PBQPSO	99.52	104
CMBQPSO	99.39	20670
CBQPSO	99.33	9870
GMBQPSO	99.68	1127
GBQPSO	99.54	137

larger than the experiment result of ROSIS dataset because the size of the Salinas dataset is about 20 times larger than ROSIS dataset.

#### 4. Conclusion

In this paper, a modified method of subpixel mapping based on binary quantum particle swarm optimization (MBQPSO) algorithm is proposed. In order to improve the adaptability of objective function and lower time complexity of optimization algorithm, we use three kinds of regional perimeter minimum to describe spatial correlation. At the same time, the discretization process of particle position updating is corrected to overcome premature convergence in the original BQPSO algorithm iteration process. Simulated and real image experiments show that the proposed method in this paper is superior to the original method in the Kappa coefficient, recognition rate, and running time. It can improve the mapping accuracy of the algorithm. The  $f\_gap$  method obtains the optimal mapping result and the fastest speed, which is suitable for subpixel mapping. The method proposed in this paper can not only reflect the spatial detail information of the ground objects, but also reconstruct the shape feature information in a small area. Compared with the original linear optimization method, visual effect and practical classification accuracy are all improved greatly. The assumption of the spatial correlation between pixels is suitable in most cases, but because of the complexity of actual situations, this assumption will have some errors in some cases. Therefore, it is necessary to conduct further research to obtain a better expression of the pixel spatial distribution rules to improve accuracy of subpixel mapping.

#### Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

#### Acknowledgments

This work was supported by the National Natural Science Foundation of China (no. 61671408), Shanghai Aerospace Science and Technology Innovation Fund (no. SAST2015033), and the Joint Funds of the Ministry of Education of China (no. 6141A02022314).

#### References

- [1] P. M. Atkinson, "Mapping sub-pixel boundaries from remotely sensed images," in *Innovations in Gis*, Taylor & Francis, Z. Kemp, Ed., vol. 4, 1997.
- [2] A. J. Tatem, H. G. Lewis, P. M. Atkinson, and M. S. Nixon, "Super-resolution target identification from remotely sensed images using a Hopfield neural network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 4, pp. 781–796, 2001.
- [3] A. J. Tatem, H. G. Lewis, P. M. Atkinson, and M. S. Nixon, "Super-resolution mapping of multiple-scale land cover features using a Hopfield neural network," in *Proceedings of the International Geoscience and Remote Sensing Symposium*, vol. 7, pp. 3200–3202, July 2001.
- [4] A. J. Tatem, H. G. Lewis, P. M. Atkinson, and M. S. Nixon, "Super-resolution land cover pattern prediction using a Hopfield neural network," *Remote Sensing of Environment*, vol. 79, no. 1, pp. 1–14, 2002.
- [5] K. Mertens, B. De Baets, L. Verbeke, and R. De Wulf, "Sub-pixel mapping with neural networks: real-world spatial configurations learned from artificial shapes," in *Proceedings of the 4th International Symposium on Remote Sensing and Urban Areas; WG VII/4 Workshop*, pp. 3046–3049, Anchorage, AK, USA, September 2004.
- [6] K. C. Mertens, L. P. C. Verbeke, T. Westra, and R. R. De Wulf, "Sub-pixel mapping and sub-pixel sharpening using neural network predicted wavelet coefficients," *Remote Sensing of Environment*, vol. 91, no. 2, pp. 225–236, 2004.
- [7] K. C. Mertens, L. P. C. Verbeke, E. I. Ducheyne, and R. R. De Wulf, "Using genetic algorithms in sub-pixel mapping," *International Journal of Remote Sensing*, vol. 24, no. 21, pp. 4241–4247, 2003.
- [8] K. Wu, R. Niu, P. Li, and L. Zhang, "Sub-pixel mapping of remote sensing images based on fuzzy ARTMAP neural network model," *Geomatics and Information Science of Wuhan University*, vol. 34, no. 3, pp. 297–300, 2009.
- [9] F. Ling, Y. Du, F. Xiao, H. Xue, and S. Wu, "Super-resolution land-cover mapping using multiple sub-pixel shifted remotely sensed images," *International Journal of Remote Sensing*, vol. 31, no. 19, pp. 5023–5040, 2010.
- [10] X. U. Xiong, Y. Zhong, and L. Zhang, "A sub-pixel mapping algorithm based on BP neural network with spatial autocorrelation function for remote sensing imagery," *Acta Geodaetica Et Cartographica Sinica*, vol. 40.3, pp. 307–311, 2011.
- [11] W. Z. Shi, Y. L. Zhao, and Q. M. Wang, "Sub-pixel mapping based on BP neural network with multiple shifted remote sensing images," *Journal of Infrared & Millimeter Waves*, vol. 33, no. 5, pp. 527–532, 2014.
- [12] L. Li, T. Xu, and Y. Chen, "Improved urban flooding mapping from remote sensing images using generalized regression neural network-based super-resolution algorithm," *Remote Sensing*, vol. 8, no. 8, article no. 625, 2016.
- [13] A. Boucher and P. C. Kyriakidis, "Super-resolution land cover mapping with indicator geostatistics," *Remote Sensing of Environment*, vol. 104, no. 3, pp. 264–282, 2006.
- [14] A. Boucher and P. C. Kyriakidis, "Integrating fine scale information in super-resolution land-cover mapping," *Photogrammetric Engineering and Remote Sensing*, vol. 73, no. 8, pp. 913–921, 2007.
- [15] H. R. Jin, G. Mountrakis, and P. Li, "A super-resolution mapping method using local indicator variograms," *International Journal of Remote Sensing*, vol. 33, no. 24, pp. 7747–7773, 2012.

- [16] T. Kasetkasem, M. K. Arora, and P. K. Varshney, "Super-resolution land cover mapping using a Markov random field based approach," *Remote Sensing of Environment*, vol. 96, no. 3-4, pp. 302-314, 2005.
- [17] V. A. Tolpekin and A. Stein, "Quantification of the effects of land-cover-class spectral separability on the accuracy of markov-random-field-based superresolution mapping," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 9, pp. 3283-3297, 2009.
- [18] J. P. Ardila, V. A. Tolpekin, W. Bijker, and A. Stein, "Markov-random-field-based super-resolution mapping for identification of urban trees in VHR images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 6, pp. 762-775, 2011.
- [19] X. Li, F. Ling, and Y. Du, "Super-resolution mapping based on the supervised fuzzy  $c$ -means approach," *Remote Sensing Letters*, vol. 3, no. 6, pp. 501-510, 2014.
- [20] Q. Liu and J. Trinder, "Sub-pixel mapping of multispectral images using markov random field with graph cut optimization," *IEEE Geoscience & Remote Sensing Letters*, vol. 13, pp. 1-5, 2016.
- [21] J. Verhoeve and R. De Wulf, "Land cover mapping at sub-pixel scales using linear optimization techniques," *Remote Sensing of Environment*, vol. 79, no. 1, pp. 96-104, 2002.
- [22] Y. Zhong and L. Zhang, "Sub-pixel mapping algorithm based on adaptive differential evolution for remote sensing imagery," in *Proceedings of the 2011 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2011*, pp. 1724-1727, July 2011.
- [23] Y. Zhong, L. Zhang, P. Li, and H. Shen, "A sub-pixel mapping algorithm based on artificial immune systems for remote sensing imagery," in *Proceedings of the 2009 IEEE International Geoscience and Remote Sensing Symposium, IGARSS 2009*, pp. 1007-1010, July 2009.
- [24] K. Wu, P. X. Li, and L. P. Zhang, "Sub-pixel mapping of remote sensed images based on evolutionary agent algorithm," *Journal of Remote Sensing*, vol. 13, no. 1, pp. 60-66, 2009.
- [25] K. C. Mertens, B. de Baets, L. P. C. Verbeke, and R. R. de Wulf, "A sub-pixel mapping algorithm based on sub-pixel/pixel spatial attraction models," *International Journal of Remote Sensing*, vol. 27, no. 15, pp. 3293-3310, 2006.
- [26] F. Ling, "Sub-pixel mapping of remote sensing images based on cellular automata model," *Journal of Image & Graphics*, vol. 10, no. 7, pp. 916-921, 2005.
- [27] Q. M. Wang, *Research on Sub-Pixel Mapping And Its Related Techniques for Remote Sensing Imagery, A Dissertation for degree of Master Engineering [M.S. thesis]*, Harbin Engineering University, 2012.
- [28] Q. Wang, L. Wang, and D. Liu, "Particle swarm optimization-based sub-pixel mapping for remote-sensing imagery," *International Journal of Remote Sensing*, vol. 33, no. 20, pp. 6480-6496, 2012.
- [29] X. Xu, Y. Zhong, L. Zhang, and H. Zhang, "Sub-pixel mapping based on a MAP model with multiple shifted hyperspectral imagery," *IEEE Journal of Selected Topics in Applied Earth Observations & Remote Sensing*, vol. 6, no. 2, pp. 580-593, 2013.
- [30] Y. Zhong, Y. Wu, X. Xu, and L. Zhang, "An adaptive sub-pixel mapping method based on MAP model and class determination strategy for hyperspectral remote sensing imagery," *IEEE Transactions on Geoscience & Remote Sensing*, vol. 53, no. 3, pp. 1411-1426, 2015.
- [31] Z. Shen and Z. Yu, "Sub-pixel mapping with partheno-genetic algorithm and sub-pixel/pixel spatial attraction model," *Remote Sensing Technology and Application*, vol. 30, no. 1, pp. 129-134, 2015.
- [32] Q. Wang and P. M. Atkinson, "The effect of the point spread function on sub-pixel mapping," *Remote Sensing of Environment*, vol. 193, pp. 127-137, 2017.
- [33] A. Villa, J. Chanussot, J. A. Benediktsson, C. Jutten, and R. Dambrevelle, "Unsupervised methods for the classification of hyperspectral images with low spatial resolution," *Pattern Recognition*, vol. 46, no. 6, pp. 1556-1568, 2013.
- [34] M. W. Thornton, P. M. Atkinson, and D. A. Holland, "Sub-pixel mapping of rural land cover objects from fine spatial resolution satellite sensor imagery using super-resolution pixel-swapping," *International Journal of Remote Sensing*, vol. 27, no. 3, pp. 473-491, 2006.
- [35] A. Ertürk, "Spatial resolution enhancement of hyperspectral images using un-mixing and binary particle swarm optimization," *IEEE Geoscience & Remote Sensing Letters*, vol. 11, no. 12, pp. 2100-2104, 2014.
- [36] J. Sun, B. Feng, and W. Xu, *Particle Swarm Optimization with Particles Having Quantum Behavior, A Dissertation for Degree of Doctor Engineering [M.S. thesis]*, Jiangnan University, 2009.
- [37] M. L. Xi, J. Sun, and Y. Wu, "Quantum-behaved particle swarm optimization with binary encoding," *Control and Decision*, vol. 25, no. 1, pp. 376-385, 2010.
- [38] C. Chang and Q. Du, "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Transactions on Geoscience & Remote Sensing*, vol. 42, no. 3, pp. 608-619, 2004.
- [39] L. Y. Zhao, M. Y. Fan, and X. R. Li, "Sub-pixel mapping of hyperspectral imagery based on object optimization," *Journal of Image and Graphics*, vol. 21, no. 6, pp. 823-833, 2016.
- [40] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Computational Cybernetics and Simulation*, vol. 5, pp. 4104-4108, Orlando, Fla, USA, October 1997.



**Hindawi**

Submit your manuscripts at  
<https://www.hindawi.com>

