A HYBRIDIZATION OF AN IMPROVED PARTICLE SWARM OPTIMIZATION AND FUZZY K-MEANS ALGORITHM FOR HYPERSPECTRAL IMAGE CLASSIFICATION

Qiang Chen¹, Jie Jiang¹, Mingyi Du^{1*}, Lei Zhou¹, Changfeng Jing¹, Chang Lu¹

¹ School of Geomatics and Urban Spatial Informatics, Beijing University of Civil Engineering and Architecture, Beijing, P.R.China - (chenqiang, jiangjie, dumingyi, zhoulei, jingcf)@bucea.edu.cn,

2108521518018@stu.bucea.edu.cn.

Commission III, WG III/4

KEY WORDS: Hyperspectral Remote Sensing, Land Cover Classification, Fuzzy K-Means, Particle Swarm Optimization, Wetlands.

ABSTRACT:

A particle swarm optimization (PSO) algorithm has been widely used in the field of remote sensing image classification. We proposed the IPSO-FKM algorithm, which use the improved PSO (IPSO) algorithm to optimize the initial parameters of the Fuzzy K-Means (FKM) clustering algorithm. We combine the crossover operator of genetic algorithms with PSO, and introduce the fuzzy membership degree of fuzzy mathematics into K-means clustering algorithm. Then we use the IPSO-FKM algorithm to optimize the classification results of the Hyperion remotely sensed images, and use FKM, IPSO, and IPSO-FKM to extract the land cover information on the wetlands in Dongting Lakes, China. The experimental results have been validated by the classification results of MLC and the field investigation data. The validation results have been evaluated from three perspectives: the overall classification accuracy and the Kappa coefficient from the pixel perspective, the intra-cluster distance and the inter-cluster distance from the feature perspective, and the partition coefficient and partition entropy from the information perspective. According to the comparison of IPSO and FKM algorithms, the IPSO-FKM algorithm has a better performance than the others in all three respects. Additionally, in terms of the fitness convergence, the IPSO-FKM algorithms.

1. INTRODUCTION

Data clustering is a process of grouping together similar multidimensional data vectors into a number of clusters. A set of pixels must be divided into many clusters according to their similarity and are adjusted over many iterations. The K-means algorithm is a clustering algorithm application in unsupervised image classification. Introducing a fuzzy mechanism to Kmeans, the Fuzzy K-means clustering algorithm performed fast convergence and rational clustering partitions. FKM is an iteration optimal algorithm; its objective function is built based on the weighted distance between each pixel in the image and each of the cluster centers.

Eberhart and Kennedy(Kennedy and Eberhart, 1995) proposed the Particle Swarm Optimization (PSO) algorithm first in 1995; this approach is a simple and effective new-style evolutionary computation technique and is similar to a genetic algorithm or an ant colony algorithm. PSO is a population-based optimization algorithm and a type of evolutionary computation that is based on the swarm intelligence algorithm(Civicioglu and Besdok, 2013). The basic idea comes from the study of birds and has a simplified social model, which includes the simulation of bird migration and aggregation behavior in the process of foraging. The algorithm has typical characteristics of swarm intelligence. It uses an information-sharing mechanism, which allows individuals to learn from each other to promote the development of the entire swarm(Jain, Nangia et al., 2018).

Compared with other swarm intelligence evolutionary algorithms, the PSO has many advantages, such as smaller

This paper improves the PSO by introducing the crossover operator of the GA to be a hybrid PSO; moreover, it explores and develops the applicability of the PSO to cluster the data vectors by combining it with fuzzy K-means. In the process of doing so, the objective of the paper is twofold: to show that the hybrid PSO algorithm can be used in the field of image clustering and to develop a clustering algorithm in which fuzzy K-means clustering is used to seed the initial swarm.

parameter settings, a stronger ability to perform a global search, and a higher precision in clustering. PSO algorithms have been successfully applied in vehicle scheduling, image segmentation, and spectral analysis(Chen, Lin et al., 2009, Ding, Jiang et al., 2014). However, the PSO easily runs into local extreme and has some drawbacks, including incomplete convergence, slow convergence speed, and poor local search ability. Learning from biological evolutionary theory, the Genetic Algorithm adopts the crossover operator and mutation operator to improve the global and local searching capability. It models the problem as a process of biological evolution, produces the next generation of solutions by reproduction, performs crossover and mutation operators, and gradually eliminates the solutions that have poor fitness values. After such an evolutionary process, the system can evolve individuals that have a high fitness function value. The crossover operator of GA advances the global searching capability and avoids running into local extreme and converging too early. Thus, this crossover operator can also be used in a PSO(Starke, Hendrich et al., 2017).

^{*} Corresponding author

2. METHODOLOGY

2.1 Improved Particle Swarm Optimization

With respect to optimization problems, most optimization algorithms are committed to solving a problem by looking for the optimal solution in a solution space that contains many potential solutions. In this search procedure, most of the optimization algorithms use the evolution operators on the individuals to drive some to find the best position for an individual. The quality of the optimal solution depends only on the evolutionary result of the individual. In a PSO, however, each individual has been regarded as a particle that has no volume in the D-dimensional search space, and the operators have not been used for the individuals yet. A swarm represents a number of potential solutions for the optimization problem, and each particle represents a candidate solution. The aim of a PSO is to find the optimal particle position that results in the best evaluation of the fitness function in this space. The performance of the particle, which reflects the closeness of a particle to the global optimum, is measured by using a predefined fitness function *fitness*, which encapsulates the characteristics of the optimization problem. The particle flies through this multidimensional search space at a certain velocity and adjusts its position dynamically according to its own flying experience (the positions it has occupied and its flying pathway) and that of the neighboring particles. Therefore, a particle adjusts its position toward an optimal solution that is based on the best position that the particle has found thus far and the best position of the neighborhood particles.

In a PSO, each particle *i* includes three parts: 1) x_i , the current position of the particle; 2) v_i , the current velocity of the particle; and 3) *pbest_i*, the personal best position of the particle. Associated with particle *i* is the best position that the particle has visited thus far; it is the position that yielded the highest fitness value for the particle.

$$pbest_{i}(t+1) = \begin{cases} x_{i}(t+1), \text{ if } fitness(x_{i}(t+1)) < fitness(pbest_{i}(t)) \\ pbest_{i}(t), \text{ if } fitness(x_{i}(t+1)) \ge fitness(pbest_{i}(t)) \end{cases} (1)$$

Given that a particle swarm is composed of *n* particles, every particle has a *D*-dimensional position vector $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ to indicate its position and a velocity vector $v_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ to determine its flying direction and distance. When the particle flies in a *D*-dimensional solution space, it has marked the best individual position $p_{id}=(p_{i1}, p_{i2}, \dots, p_{iD})$. The particle swarm also has marked the best group position $p_{gd}=(p_{g1}, p_{g2}, \dots, p_{gD})$. Additionally, every particle has a fitness value to evaluate itself, and the fitness value is a type of quantization error that is calculated by an optimized function.

Initially, the particles are distributed randomly, and they update themselves according to the best individual position $pbest_{id}$ and the best group position $gbest_{id}$. Then, the current optimal particle in the solution space has been found, and the position and velocity of the particle has been updated according to equations 2 and 3.

$$v_{id}(t+1) = \omega v_{id}(t) + c_1 r_1(pbest_{id}(t) - x_{id}(t)) +$$
(2)

$$c_2 r_2(gbest_{id}(t) - x_{id}(t))$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1)$$
(3)

where $i=1,2, \dots,n$; N is the total number of particles in the swarm; $gbest_{id}$ is the best position of the current group; $pbest_{id}$ is the best position of the current particle; r_1 and r_2 are random numbers that are distributed uniformly in the range (0,1); and c_1 and c_2 are learning factors (c_1 denotes the preference for the

particle's own experience, and c_2 denotes the preference for the experience of the group). Where *t* is the time of the iteration, and ω is the inertia weight factor that controls the impact of the former velocity on the current velocity. A large ω enhances the global search capability of the PSO, and a small ω enhances the local search capability. The algorithm balances the global and local search capabilities, and it decreases linearly while the iterations increase. The factor ω decreases linearly from ω_{max} to ω_{min} , and it is determined by the following equation:

$$\omega = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{iter_{\max}} \times iter$$
(4)

where ω_{max} is the maximum value of the inertial weight factor, ω_{min} is the minimum value of the inertial weight factor, *iter_{max}* is the maximum times of the iteration, and *iter* is the current iteration times. Shi and Eberhart(Shi and Eberhart, 2002) considered that the initial inertial weight factor ω is set to 0.9, and it decreased linearly from 0.9 to 0.4 while the iterations increased.

According to equation 2, the velocity is calculated based on three contributions: 1) a fraction of the previous velocity; 2) the cognitive component, which is a function of the distance of the particle from its personal best position; and 3) the social component, which is a function of the distance of the particle from the best particle found thus far.

The PSO is usually executed with the repeated application of equations 2 and 3 until the specified number of iterations has been exceeded or the velocity has no significant change. Figure 1 illustrates the flowchart of the standard PSO.



Fig. 1 Flowchart of the Standard PSO (Liang, Qin et al., 2006)

Learning from the Genetic Algorithm (GA), a crossover operator has been used in the PSO to advance the global searching capability and to avoid running into local extreme.

GA uses selection, crossover, and mutation as genetic operations to evolve the individuals. The crossover operation is the process in which the parent individual pairs the reorganization of a genetic exchange, which results in a large number of new individuals and the generation of more gifted individuals. The crossover mechanism selects the particle from all of the particles in the cross-matching pool with a certain degree of crossover probability, which has been determined beforehand and remains unchanged throughout the crossover process; it matches any two particles in the pool randomly and determines the crossover point by the crossover weight wi, which has been calculated by the fitness value of the particle, and it generates the descendant particle by the crossover operation(Hong and Ishikawa, 2008, Moradi and Abedinie, 2011). After updating the position vector x_i and the velocity vector v_i of the particle based on equations 2 and 3, the algorithm calculates a crossover with two particles, as follows:

$$child_1(x_i) = w_i \times parent_1(x_i) + (1 - w_i) \times parent_2(x_i)$$
 (5)

$$child_2(x_i) = w_i \times parent_2(x_i) + (1 - w_i) \times parent_1(x_i)$$
 (6)

$$child_{1}(v_{i}) = \frac{(parent_{1}(v_{i}) + parent_{2}(v_{i})) \times |parent_{1}(v_{i})|}{|parent_{1}(v_{i}) + parent_{2}(v_{i})|}$$
(7)

$$child_{2}(v_{i}) = \frac{(parent_{1}(v_{i}) + parent_{2}(v_{i})) \times \left| parent_{2}(v_{i}) \right|}{\left| parent_{1}(v_{i}) + parent_{2}(v_{i}) \right|}$$
(8)

The crossover process of the descendant particles is shown in equations 5, 6, 7, and 8. In contrast to the standard PSO, this improved PSO algorithm has a crossover operation that occurs after updating the position and velocity, and instead of using the parent particles, it uses the gender-ed descendant particles. The crossover operation helps descendant particles to inherit the advantages of their parent particles and maintains population diversity. Moreover, it enhances the regional search capabilities between particles and avoids the local optimum and premature convergence traps, and it obtains good results(Starke, Hendrich et al., 2017). This algorithm, which combined the PSO algorithm with a crossover operator, has been regarded as the Improved-PSO algorithm (IPSO).

2.2 Fuzzy K-Means Algorithm

In the K-means algorithm, each sample completely belongs to some cluster in iteration according to the constraint condition. In the real case, one pixel could relate to more than one class. By introducing the concept of fuzzy theory into the classification, each pixel can relate to several classes, where the percentage representation of each of the classes, within a pixel, can be represented by the corresponding membership value. Thus, a fuzzy membership mechanism should be introduced to the constraint condition and clustering criterion. The clustering partition matrix U has been changed to be a fuzzy matrix, and the membership degree function $\mu_k(x_i)$ has been improved to be a fuzzy membership degree function) by introducing the fuzzy index m. This index controls the mixture degree of the different clusters; it is an adjustable liberty parameter. This arrangement means that each sample belongs to only one cluster when m=0, and each sample could belong to several clusters when m>0. It is considered to be a soft classification when a fuzzy clustering partition matrix U is used to classify the sample set X. In the soft classification, the new clustering criterion is as follows:

The objective function J_m :

$$J_{m} = \sum_{k=1}^{c} \sum_{i=1}^{n} [\mu_{k}(x_{i})]^{m} [d(x_{i}, z_{k})]^{2}$$
(9)

The constraint condition:

$$\sum_{k=1}^{c} \mu_k(x_i) = 1, 1 \leq i \leq n; \mu_k(x_i) \in [0,1], 1 \leq i \leq n, 1 \leq k \leq c.$$
(10)

$$\mu_{k}(x_{i}) = \left[\sum_{l=1}^{c} \left(\frac{d(x_{i}, z_{k})}{d(x_{i}, z_{l})}\right)^{1/(m-1)}\right]^{-1}$$
(11)

$$z_{k} = \frac{\sum_{i=1}^{n} [u_{k}(x_{i})]^{m} x_{i}}{\sum_{i=1}^{n} [u_{k}(x_{i})]^{m}}$$
(12)

where *m* is a weighting coefficient $m \ge 1$. The larger *m* is, the more fuzzy the classification; the classification becomes a hard classification when m=1, and $U=[\mu_k(x_i)]_{n \times c}$ is the fuzzy partition matrix. As before, a good clustering should make the objective function J_c be as small as possible by having many iterations to search or the cluster centers z_k and the membership degree $\mu_k(x_i)$. Thus, during the clustering process of FKM, when each cluster center z_k is near the sample x_i with a high membership degree $\mu_k(x_i)$ to cluster k, J_m would be a minimum. In FKM, the membership function, which is also the similarity measurement, is a Euclidean distance in KM; however, it became a probability function to compare the two sample vectors. There is no direct assignment, but the membership degree of each sample to the cluster k has been used to adjust the cluster centers z_k .

2.3 IPSO-FKM Clustering Algorithm

As described above, the clustering process is a process for finding the best clustering partition in the partition space, and this process can also be referred to as an optimal problem. In IPSO clustering, a particle is referred to as a clustering partition, and the particle swarm is referred to as all of the candidate clustering partitions; the partition space is the search space, and the particle of *gbest* is referred to as the best clustering partition.

There are two pathways for combining a PSO with K-means. One pathway is using the PSO to find the best cluster center points for the K-means; the other pathway is using K-means to calculate the cluster center points and using the PSO to optimize those centroids points.

First, define the following parameters in image classification, N_b is the number of image bands, N_c is the number of clusters to be formed, N_p is the number of pixel points in the image; x_i is the vector of the *i*-th pixel point, combined with the N_b component, z_k is the vector of the *k*-th clustering centroids, *n* is the number of particles in the swarm, C_k is the subset of data vectors that form cluster *k*. In the context of clustering, given that the data set is $X=\{x_{id}, i=1,2, \dots, N_p; d=1,2, \dots, N_b\}$ and the clustering centroids set is $Z=\{z_{jk}, j=1,2, \dots, n; k=1,2, \dots, N_c\}$, a single particle *j* represents N_c cluster centers vectors, and the position vector of each particle *j* is z_j ,

$$z_{j} = (z_{j1}, \dots, z_{jk}, \dots, z_{jN_{c}})$$
(13)

A single particle represents a clustering partition, and other particles in the swarm represent a number of candidate clustering partitions. As described above, z_j corresponding to the clustering partition U_j indicates one particle *j* in the swarm, and z_{jk} is a sub-vector of the particle *j*. Each cluster center adopts a real number code; the Z then has been changed to an $N_c \times n$ one-dimensional row vector and has the membership degree $\mu_k(x_i)$. As is known, a good clustering means that the data of the same class should be as close as possible, and the vectors of each cluster's center points should be as distant as possible. According to the clustering objective function and the clustering law, the fitness function of the particle can be defined as:

$$fitness = \sum_{k=1}^{N_c} \sum_{i=1}^{N_p} [\mu_k(x_i)]^m \left\| x_i - z_k \right\|^2$$
(14)

The *fitness* measures the fitness of the particle *j* to the current position z_j .

When setting the initial search range of the PSO, the range would be too large, and the iteration would take a substantial amount of time because the range setting still relies on the human's experience and the initial positions of the particle swarm are generated randomly too far from the final iteration result. However, if the initial cluster centers could be generated by K-means, then the range should be smaller and there should be less iteration than before. Therefore, K-means has been selected to generate the initial cluster centers, and the initial swarm of PSO has been generated in the required range; the search range has been narrowed, and the iteration time has been decreased.

According to Fig. 2, the implementation procedures of the IPSO-FKM algorithm are as follows: 1) N_c pixel points were selected randomly from the input image. 2) The initial cluster centers set z_1 has been generated by K-means. These points can be regarded as one particle of the particle swarm in the IPSO. 3) n-1 particles, the remaining particles of the swarm, are initially random. 4) These particles have been optimized by updating their position and velocity in IPSO, which can produce a variety of clustering results until the fitness value achieves the termination condition or the iteration achieves the maximum number of iterations. After the optimization in IPSO, the final Z is the optimal solution for the cluster centers. 5) The cluster centers that correspond to the optimal particle are referred to the initial solution of the Fuzzy K-means, and the image can be classified in the FKM algorithm. 6) Calculate the final membership degree and the fitness value of the particle swarm to appraise the result.



Figure. 2 The procedures of the IPSO-FKM algorithm

3. EXPERIMENTAL STUDY

3.1 Materials and Study Area

Hyperion, the first space-borne hyperspectral sensor, was launched by NASA on the Earth Observing Mission 1 (EO-1) spacecraft on November 21, 2000. Hyperion acquires data in a push-broom mode, resolving 220 spectral bands (from 400 to

2500 nm) with a 30-meter resolution. The Level 1 Radiometric product has a total of 242 bands, with 70 bands in VNIR (400 to 1000 nm) and 172 bands in SWIR (900 to 2500 nm), but only 198 bands are calibrated. Because of the overlap between the VNIR and SWIR focal planes, only 196 unique channels are available. The data used in this paper is the Hyperion L1 R product provided by USGS, acquired on September 2, 2002.

The study area is located at the confluence of East Dongting Lake and the Yangtze River. Dongting Lake is the second largest lake in China; it is located in the northeast of Hunan province along the south shore of the Jin River in the middle of the Yangtze River, and it includes East Dongting Lake National Nature Reserve, West Dongting Lake City Marsh Park, and South Dongting Lake International Wetland Conservation Area. Dongting Lake is a typical area for international river, lake, and wetland research.



Figure. 3 The study area and materials

The area has a latitude that ranges from 29°22'36" to and a longitude that ranges from 113°2'36" to 29°30′48′ 113°8′46′′; it is located at the junction of river and lake areas and city and rural areas. In the study area, there are not only rivers, lakes, ponds, beaches, and marsh that all belong to nature wetlands but also crop lands and aquaculture ponds that belong to constructed wetlands, towns, and polders, which all belong to non-wetlands. The study area is very suitable for extracting and classifying information on wetlands with the hyperspectral data (Fig. 3), which can be grouped into the following six classes based on the class scheme from the Convention on Wetlands: river, lake, beach land, reed land, constructed wetlands including dry land and paddy fields, and others that belong to non-wetlands.

3.2 Initial Parameters

By using the FORTRAN language and MATLAB condition, the third dimension IPSO-FKM program for image classification

has been developed based on the principle of the PSO and Kmeans improved algorithm. Through setting the parameters and iterating many times, the initial parameters can be determined.

The initial parameters for the algorithm program include the number of clustering clusters N_c ; the number of image pixels N_p ; the number of image bands N_b ; the number of particles in the swarm *n*; the number of maximum iterations for the termination conditions *iter_{max}*; the learning factors c_1 and c_2 ; the inertial weighted factors ω_{max} and ω_{min} ; the range of velocity v_{max} ; and the fuzzy index *m*.

Parameter	Value	Parameter	Value
Nc	6	c_1, c_2	2.8, 1.3
N_p	107786	<i>iter_{max}</i>	50
N_{b}	3	V _{max}	4
п	30	ω_{max}	0.9
m	3	ω_{min}	0.4

Table 1. Setting of the initial parameters

According to Table I, the parameters of N_c , N_p , and N_b are provided by the user; the parameter learning factors, the inertial weight factor, and the fuzzy index m have been discussed before, which means that the remainder are n, *iter_{max}*, and v_{max} . Additionally, the settings of *iter*_{max} and v_{max} are determined after many experiments. With respect to the parameter n, a smaller value would go against a cosmopolitan search, but a larger value would augment the computational complexity. With respect to the range of the velocity v_{max} , this parameter can control the movement action effectively; it is also used to avoid the "explosion" phenomenon in the search procedure. The setting of v_{max} relies on a concrete situation: a smaller value for v_{max} can lead to a local optimum, and the particle finds it difficult to jump out of the local optimal point; a larger v_{max} leads to having a particle cross the global optimum, and the search procedure cannot converge effectively. Therefore, the value of v_{max} has been set to 4 after many experiments.

3.3 Procedures of the Experiment



Fig. 4 Flowchart of the experiment

In Fig. 4, the processes of the experiment contain four parts: 1) data preprocessing; 2) data down-scaling; 3) image classification by using these algorithms with a program compiled in the FORTRAN language; and 4) result analysis on the classification precision and fitness convergence.

The main pre-processing procedures include band removal, radiometric correction, strip pixels removal, and image registration. First, 87 of the 242 bands were deleted. Second, the image of 155 bands has been converted to a radiance image with the coefficient provided by the EO-1 User Guide. Third, strip pixels in the image have been corrected. Some pixels in a strip usually have a lower DN value than adjacent pixels. Thus, these strip pixels can be detected and replaced by the average of the immediately adjacent columns. Finally, the Hyperion data have been registered to the panchromatic band of the ALI data, which have already been geometrically corrected.

A hyperspectral image has hundreds of bands, and spectrally adjacent bands are usually statistically correlated. The redundant data are useless for image classification, which implies that data dimension reduction is necessary and can remove the redundant data effectively. PCA (Principal Component Analysis), MNF (Minimum Noise Fraction), and DWT (Discrete Wavelet Transform) are usually applied in data dimension reduction(Datt, Mcvicar et al., 2003). Finally, the first three MNF bands have been chosen according to the eigenvalues and the spectral characteristics of the surface materials. Then, the experimental image data that has been acquired from the RGB combination image is based on these three bands.

After the data preprocessing and down-scaling on the input data, the images with 3 bands and 107,787 pixels have been classified by these algorithms. To compare the results of the classification and to analyze the fitness convergence situation, FKM and IPSO have been selected to compare with the IPSO-FKM algorithm. Maximum Likelihood Classification (MLC) has also been used to classify the image; for the truth data, the classification result of MLC has been used to evaluate the classification accuracy of FKM, IPSO, and IPSO-FKM. Moreover, the analysis of fitness convergence is applied to the three algorithms.

4. RESULTS AND DISCUSSION

This chapter discusses the results of the application of three algorithms to remote sensing hyperspectral image classification. By comparing the classification results of Fuzzy K-means and IPSO with that of IPSO-FKM based on the truth data, we analyzed the classification precision in terms of the three aspects and the convergence speed of the new hybrid clustering algorithm. The three evaluation methods are discussed first, and then the fitness convergence and classification precision are evaluated.

4.1 Evaluation Methods

Three evaluation methods are selected to compare the results of the three clustering algorithms' application of remote sensing hyperspectral image classification. From the pixel aspect, by selecting dispersed sample data from the classified image and comparing it with the truth data acquired from the classification result of MLC and the field investigation data, the first method evaluates the overall classification accuracy and the Kappa coefficient calculated from the confusion matrix. The confusion matrix is calculated by comparing the classified image and the truth image. The overall classification accuracy p_c represents the classification correct rate for the truth data and the Kappa coefficient *kappa*, which is calculated from the confusion matrix and represents the goodness of fit to the truth data. The p_c calculates only the pixels on the diagonal of the confusion matrix, but *kappa* even considers the omission and commission error out of the diagonal of the confusion matrix. The overall accuracy and Kappa coefficient are as follows:

$$p_{c} = \frac{\sum_{i=1}^{c} X_{ii}}{N} \times 100\%$$
(15)

$$Kappa = \frac{N\sum_{i=1}^{c} X_{ii} - \sum_{i=1}^{c} X_{i+} X_{+i}}{N^2 - \sum_{i=1}^{c} X_{i+} X_{+i}}$$
(16)

where X_{ii} represents the number of pixels for which, on the *i*-th row and the *j*-th column of the confusion matrix, this number is also the number of correctly classified pixels; N represents the total number of pixels used for the accuracy assessment; c is the number of all of the columns and is also the number of categories; X_{i+} represents the number of total pixels on the *i*-th row; and X_{+i} represents the number of total pixels on the *j*-th column.

ŀ

From the feature aspect, the second method evaluates the discrepancies of the attributes of image features by calculating the inter-distance and intra-distance. The qualities of the respective clustering algorithms are measured according to the following three parameters: the quantization error as defined by the *fitness*; the intra-cluster distances, which means the distance between the data vectors within a cluster; and the inter-cluster distances, which means the distance between the centroids of the clusters. The objective of data clustering is to minimize the intra-cluster distances and to maximize the inter-cluster distances. The smaller $d_{max}(X, z_i)$ and the larger $d_{min}(z_i)$ mean that most of the classes have better classification results and better performance.

$$d_{\max}(X, z_i) = \max_{k=1,...,N_c} \{ \sum_{\forall x_i \in C_a} d(x_i, z_{jk}) / |C_{jk}| \}$$
(17)

$$d_{\min}(z_i) = \min_{\forall k_1, k_2, k_1 \neq k_2} \{ d(z_{jk_1}, z_{jk_2}) \}$$
(18)

From the aspect of information entropy, the third method evaluates the partition entropy according to the number of final cluster centers and the final membership degree matrix. The following define the partition coefficient V_{pc} and the partition entropy V_{pe} , and the larger V_{pc} and the smaller V_{pe} indicate better clustering.

$$V_{pc} = \left(\sum_{k=1}^{N_c} \sum_{i=1}^{N_p} \mu_k(x_i)^2\right) / N_p$$
(19)

$$V_{pe} = -\left(\sum_{k=1}^{Ne} \sum_{i=1}^{Np} \mu_k(x_i)^2 \log_2 \mu_k(x_i)\right) / N_p \quad (20)$$

The first method measures the classification character of the pixel, the second method measures the spatial geometry character of the pixel, and the third method measures the information quantity of the pixel. Measurement of the three aspects of the IPSO-FKM algorithm and comparison with FKM and IPSO can comprehensively evaluate the application performance of the IPSO-FKM in image classification.

4.2 Analysis of Fitness Convergence

Fig. 5 illustrates that in the fitness convergence behavior of the three algorithms, the final fitness is referred to as a quantization error (the points have been circled by the red circle in Fig. 5), which indicates the integrity of the fitness convergence in which 3.684 for FKM is the highest and 2.427 for IPSO is close to 2.295, which was obtained for IPSO-FKM. As indicated in Fig. 5, the fitness of the FKM algorithm converged after 16 iterations, the IPSO-FKM algorithm converged after 36 iterations, and the IPSO algorithm converged at 43 iterations. FKM has the fastest convergence speed but the largest quantization error, IPSO has a lower quantization error but the fitness convergence speed is too slow, and IPSO-FKM has a lower quantization error than FKM and a faster fitness convergence speed than IPSO.



4.3 Results Evaluation and Comparison

After the image classification with the three algorithms, six objects in the image were extracted, and these six objects were classified into six surface feature types based on the field investigation data. Fig. 6 illustrates the classification results of the three algorithms, which were obtained by making a comparison with the truth data (the truth data are obtained by the field investigation and MLC classification), which was directly based on the spatial distributions of the beach, reed, crop land, and other ground objects. Direct observation suggests that IPSO-FKM is better than the other methods.



Figure. 6 The classification results of the three algorithms

A total of 100 sample points were selected randomly from the truth data to compare with the results of the three algorithms; then, every measurement index of the results was calculated.

Table 2 illustrates that the IPSO-FKM algorithm is better than the others in terms of the classification accuracy and the spatial character, but it is not better than IPSO in terms of the information quantity and is not better than FKM in terms of the number of iterations.

Algorithm	FKM	IPSO	IPSO-FKM
p_c	63.19%	70.98%	72.21%
Kappa	0.5964	0.6782	0.6964
Final fitness	3.684	2.427	2.295
Maximum inter- cluster distance	12.716	8.425	7.991
Minimum intra- cluster distance	23.684	30.362	32.403
Iterations	16	43	36
V_{pc}	0.7886	0.8012	0.7929
V_{pe}	0.2803	0.2681	0.2779

Table2. The comparison of the three methods

5. CONCLUSIONS

This paper discussed the PSO and K-means algorithms, introduced the fuzzy membership function and the crossover operator into the mutation operation, and developed them to be the IPSO-FKM algorithm, which is successfully applied to wetlands information extraction and classification based on hyperspectral remote sensing image-Hyperion.

Moreover, this paper also evaluated the classification results of the experiments in terms of three aspects. In terms of the pixel aspect, the overall classification accuracy of IPSO-FKM reached 72.21%, which is better than 63.19% for FKM and 70.98% for IPSO; and the Kappa coefficient is 0.6964, which is better than 0.5964 for FKM and 0.6782 for IPSO. In the feature aspect, the maximum inter-cluster distance of IPSO-FKM is 7.991, which is smaller than 12.716 for FKM and 8.425 for IPSO; the minimum intra-cluster distance is 32.403, which is larger than 23.684 for FKM and 30.362 for IPSO. In the aspect of information entropy, the partition coefficient is 0.7929, which is not better than 0.8012 for IPSO, but the partition entropy is 0.2779, which is better than 0.2803 for FKM and 0.2681 for IPSO. Most of the evaluation indexes of IPSO-FKM are better than those of the other methods.

Additionally, this paper analyzed the fitness convergence of IPSO-FKM compared with FKM and IPSO. The number of iterations for IPSO-FKM is 36, which is smaller than 16 for FKM but larger than 43 for IPSO, and the final fitness value of IPSO-FKM is 2.295, which is better than 3.684 for FKM and 2.427 for IPSO. Thus, the analysis result of the fitness convergence suggests that the IPSO-FKM algorithm has a better searching velocity and better convergence at lower quantization errors.

Future studies will extend the fitness function to optimize the inter-cluster distance and intra-cluster distance. More elaborate tests on higher dimensional problems and large numbers of patterns will be performed. The clustering algorithms will be extended to dynamically determine the optimal number of clusters. Additionally, future research will explore the influence of the number of particles, as well as the number of iterations and other PSO parameters on the performance of the PSO.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (No. 41801235), and the Fundamental Research Funds for Beijing Universities(No. X18055).

REFERENCES

Chen, J., et al. 2009. Application of novel clonal algorithm in multiobjective optimization. *International Journal of Information Technology & Decision Making*, 9(2), 28-41.

Civicioglu, P. and E. Besdok 2013. A conceptual comparison of the Cuckoo-search, particle swarm optimization, differential evolution and artificial bee colony algorithms. *Artificial Intelligence Review*, 39(4), 315-346.

Datt, B., et al. 2003. Preprocessing EO-1 Hyperion hyperspectral data to support the application of agricultural indexes. *Geoscience & Remote Sensing IEEE Transactions on*, 41(6), 1246-1259.

Ding, S., et al. 2014. Optimization of well placement by combination of a modified particle swarm optimization algorithm and quality map method. *Computational Geosciences*, 18(5), 747-762.

Hong, Z. and M. Ishikawa 2008. Evolutionary Particle Swarm Optimization: A Metaoptimization Method with GA for Estimating Optimal PSO Models.

Jain, N. K., et al. 2018. A Review of Particle Swarm Optimization. *Journal of the Institution of Engineers*, 2(3), 1-5.

Kennedy, J. and R. Eberhart 1995. Particle swarm optimization. *Icnn95-international Conference on Neural Networks*.

Liang, J. J., et al. 2006. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Transactions on Evolutionary Computation*, 10(3), 281-295.

Moradi, M. H. and M. Abedinie 2011. A combination of Genetic Algorithm and Particle Swarm Optimization for optimal DG location and sizing in distribution systems. *Ipec, Conference.*

Shi, Y. and R. C. Eberhart 2002. Empirical study of particle swarm optimization. *Congress on Evolutionary Computation*.

Starke, S., et al. 2017. An efficient hybridization of Genetic Algorithms and Particle Swarm Optimization for inverse kinematics. *IEEE International Conference on Robotics & Biomimetics*.