Science in China Series D: Earth Sciences

Swarm intelligence for classification of remote sensing data

LIU XiaoPing[1†], LI Xia[1†], PENG XiaoJuan[2], LI HaiBo[1] & HE JinQiang[1]

[1] School of Geography and Planning, Sun Yat-sen University, Guangzhou 510275, China;
[2] South China Sea Environment Monitor Center, Guangzhou 510300, China

This paper proposes a new method to classify remote sensing data by using Particle Swarm Optimization (PSO). This method is to generate classification rules through simulating the behaviors of bird flocking. Optimized intervals of each band are found by particles in multi-dimension space, linked with land use types for forming classification rules. Compared with other rule induction techniques (e.g. See5.0), PSO can efficiently find optimized cut points of each band, and have good convergence in the search process. This method has been applied to the classification of remote sensing data in Panyu district of Guangzhou with satisfactory results. It can produce higher accuracy in the classification than the See5.0 decision tree model.

swarm intelligence, particle swarm optimization (PSO), remote sensing

1 Introduction

Image classification is a fundamental process in remote sensing applications, which is to extract useful geographic information from raw image data[1]. Conventional classification methods include minimum distance from means, maximum-likelihood, cluster analysis and Bayesian classification[2,3], which prove to be simple and useful. However, these approaches are all based on statistical principles, and require training data following a normal distribution. Training samples and model parameters will directly affect the overall quality of classification[4]. Recently, numerous new methods for remote sensing classification have been developed, such as machine learning[5,6], support vector machine (SVM)[7,8], neural network[9,10], fuzzy set[11] and genetic algorithm[12], These methods may have higher accuracies than conventional classifiers. However, there is still considerable scope for further increases in classification accuracies so that the results can satisfy most of the applications[13]. Thus, it is still a key topic in remote sensing for exploring new methods to increase classification accuracies.

Recently, Artificial Intelligence (AI) techniques have been increasingly incorporated in the classification of remote sensing images[14]. As a bottom-up approach, Swarm Intelligence (SI) is actually a complex multi-agents system, consisting of numerous simple individuals (e.g., ants, birds, etc.), which exhibit their swarm intelligence through cooperation and competition among the individuals. Although there is typically no centralized control dictating the behavior of the individuals, the accumulation of local interactions in time often gives rise to a global pattern, SI has currently become a hot topic in artificial intelligence research, and it has succeeded in solving problems such as traveling salesman problems, data clustering, combination optimization, network routing, rule induction, and pattern recognition[15−20]. However, using SI in remote sensing classification is a fairly new research area and needs much more work to do.

Received May 21, 2007; accepted July 18, 2007
doi: 10.1007/s11430-007-0133-6
†Corresponding author (email: lixia@mail.sysu.edu.cn)
Supported by the National Outstanding Youth Foundation of China (Grant No. 40525002), the National Natural Science Foundation of China (Grant No. 40471105), and the Hi-tech Research and Development Program of China (863 Program) (Grant No. 2006AA12Z206).

SI mainly involves two algorithms, i.e., particle swarm optimization (PSO) and ant colony optimization (ACO). In this paper we try to introduce PSO into remote sensing image classification. Particle swarm optimization (PSO), a new population-based evolutionary computation technique inspired by social behavior simulation, was first introduced in 1995 by Kennedy and Eberhart[21]. PSO is an efficient and effective global optimization algorithm, which has been widely applied to nonlinear function optimization[22]. Complex society behavior can be well simulated and explained by PSO, which is effective to solve complex optimization problems[22]. Compared with evolution algorithms, PSO reserves the global searching strategy based on community; avoiding complex genetic operators with a simple speed-offset model; tracing current searching situation and tuning the strategy when necessary for strong memory, which makes PSO have powerful global convergence and stronger robustness[22].

This paper proposes a new method to classify remote sensing data based on particle swarm optimization classifiers (PSO-Miner). Classification rules were designed through simulating the behaviors of bird flocking. Optimized intervals of each band were found by particles in multi-dimension search space, linked with land use types for forming classification rules. Training data are removed by a sequence covering algorithm. If the remaining pixels of certain land use type are less than a threshold value, rule induction will stop, and go to next land use type. This procedure continues until all the land use types have been examined.

2 Particle swarm optimization

The PSO is a population-based optimization technique, where the population is called a swarm. A swarm of individuals (called particles) fly through the multi-dimensional search space with a velocity, which is constantly updated by the particle's own experience and the experience of the particle's neighbors or the experience of the whole swarm[21]. Each particle represents a candidate solution to an optimization problem. Particles are initialized with random positions and velocities, the optimum is searched by iteration processes. During each iteration, the particle is updated through tracking two "extrema". The former is the best position found by the particle by far, which is called the personal best (*pbest*). The latter is the position with the highest fitness value in the entire run, which is called the global best (*gbest*). Each particle searches in the problem space by tracking the particle that has reached its best position, until the best position for itself can be located[21].

As shown in Figure 1, the particle performs an optimized searching behavior in a two-dimensional search space. Here X_t denotes the position of a particle during its previous iteration, X_{t+1} denotes the position of a particle right after the previous iteration; the velocity of a particle is the sum of three velocity vectors, i.e., V_t is velocity during the previous iteration, V_{pbest} is velocity of a particle at its personal best, V_{gbest} is velocity of a particle at the global best. Under the combined influence of V_t, V_{pbest} and V_{gbest}, the particle will move to its new position at X_{t+1} with velocity V_{t+1}, getting closer to the best position. As iterations continue, a particle would be approaching the best position, and would eventually locate the best position or the best possible position.

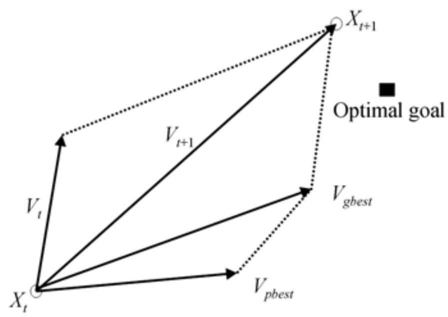

Figure 1 An illustration of optimized searching behavior of particle in a two-dimensional search space.

3 Remote sensing classification based on PSO

Each particle has a memory, remembering the best position of the search space it has ever visited, which makes PSO have powerful global convergence and stronger robustness[22]. As a result, PSO can be efficiently used to solve nonlinear problems, and it is particularly suitable for use in complex remote sensing classification. By taking TM image as an example in this study, this paper introduces a new method to classify remote sensing data based on particle swarm optimization classifiers (PSO-Miner).

PSO has been succeeded in solving problems such as traveling salesman problems (TSP), data clustering, combination optimization and pattern recognition[15,16,17,20]. However, the studies on classification rule induction using PSO are still relatively unexplored.

Sousa et al. first developed a binary-encoding PSO-based rule induction algorithm in 2003[23]. In this paper, we propose a PSO-based rule induction algorithm (PSO-Miner) for remote sensing classification, which adopts a real-encoding way[24]. In comparison with other rule classifiers, PSO-Miner can be applied to both discrete and continuous attributes. In PSO-Miner algorithm, classification rules were designed through simulating the behaviors of bird flocking. Each particle corresponds to a route, Optimized intervals of each band were found by particles in multi-dimension space, linked with land use types for forming classification rules (Figure 2). The rule format is described as

$$\begin{aligned} &\text{IF} \quad band\ 1 = Value\_1 \\ &\quad \text{AND} \quad band\ 2 = Value\_2 \\ &\quad \quad \vdots \\ &\quad \text{AND} \quad band\ j = Value\_j \\ &\quad \quad \text{THEN} \quad Class\_x \end{aligned}$$

For the PSO-Miner algorithm each rule is treated as a volumeless particle in D-dimensional searching space. In this paper the best zone $[x_-, x_+]$ of a band is defined for each rule, here x_- denotes the lower threshold of the best zone while x_+ denotes the upper threshold of the best zone (Figure 2). Since the lower and upper thresholds for a zone occur in pairs, D=2n for remote sensing data with n bands. Suppose there are m particles, the position of the ith particle is represented as $(x_{-i1}, x_{+i1}, x_{-i2}, x_{+i2}, \cdots, x_{-in}, x_{+in})$, while the velocity of the ith particle is represented as $(v_{-i1}, v_{+i1}, v_{-i2}, v_{+i2}, \cdots, v_{-in}, v_{+in})$. A particle will successively adjust its velocity and position in accordance with the current personal best $p(t)$ and global best p_g during its flying process. The position at next iteration is calculated according to the following equations[21]:

$$\begin{cases} v_{-ij}(t+1) = w(t)v_{-ij}(t) + c_1 r_{1ij}(t)(p_{-ij}(t) \\ \qquad - x_{-ij}(t)) + c_2 r_{2ij}(t)(p_{g-} - x_{-ij}(t)) \\ v_{+ij}(t+1) = w(t)v_{+ij}(t) + c_1 r_{3ij}(t)(p_{+ij}(t) - x_{+ij}(t)) \\ \qquad + c_2 r_{4ij}(t)(p_{g+} - x_{+ij}(t)) \end{cases}, \tag{1}$$

$$\begin{cases} x_{-ij}(t+1) = x_{-ij}(t) + v_{-ij}(t+1) \\ x_{+ij}(t+1) = x_{+ij}(t) + v_{+ij}(t+1) \end{cases}, \tag{2}$$

where I=1, 2, ⋯, , m; j=1, 2, ⋯, n; t is the number of iterations; $w(t)$ is the inertia weight, which is employed to control the influence of the previous history of velocities on the current velocity. A larger inertia weight $w(t)$ facilitates globe exploration, while a smaller inertia weight $w(t)$ facilitates local exploration; c_1 and c_2 are two positive constants, called cognitive learning rate and social learning rate respectively; r_{1ij}, r_{2ij}, r_{3ij} and r_{4ij} are random values varying in the range [0,1]; $p_{-ij}(t)$ and $p_{+ij}(t)$ are the best fitness values for the lower threshold and the upper threshold, respectively, as searched by the ith particle up to the moment; p_{g-} and p_{g+} are the best fitness values for the lower threshold and the upper threshold, respectively, as searched by the entire particle swarm up to the moment, and $p_{g-} = (p_{g-1}, \ p_{g-2}, \cdots p_{g-n})$, $p_{g+} = (p_{g+1}, \ p_{g+2}, \cdots p_{g+n})$.

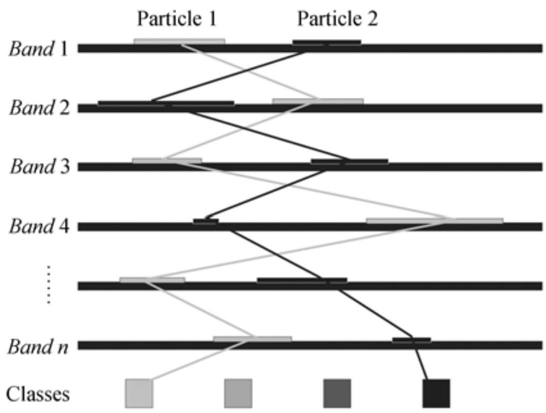

Figure 2 The principle of remote sensing classification based on PSO-Miner.

Remote sensing classification based on PSO-Miner can be divided into three stages, i.e., rule construction, rule evaluation and covering algorithm of training set.

3.1 Rule construction

Rule construction stimulates the behaviors of bird flocking. Each particle searches the upper threshold and the lower threshold of the best value at each band. For remote sensing data with n bands, particles search the best value in 2n-dimensional space. The best value zone at each band can be connected with one another using the operator ‘And’, linked with land use types for forming classification rules.

Initially, particles are randomly distributed in a 2n-dimensional space, the initial position values of particle can be expressed as

$$\begin{cases} x_{-ij} = Rand * (band_{j\max} - band_{j\min}) + band_{j\min} \\ x_{+ij} = Rand * (band_{j\max} - band_{j\min}) + band_{j\min} \end{cases}, \tag{3}$$

where x_{-ij} and x_{+ij} represent the upper threshold and the lower threshold of the ith particle at the jth band respectively, and if $x_{-ij}>x_{+ij}$, they will be reciprocally replaced. *Rand* is a random value varying in the range of [0,1]. $band_{j\min}$ and $band_{j\max}$ are respectively the maximum value and the minimum value of the jth band. The initial velocity of a particle can be expressed as

$$\begin{cases} v_{-ij} = Rand * v_{-j}^{\max} \\ v_{+ij} = Rand * v_{+j}^{\max} \end{cases}, \tag{4}$$

where $v_{-j}^{\max}$ and $v_{+j}^{\max}$ are respectively the maximum velocity values in the upper threshold direction and in the lower threshold direction.

After each iteration run, the fitness value for each particle is calculated, and the current fitness value is compared with the individual optimum for each particle prior to iteration. If the current fitness value is better than the individual optimum for a particle prior to iteration, then the individual optimum will be updated, otherwise, the individual optimum will not be updated. After individual optimum is calculated for all particles, the best individual optimum becomes the global optimum. Thereafter, the inertia weight in eq. (1) is updated according to the following equation:

$$w(t) = w_{\max} - t \cdot (w_{\max} - w_{\min}) / I_{\max}, \tag{5}$$

where t denotes the number of iterations, $w_{\max}$ is the maximum inertia weigh, $w_{\min}$ is the minimum inertia weigh, $I_{\max}$ is the predefined maximum number of iterations. Based on the current individual optimum, global optimum as well as inertia weight for a particle, eqs. (1) and (2) can be used to upgrade the flying velocity and position for each particle. When the absolute values for global optimum fitness and average fitness are smaller than a threshold, or the iteration number exceeds $I_{\max}$, the cycling is terminated, and a set of classification rules are generated. The pseudo-code for classification rules construction is as follows (Table 1).

3.2 Rule evaluation

The fitness of a classification rule (particle) can be used to evaluate the position of a particle, and to make judgment about its flying direction. Therefore, a reasonable choice of fitness function is importance for problem solving. The classification rule (particle) evaluation can be calculated using the following equation:

$$Q = \left(\frac{\text{TruePos}}{\text{TruePos} + \text{FalseNeg}} \right) \cdot \left(\frac{\text{TrueNeg}}{\text{FalsePos} + \text{TrueNeg}} \right), \tag{6}$$

where TruePos is the number of cases covered by the rule that have the class predicted by the rule; FalsePos is the number of cases covered by the rule that have a class different from the class predicted by the rule; FalseNeg is the number of cases that are not covered by the rule but that have the class predicted by the rule; TrueNeg is the number of cases that are not covered by the rule and that do not have the class predicted by the rule. The larger the value of Q is, the higher the quality of the rule becomes.

Table 1

```
Input training data
Initialize particle swarm
While (t<Max_ iteration or err>Min_error_criteria)
  For i = 1 to No_of_particles
      Calculate particle's down_ fitness
      If the down_fitness value is better than the best down_fitness value (p-best) in history
      Then
          Set current value as the new p-best
      End if
      Choose the particle with the best down_fitness value of all the particles as the g-best
      Calculate particle's upper_ fitness
      If the upper _fitness value is better than the best upper_fitness value (p+best) in history
      Then
          Set current value as the new p+best
      End if
      Choose the particle with the best upper_fitness value of all the particles as the g+best
      Update particle velocity according to eq. (1)
      Update particle position according to eq. (2)
  Next i
Loop
```

3.3 Covering Algorithm on training set

The covering algorithm is basically a divide-and- conquer technique. Being given an instance training set, it runs the rule induction algorithm in order to obtain the highest quality rule for the predominant class in the training set[23].

The best position p_g for a particle as searched by using the algorithm (the best classification rule) is put into the rule set R, then sequence covering algorithm is employed to remove correctly classified instances from the training set. The rule induction algorithm is run once more, iteratively a ordered rule set is built. The covering algorithm runs until only a pre-defined number of instances are left to classify.

4 Classification experiment and results

A satellite Landsat TM image in Panyu district of Guangzhou acquired on July 18, 2004 was used for the experiment of classification using this PSO-Miner method. The study area has a size of 1666×2211 pixels with a ground resolution of 30 m (Figure 3). Selection of training samples is a key step for remote sensing classification because these samples influence the quality of a discovered rule. Based on field investigation and land use maps, sample data were acquired by using a hierarchically random sampling method. A total of 4120 samples were obtained, which were divided into two groups –2120 as the training data set, and 2000 as the test data set.

Figure 3 TM image (5, 4, 3) in the study area of Guangzhou.

The PSO-Miner classification model involves a two step process: extraction of classification rules and recognition of land use types from remote sensing images. Classification rule is generally discovered from training data using PSO-Miner algorithm, which is developed using Visual Basic 6.0 language. Land use types are obtained by applying these rules discovered by the Ant-Miner to the classification of remote sensing images, and is also developed using Visual Basic 6.0 language.

When PSO is employed for extraction of classification rules, parameters shall be given their pre-defined values. In this study, the pre-defined values for all the parameters are listed in Figure 4. Because the best upper threshold and the best lower threshold shall be predefined for each band among the 6 TM bands (including bands 1－5 and band 7), the optimized searching by a particle actually proceeds in D=2×6-dimensional space, where there are 8 classes for remote sensing classification. Here the population size for each class (particle number, *Numb*) is set at 20, and the maximum velocity for a particle ($v_{\max}$) is set at 10. As has already been shown from the study results by Shi et al.[25], big inertia weight is favorable for prevention of local optimum, while small inertia weight is favorable for algorithm convergence. As a result, inertia weight can be set to be linearly decreased with time, with the maximum inertia weight ($w_{\max}$) set at 0.9, and the minimum inertia weight ($w_{\min}$) set at 0.4. When the iteration number exceeds 100, the recycle will be forced to terminate. When the number of remaining instances for each class is smaller than 5, rule discovery will be terminated for this type of data, but will proceed for the next type of data. The weight for PSO's learning rate also seriously affects the performance of the algorithm. The cognitive learning rate of particles is expressed by c_1, and c_1=0 means that the par-

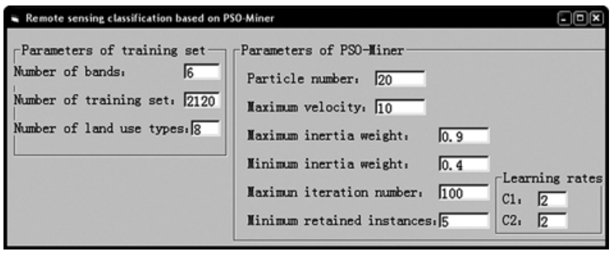

Figure 4 The parameters of PSO-Miner for remote sensing classification.

ticle shows no cognitive function, a particle is capable of entering a new searching space under the interaction of particles, but tends to get into local optimum. The social learning rate of particles is expressed by c_2, and $c_2=0$ means no such information sharing among the particles, hence low probability in obtaining an globe optimum. The learning rates were $c_1=c_2=2$ in this experiment.

Based on the training data selected previously, PSO-Miner algorithm was run and 40 classification rules were obtained. A selected set of the classification rules is listed in Table 2. Figure 5(a) is the classified remote sensing image based on PSO-Miner.

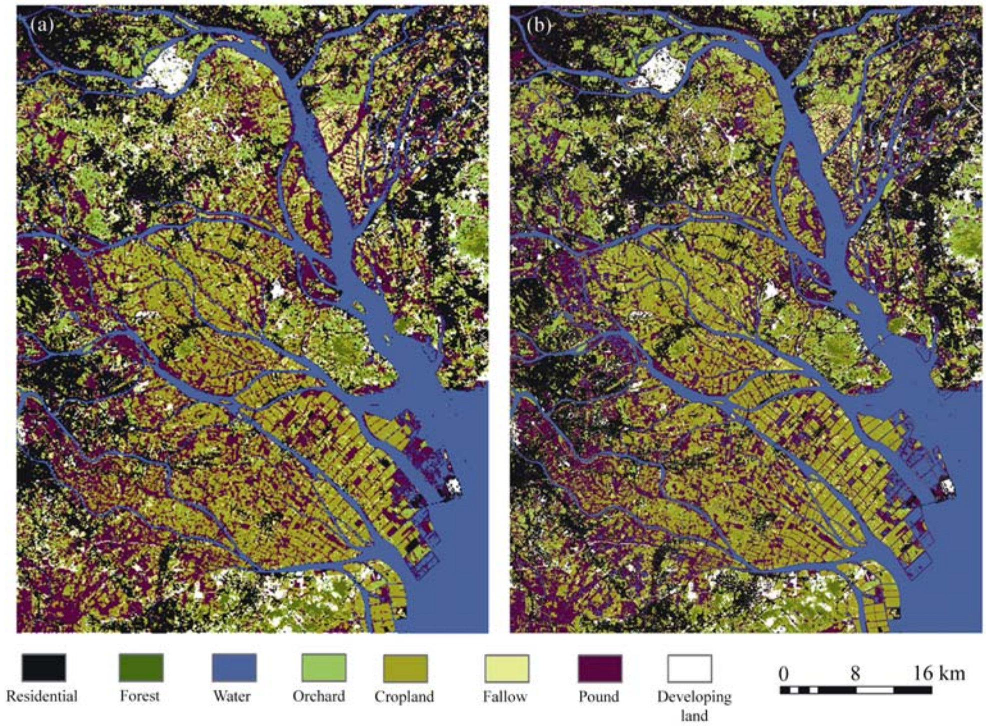

Figure 5 Land use classification in the study area of Panyu district.

Table 2 Selected classification rules by using PSO-Miner

Rule 1
IF
$96.7<B_1<141.7$ & $48.9<B_2<73.9$ & $64.5< B_3 <118.4$ & $81.4< B_4 <103.3$ & $110.8< B_5 <150.3$& $46.9< B_7 <98.4$
Then
class=Urban

Rule 2
IF
$84<B_1<89.7$ & $37<B_2<48.2$ & $42< B_3 <62.9$ & $20< B_4 <41.2$ & $12.8< B_5 <45.3$& $4.4< B_7 <23.9$
Then
class=Water

Rule 3
IF
$77.4<B_1<90.8$ & $34.9<B_2<44.8$ & $36.5< B_3 <56.7$ & $81.4< B_4 <140$ & $59.3< B_5 <97$& $18.4< B_7 <46.5$
Then
class=Agriculture

Rule 4
IF
$107.5<B_1<152.6$ & $58.9<B_2<105.6$ & $89.7< B_3 <160.5$ & $85.8< B_4 <131.4$ & $137.7< B_5 <238.1$& $64.2< B_7 <126.4$
Then
class=Developing land
……

The results from this PSO-Miner method are compared with those from a decision tree method. The decision tree method is to reconstruct classification rules automatically by using some machine learning techniques, such as the See 5.0 system. The See 5.0 system is based on the ‘information gain ratio’ to determine the splits at each internal node of the decision tree.

In the comparison, the same training data (2120 samples) were used for the classification and the same test data (2000 samples) were used for validation. The classification result of the decision tree method using the See 5.0 system is shown in Figure 5(b). The comparison between Figure 5(a) and (b) indicates that the PSO method is better than the decision tree method. An enlarged part of the study area is shown in Figure 6.

Figure 6(a) and (d) are the original TM image. The visual interpretation indicates that some ponds were incorrectly classified as water by the See 5.0 method in Figure 6(c) and (f), but they were correctly classified by this PSO algorithm. Moreover, this proposed method produces more homogeneous patterns than the See 5.0 method. The former can generate the results more similar to those of traditional land use mapping.

As shown in Table 2, the total accuracy is 84.6% by using this PSO-Miner method. The ponds in the study area have been identified successfully. In contrast, as shown in Table 3, a lower total accuracy (81.8%) is obtained by using the See 5.0 method. The total accuracy has a bias because of the difference between the actual agreement and chance agreement, which can be effectively explained with the Kappa coefficient[26]. As a result, more meaningful results will be yielded through comparing the Kappa coefficient for remote sensing classification. The Kappa coefficient is calculated as follows:

$$\text{Kappa} = \frac{M\sum_{i=1}^{r} x_{ii} - \sum_{i=1}^{r}(x_{i+}\cdot x_{+i})}{M^2 - \sum_{i=1}^{r}(x_{i+}\cdot x_{+i})}, \tag{7}$$

where x_{ii} is the elements on the main diagonal of the error matrix, x_{i+} is the sum of the ith row of the error matrix, x_{+i} is the sum of ith column of the error matrix. According to eq. (7), the Kappa coefficients for this two classification algorithms were calculated (Tables 3 and 4). The Kappa coefficient of PSO-Miner method is 0.821, a lower Kappa coefficient (0.788) is obtained by using See 5.0 Obviously, the PSO-Miner classification results are better than the See 5.0-based classification results.

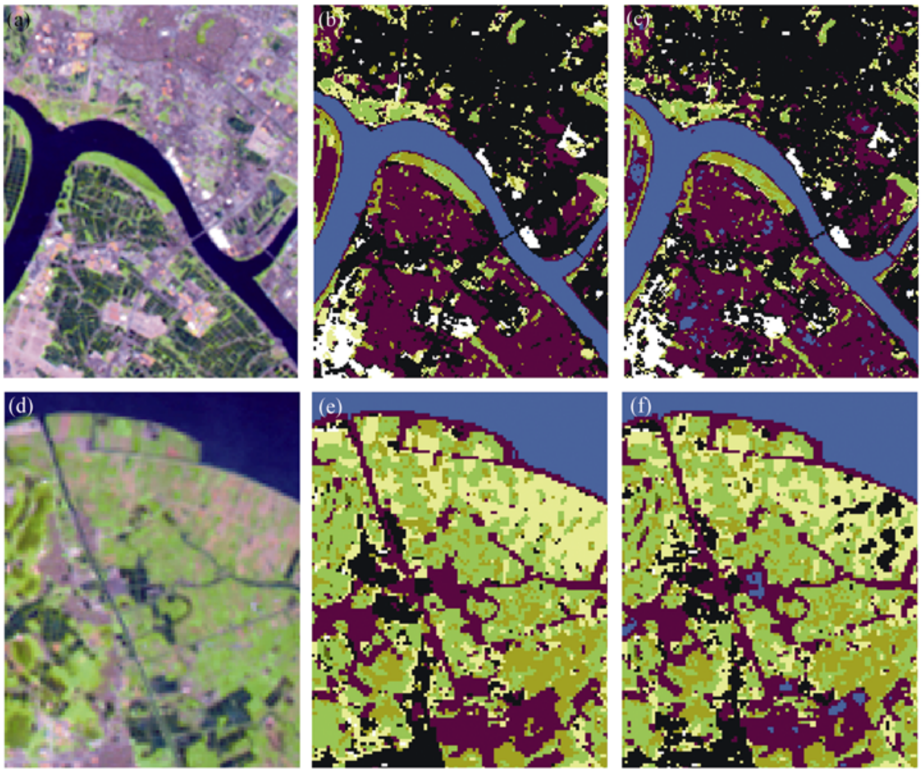

Figure 6 Land use classification in the local enlargement area of Panyu district.

Table 3 Confusion matrix of classification for Panyu district by using PSO-Miner

Actual	Class									
	Residential	Forest	Water	Orchard	Cropland	Fallow	Pond	Developing land	Total	User's accuracy (%)
Residential	288	1	2	1	8	19	3	7	329	87.6
Forest	1	98	1	9	5	0	0	1	115	85.2
Water	1	3	313	2	2	1	28	1	351	89.2
Orchard	9	10	3	184	35	4	2	0	247	74.5
Cropland	1	4	2	31	259	7	1	2	307	84.4
Fallow	10	1	1	6	4	107	1	1	131	81.3
Pond	11	3	37	1	0	4	285	1	342	83.4
Developing land	10	1	0	3	2	3	1	158	178	88.7
Total	331	121	359	237	315	145	321	171	2000	
Producer's accuracy (%)	87.0	81.0	87.2	77.6	82.2	73.8	88.7	92.4		
				Total accuracy =84.6%		Kappa coefficient =0.821				

Table 4 Confusion matrix of classification for Panyu district by using See 5.0

Actual	Class									
	Residential	Forest	Water	Orchard	Cropland	Fallow	Pond	Developing land	Total	User's accuracy (%)
Residential	281	2	3	0	9	22	5	7	329	85.4
Forest	1	96	0	8	8	0	1	1	115	83.4
Water	2	4	304	3	3	1	33	1	351	86.6
Orchard	7	12	2	183	36	6	1	0	247	74.1
Cropland	1	5	2	31	258	8	1	1	307	84.0
Fallow	13	2	0	5	5	104	1	1	131	79.4
Pond	14	2	60	3	1	6	254	2	342	74.3
Developing land	13	1	1	5	2	1	0	155	178	87.1
Total	332	124	372	238	322	148	296	168	2000	
Producer's accuracy (%)	84.6	77.4	81.7	76.9	80.1	70.3	85.8	92.3		
				Total accuracy =81.8%		Kappa coefficient =0.788				

5 Conclusions

Intelligent classification is a hot topic in remote sensing study. Traditional approaches of classification may have some limitations in constructing proper classifiers when the study area is complex. Therefore, it is necessary to introduce intelligent methods to improve the accuracy of classification. This paper has presented a new method to classify remote sensing data by using this PSO-Miner algorithm. PSO is actually a complex multi-agents system, which exhibits their swarm intelligence by cooperation and competition among the simple individuals, and the entire problem solving would not be affected by failure of one or several intelligent individuals. In contrast to evolution algorithm, PSO adopts a simple velocity-displacement model and avoids complex genetic operations, while its memorizing capability enables it to track the current searching conditions, to make dynamic adjustments concerning its searching strategies, and to behave strong global convergence and robustness. Therefore, PSO is particularly suitable for extraction of complex geographical rules.

Classification rules can be constructed with PSO-Miner algorithm through simulating the behaviors of bird flocking, Training data covered by the discovered rule can be removed using sequence covering algorithm, and this process will proceed until all rule discovery is completed. The PSO-Miner algorithm proves to be particularly suitable for processing continuous data, since particles can automatically search the best divisions of each attribute. Furthermore, the If-Then classification rules discovered by this PSO algorithm can describe the complex relationship more conveniently and more comprehensible than mathematical equations.

The PSO-Miner algorithm has been applied to the classification of remote sensing images of Panyu district. The comparison of classification accuracies is carried out between this PSO-Miner and the See 5.0 method. It is found that this PSO-Miner method has the total accuracy of 84.6% and the Kappa coefficient of 0.821. The decision tree method has the accuracy of 81.8% and the Kappa coefficient of 0.788. It clearly indicates that this PSO-Miner method has a better accuracy than the decision tree method.

1 Wilkinsom G G. Results and implications of a study of Fifteen years of satellite image classification experiments. IEEE T Geosci Remote Sens, 2005, 43(3): 433－440[DOI]
2 Strahler A H. The use of prior probabilities in maximum likelihood classification of remotely sensed data. Remote Sens Environ, 1980, 10: 135－163[DOI]
3 Aksoy S, Koperski K, Tusk C, et al. Learning Bayesian classifiers for scene classification with a visual grammar. IEEE T Geosci Remote Sens, 2005, 43(3): 581－589[DOI]
4 Wilkinsom G G. A Review of Current issue in the integration of GIS and Remote sensing data. Int J Geogr Inf Sci, 1996, 10(1): 85－101[DOI]
5 Pal M, Mather P M. An assessment of the effectiveness of decision tree methods for land cover classification. Remote Sens Environ, 2003, 86: 554－565[DOI]
6 Friedl M A, Brodley C E. Decision tree classification of land cover from remotely sensed data. Remote Sens Environ, 1997, 61: 399－409[DOI]
7 Huang C, Davis L S, Townshend J R G. An assessment of support vector machines for land cover classification. Int J Remote Sens, 2002, 23: 725－749[DOI]
8 Melgani F, Bruzzone L. Classification of hyperspectral remote sensing images with support vector machines. IEEE T Geosci Remote Sens, 2004, 42: 1778－1790[DOI]
9 Frizzelle B G, Moody A. Mapping continuous distributions of land cover: A comparison of maximum-likelihood estimation and artificial neural networks. Photogramm Eng Remote Sens, 2001, 67(6): 693－705
10 Heerman P D, Khazenie N. Classification of multispectral remote sensing data using a back propagation neural network. IEEE T Geosci Remote Sens, 1992, 30(1): 81－88[DOI]
11 Wang F. Fuzzy supervised classification of remote sensing images. IEEE T Geosci Remote Sens, 1990, 28(2): 194－201[DOI]
12 Bandyopadhyay S, Pal S. Pixel classification using variable string genetic algorithms with chromosome differentiation. IEEE T Geosci Remote Sens, 2001, 39(2): 303－308[DOI]
13 Foody G M, Mathur A. A relative evaluation of multi class image classification by support vector machines. IEEE T Geosci Remote Sens, 2004, 42(6): 1335－1343[DOI]
14 Stathakis D, Vasilakos A. Comparison of computational intelligence based classification techniques for remotely sensed optical image classification. IEEE T Geosci Remote Sens, 2006, 44(8): 2305－2318
15 Machado Th R, Lopes H S. A Hybrid Particle Swarm Optimization Model for the Traveling Salesman Problem. Adaptive and Natural Computing Algorithms. New York: Springer, 2005. 255－258
16 Kennedy J. Stereotyping: improving particle swarm performance with cluster analysis. In: Proceedings of the Congress on Evolutionary Computing. Piscataway: IEEE Service Center, 2000. 1507－12
17 Ting T O, Rao M V C, Loo C K, et al. Solving unit commitment problem using hybrid particle swarm optimization. J Heuristics, 2003, 9(6): 507－520[DOI]
18 Kwang M S, Weng H S. Multiple ant-colony optimization for network routing. In: Proceedings of the First International Symposium on Cyber Worlds. 2002. 277－281
19 Liu X P, Li X, Yeh A G O, et al. Discovery of transition rules for geographical cellular automata by using ant colony optimization. Sci China Ser D-Earth Sci, 2007, 50(10): 1578－1588
20 Omran M. Particle Swarm Optimization methods for pattern Recognition and Image Processing. Dissertation for the Doctoral Degree. University of Pretoria, 2005
21 Kennedy J, Eberhart R C. Particle swarm Optimization. In: Proc. IEEE, International Conference on Neural Networks. Piscataway. New Jersey. 1995. 1942－1948
22 Kennedy J, Eberhart R C, Shi Y. Swarm Intelligence. San Francisco: Morgan Kaufman Publishers, 2001
23 Sousa T, Neves A, Silva A. A particle swarm data miner. In: 11th Portuguese Conference on Artificial Intelligence, Workshop on Artificial Life and Evolutionary Algorithms, 2003, 43－53
24 Liu Y, Qin Z, Shi Z W, et al. Rule discovery with particle swarm optimization. Lect Notes Comput Sci, 2004, 291－296
25 Shi Y H, Eberhart R C. Parameter selection in particle swarm optimization. The Seventh Annual Conference on Evolutionary Programming. Washington DC. 1998. 591－600
26 Congalton R G. A review of assessing the accuracy of classification of remotely sensed data. Remote Sens Environ, 1991, 37: 35－46[DOI]