Computers & Geosciences 89 (2016) 252–259

Contents lists available at ScienceDirect

Computers & Geosciences

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

Research paper

Novel classification method for remote sensing images based on information entropy discretization algorithm and vector space model

Li Xie*, Guangyao Li, Mang Xiao, Lei Peng

College of Electronics and Information Engineering, Tongji University, Shanghai, China

A R T I C L E I N F O

Article history:
Received 19 March 2015
Received in revised form
29 October 2015
Accepted 16 December 2015
Available online 28 December 2015

Keywords:
Classification
Remote sensing image
Information entropy discretization algorithm
VSM

A B S T R A C T

Various kinds of remote sensing image classification algorithms have been developed to adapt to the rapid growth of remote sensing data. Conventional methods typically have restrictions in either classification accuracy or computational efficiency. Aiming to overcome the difficulties, a new solution for remote sensing image classification is presented in this study. A discretization algorithm based on information entropy is applied to extract features from the data set and a vector space model (VSM) method is employed as the feature representation algorithm. Because of the simple structure of the feature space, the training rate is accelerated. The performance of the proposed method is compared with two other algorithms: back propagation neural networks (BPNN) method and ant colony optimization (ACO) method. Experimental results confirm that the proposed method is superior to the other algorithms in terms of classification accuracy and computational efficiency.

1. Introduction

Classification is typically regarded as one of the most essential processes in remote sensing image processing (Wilkinson, 2005). There are two key aspects of classification, classification accuracy and computational efficiency. A considerable number of methods have been researched for remote sensing image classification over the last four decades, including statistical methods (Strahler, 1980; Benediktsson et al., 1990; Bruzzone et al., 1999; Kerroum et al., 2010), decision trees (DT) (Friedl and Brodley, 1997; McIver and Friedl, 2002; Pal and Mather, 2003; Xu et al., 2005), artificial neural networks (ANN) (Heermann and Khazenie, 1992; Bischof et al., 1992; Miller et al., 1995; Kavzoglu, 2009; Han et al., 2012), and other artificial intelligence algorithms (Tso and Mather, 1999; Melgani and Bruzzone, 2004; Liu et al., 2008; Zhong and Zhang, 2012; Yang et al., 2012; Pal et al., 2013). However, these approaches have limitations in solving classification problems.

Statistical methods such as maximum likelihood classifier and Bayesian classifier rely on the assumption that the members of each class follow a normal distribution in the feature space (Benediktsson et al., 1990; Bruzzone et al., 1999; Pal and Mather, 2003). The classification accuracy decreases when the dimension of features increases. DT methods make no assumptions concerning the distribution of the input features and are thus robust in managing the nonlinear relationship among the class members (Friedl and Brodley, 1997; McIver and Friedl, 2002). However, the efficiency of DT methods is always influenced by the size of the feature space. Pal and Mather (2003) indicated that DT methods are not recommended for a data set with a high-dimensional feature space. Ant colony optimization (ACO) methods have been demonstrated to work with remote sensing image classification in recent years (Dai and Liu, 2007; Liu et al., 2008). ACO methods can produce more succinct decision rules and improve the precision of classification compared to DT methods. Similar to DT methods, ACO methods are not appropriate for a high-dimensional feature space. ANN methods demonstrate superior performance with a high-dimensional feature space compared to statistical methods because they are distribution-free (Benediktsson et al., 1990; Heermann and Khazenie, 1992; Shao and Lunetta, 2012). However, ANN models require a significant amount of training data and a considerable number of iterative training procedures to ensure that the models are trained successfully. Consequently, the computation of ANN algorithms can be extraordinarily complex.

To resolve the problems in terms of both classification accuracy and computational efficiency, a novel remote sensing image classification method is developed in this study. First, the remote sensing image is divided into a training data set and testing data set. Next, the features are extracted from the training data set. In this step, a discretization algorithm based on information entropy is employed to segment the brightness values of each band; each section of the brightness values is regarded as a feature. Then, the feature space is constructed for all the training and testing data. Vector space model (VSM) is applied as the feature representation algorithm. Finally, the testing data set is classified using a *k*-

* Corresponding author.
E-mail address: lixie@tongji.edu.cn (L. Xie).

http://dx.doi.org/10.1016/j.cageo.2015.12.015
0098-3004/

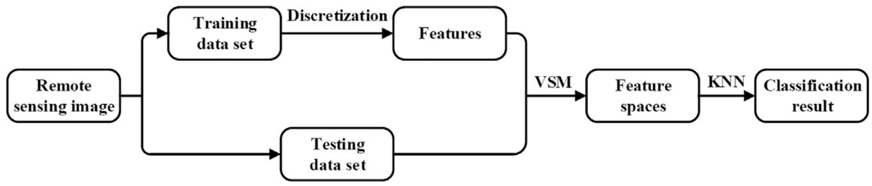


Fig. 1. The flowchart of the proposed method.

nearest neighbors (KNN) model. Fig. 1 presents the flowchart of the proposed method.

This paper is composed of four sections. The remaining sections are organized as follows. Section 2 describes the required algorithms and models for the proposed work. Section 3 introduces the data set and the design of the experiment. The experimental results and discussion are presented in Section 4. Section 5 concludes this paper.

2. Methodology

2.1. Data discretization

The brightness values of each band of remote sensing data are continuous values, typically ranging from zero to 65,535. However, the majority of classification algorithms have difficulty in processing continuous values. Therefore, the continuous attributes must be separated into multiple intervals to accommodate the decision rules. In this paper, a data discretization algorithm is adopted to discretize the continuous brightness values based on information entropy (Xie et al., 2005).

2.1.1. Decision table

To begin, the concept of a decision table is defined. A decision table can be described as a 4-tuple (U, R, V, F). In the tuple, U represents the set of objects. R refers to the set of attributes and $R = C \cup D$, where C is the set of condition attributes and D is the set of decision attributes. V denotes the domain of the attribute values and $V = V_C \cup V_D$, where V_C is the domain of the condition attribute values and V_D is the domain of the decision attribute values. F: $U \times R \to V$. Supposing a decision table contains n objects and m condition attributes, then $U = \{x_1, x_2, \ldots, x_n\}$, $C = \{c_1, c_2, \ldots, c_m\}$, $F(x_i, c_j) = u_{ji}$, and $F(x_i, D) = v_i$. Table 1 provides an example of a decision table.

In terms of remote sensing image classification, the experimental samples (image pixels) correspond to U. The bands of the image and the land cover classes correspond to C and D. V_C refers to the range of the brightness value and V_D is the domain of the land cover class value.

2.1.2. Breakpoint and equivalence class

For a condition attribute c, $c \in C$, its domain is $V_c = [l_c, r_c]$. If there exists values $b_1^c, b_2^c, \ldots, b_m^c$, that split V_c into $(m + 1)$ intervals, namely

Table 1
An example of decision table.

U	C				D
	c_1	c_2	⋯	c_m	
x_1	u_{11}	u_{21}	⋯	u_{m1}	v_1
x_2	u_{12}	u_{22}	⋯	u_{m2}	v_2
⋮	⋮	⋮	⋮	⋮	
x_n	u_{1n}	u_{2n}	⋯	u_{mn}	v_n

$$V_c = [l_c, b_1^c] \cup [b_1^c, b_2^c] \cup \cdots \cup [b_{m-1}^c, b_m^c] \cup [b_m^c, r_c], \tag{1}$$

then, b_i^c is defined as a breakpoint. The sample set is divided into $(m + 1)$ equivalence classes on attribute c by the breakpoint set. The breakpoints can be calculated by the values of attribute c. Supposing the ordered values of attribute c are $l_c = v_0^c < v_1^c < \cdots < v_n^c = r_c$, the candidate breakpoints can be

$$b_i^c = \frac{v_{i-1}^c + v_i^c}{2}, \quad i = 1, 2, \ldots, n. \tag{2}$$

2.1.3. Discretization algorithm

The goal of discretization is to determine an adaptive breakpoint set for each continuous attribute. Supposing $X \subseteq U$ and the number of samples in X is $|X|$; j is the decision attribute of X, $j = 1, 2, \cdots, k$. Then, the information entropy of X can be denoted as

$$H(X) = -\sum_{j=1}^{k} p_j \log_2 p_j, \quad p_j = \frac{k_j}{|X|}. \tag{3}$$

A smaller value of $H(X)$ indicates less disorder in X. $H(X) = 0$ when all samples in X share the same decision values. A breakpoint b_i^c divides X into two subsets, $l^X(b_i^c)$ refers to the quantity of samples whose attribute value on c is smaller than b_i^c and $r^X(b_i^c)$ refers to the quantity of samples whose attribute value on c is greater than b_i^c, namely

$$l^X(b_i^c) = \sum_{j=1}^{k} l_j(b_i^c), \tag{4}$$

$$r^X(b_i^c) = \sum_{j=1}^{k} r_j(b_i^c). \tag{5}$$

The two subsets of X are denoted as X_l and X_r, and their information entropy can be calculated as follows:

$$H(X_l) = -\sum_{j=1}^{k} p_j \log_2 p_j, \quad p_j = \frac{l_j^X(b_i^c)}{l^X(b_i^c)}, \tag{6}$$

$$H(X_r) = -\sum_{j=1}^{k} q_j \log_2 q_j, \quad q_j = \frac{r_j^X(b_i^c)}{r^X(b_i^c)}, \tag{7}$$

additionally, the information entropy of b_i^c to X is

$$H^X(b_i^c) = \frac{|X_l|}{|U|}H(X_l) + \frac{|X_r|}{|U|}H(X_r). \tag{8}$$

Supposing P is the set of selected breakpoints and B is the set of candidate breakpoints, $L = \{X_1, X_2, \ldots, X_m\}$ is considered as the equivalence classes set divided by P. The information entropy after adding a candidate breakpoint b $(b \in B, b \notin P)$ to P can be calculated by

$$H(b, L) = H^{X_1}(b) + H^{X_2}(b) + \cdots + H^{X_m}(b). \tag{9}$$

The steps of the discretization algorithm can be described as

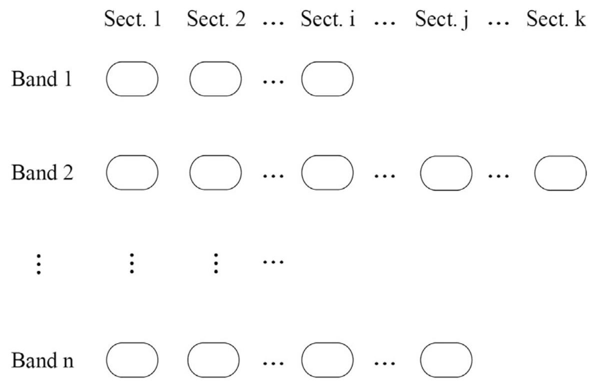


Fig. 2. The outcome of discretization on a sample.

follows:

(i) To begin, let $P = \varnothing$, $L = \{U\}$, $H = H(U)$.
(ii) Compute $H(b, L)$ for each $c \in B$.
(iii) If $H \le \min\{H(b, L)\}$, terminate the loop.
(iv) Add b_{min} to P, creating the minimum value of $H(b, L)$, then $H = H(b, L)$, $B = B - \{b\}$.
(v) For all $X \in L$, if X is divided into X_1 and X_2 by b_{min}, add X_1 and X_2 to L, and remove X from L.
(vi) For each $X_n \in L$, if X_n has the same decision, end the loop, otherwise go to step ii.

After discretizing, each band is separated into several sections and each section is regarded as a feature. Fig. 2 presents the discretized result of a sample.

2.2. Feature representation

Vector space model (VSM) (Salton et al., 1975) is one of the most commonly used methods in feature representation. VSM was initially applied to information retrieval (Salton et al., 1975; Yang and Chute, 1994; Gethers et al., 2011) and is being widely used in textual (Yang, 1999; Tan, 2005; Soucy and Mineau, 2005; Xia and Du, 2011) and visual (Kesorn and Poslad, 2012) content representation.

Take the application of VSM in text classification as an example. The collection of documents is defined as the document space and the words that are used to discriminate the document categories are defined as terms. In VSM, the document space is represented by the documents, each identified by weighted terms. The terms can be denoted as an m-dimensional vector

$$T = (t_1\ t_2\ \cdots\ t_m), \tag{10}$$

where t_j is the jth term. Then, each document d_i can be described by a weighted vector

$$d_i = (w_{t_1}\ w_{t_2}\ \cdots\ w_{t_m}), \tag{11}$$

and the n-dimensional document space is

$$D = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{pmatrix} = \begin{pmatrix} w_{d_1t_1} & w_{d_1t_2} & \cdots & w_{d_1t_m} \\ w_{d_2t_1} & w_{d_2t_2} & \cdots & w_{d_2t_m} \\ \vdots & \vdots & \ddots & \vdots \\ w_{d_nt_1} & w_{d_nt_2} & \cdots & w_{d_nt_m} \end{pmatrix}, \tag{12}$$

where $w_{d_it_j}$ refers to the weight of the jth term on the ith document. The weights may be a binary value according to the existence of the terms (Salton et al., 1975; Yang and Chute, 1994; Kesorn and Poslad, 2012). For example, $w_{d_it_j}=1$ when t_j is present in d_i and $w_{d_it_j}=0$, otherwise.

Because only the band values are studied in this paper, remote sensing image classification can be regarded as a data mining process. The sample set is considered as the document space; a sample is the equivalent of a document, and an attribute node of the sample is the equivalent of a term. The section that contains the brightness value is marked by "1" and other sections are marked by "0". The following array is a case of a sample represented by VSM:

$$S = \begin{pmatrix} 0 & 0 & 0 & 1 & \\ 0 & 0 & 1 & & \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}. \tag{13}$$

It is worth noting that the sample contains three bands that are divided into four, three, and five sections. Moreover, the brightness values of the bands are in section 4, section 3, and section 2, respectively.

2.3. Classification

K-Nearest neighbors (KNN) is a classical pattern recognition algorithm based on statistics. Cover and Hart (1967) demonstrated that the KNN rule is superior to other decision rules because it owns a probability of error that is less than twice the Bayes probability of error. The main function of a KNN model is to identify the k most similar training samples (nearest neighbors) for a testing sample and label the testing sample class according to the category of the k nearest neighbors. The decision function of the testing sample using the KNN model is given as follows (Kim et al., 2005):

$$y(d, c) = \sum_{j=1}^{k} b(d_j, c) sim(d, d_j), \tag{14}$$

where k refers to k nearest neighbors of sample d; $b(d_j, c)$ is a boolean value. If sample d_j belongs to class c, $b(d_j, c)$ is assigned to "1", otherwise $b(d_j, c)$ is assigned to "0"; $sim(d, d_j)$ represents the similarity between sample d and d_j, which can be calculated on the basis of vector angle cosine (Xia and Du, 2011). The vector angle cosine is defined as follows:

$$sim(d_1, d_2) = \frac{\sum_{i=1}^{m} w_{1i} w_{2i}}{\sqrt{\sum_{i=1}^{m} w_{1i}^2 \sum_{i=1}^{m} w_{2i}^2}}, \tag{15}$$

where w_{1i} and w_{2i} refer to the weight of the ith term on sample d_1 and d_2.

Consider n is the total number of categories in a sample set. For each sample d, there are n values of $y(d, c)$, i.e., $y(d, 1), y(d, 2), \cdots, y(d, n)$. Among these values, the maximum is selected and the class of d is labeled c.

3. Experimental design

A Landsat8 satellite image with a ground resolution of 30 m was used for the experiments. The study area was located in Huairou district (116°32′ – 116°43′ E, 40°16′ – 40°24′ N) in Peking, China (Fig. 3). The study area consisted of 500×500 samples (pixels), which were composed primarily of five land cover types including cropland, fallow, forest, residential area, and water. The samples were labeled with class information based on an official land use map and field observation. All of the samples were used to generate a set of 50,000 training samples and a set of 200,000 testing samples using a random sampling method. Furthermore, to simplify the experimental process and accelerate the speed of computing, only the first eight bands were selected in the experiments.

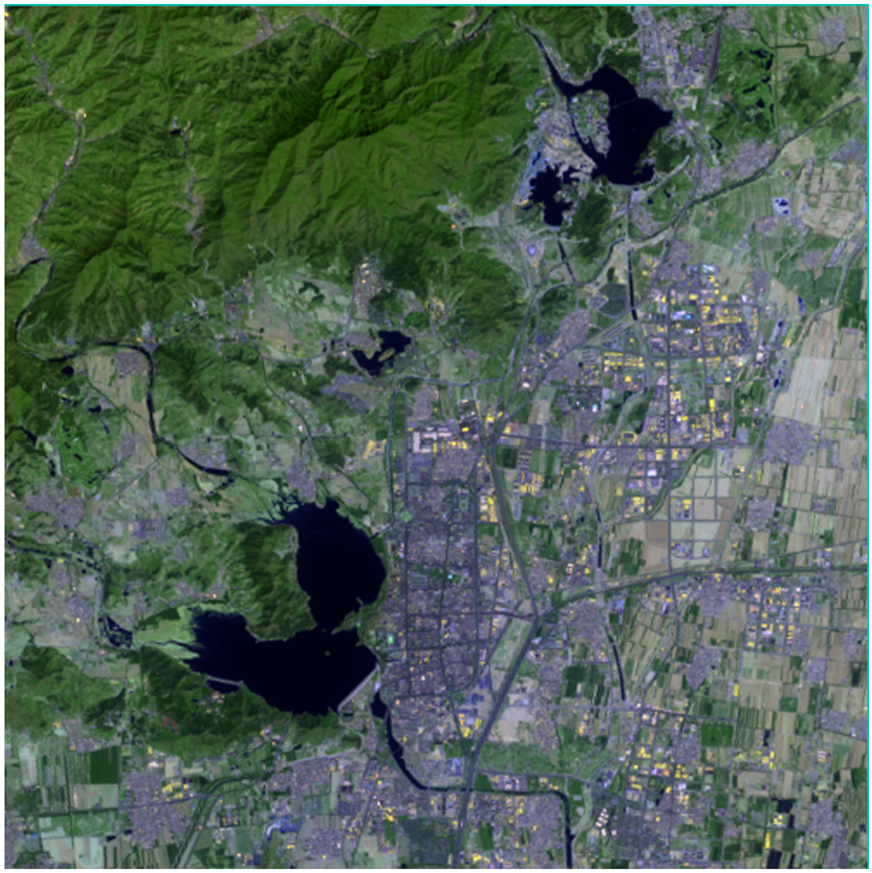


Fig. 3. The Landsat8 image (7,6,4) of the study area in Peking.

The experimental results of the proposed method were compared with the back propagation neural network (BPNN) method and the ant colony optimization (ACO) method. Compared with classical neural network models, the BPNN method reduces the training time and improves the computational efficiency (Heermann and Khazenie, 1992). Furthermore, the BPNN method is flexible and can be modified to function with a high-dimensional feature space. The ACO method is a swarm intelligence algorithm that searches for the shortest path or optimal solution to a specific problem. When it comes to classification, the ACO method searches for a solution in the classification rules. Liu et al. (2008) proved that the ACO method is more efficient and accurate than the decision tree method in remote sensing image classification.

All three methods were implemented in Matlab R2012b and executed on a computer with an Intel(R) Core(TM) i7-3612QM with a 2.10 GHz CPU and 8.00 GB RAM.

4. Results and discussion

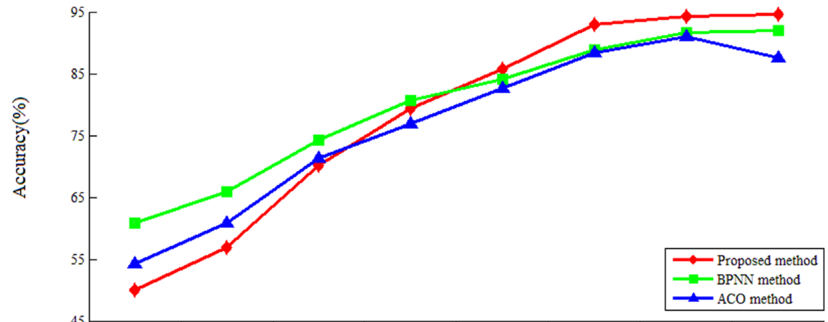
4.1. Evaluation of classification accuracy

In this section, the classification accuracy of the proposed method, BPNN method, and ACO method was evaluated by two groups of experiments. The objectives of the experiments were to study the effects of feature dimensionality and training set size on the classification accuracy of the three algorithms.

The purpose of the first experiment in this section was to evaluate the effects of feature dimensionality on classification accuracy. As mentioned in Section 2.1.3, each section of a band was regarded as a feature and hence, a band was considered as a dimension of features. In the first experiment, eight subsets were derived from the training set. The first subset included only band 1, the second subset used bands 1 and 2, the third subset used bands 1–3, continuing similar to the eighth subset. The number of pixels in each subset was 50,000. The eight subsets were employed to train three classifiers separately; the same testing set was used for validation.

Fig. 4 illustrates the relationship between the classification accuracy and the number of bands. The results indicate that the classification accuracy of the three algorithms demonstrated a remarkable increase for the first seven subsets. The level of accuracy for the proposed and BPNN methods continued to increase slightly until the feature dimensionality reached eight. As the feature dimensionality increased from one to eight, the classification accuracy increased from 49.9% to 94.7% for the proposed method; it increased from 60.9% to 91.9% for the BPNN method. Consequently, it can be concluded that the classification accuracy of the proposed and BPNN methods increased with the feature dimensionality growth.

An interesting phenomenon occurred with the ACO method. The accuracy reached the peak (90.9%) when the number of bands was seven. Then, the accuracy declined by 4.3% when the eighth subset was used. A possible reason may be that the ACO method searches for the classification rules according to the probability (Dorigo et al., 1996; Parpinelli et al., 2002; Liu et al., 2008; Jovanovic and Tuba, 2013). A classification rule is composed of several conditions and the number of conditions is equal to the feature dimensionality. The probability of a condition being selected to construct a rule depends on the heuristic and pheromone value. Thus, it is difficult to acquire a rule when the feature dimensionality is high. A solution to this problem would be to use a rule-pruning technique to simplify the rules and improve the classification accuracy (Parpinelli et al., 2002; Liu et al., 2008). However, a



	1	2	3	4	5	6	7	8
Proposed method	49.9	56.9	70.1	79.4	85.8	92.9	94.3	94.7
BPNN method	60.9	65.8	74.3	80.7	84.2	88.9	91.7	91.9
ACO method	54.2	60.8	71.3	76.9	82.6	88.4	90.9	87.5

Number of bands

Fig. 4. Effects of feature dimensionality on classification accuracy.

rule-pruning technique for the ACO method is not discussed in this paper because it is not the emphasis of this research.

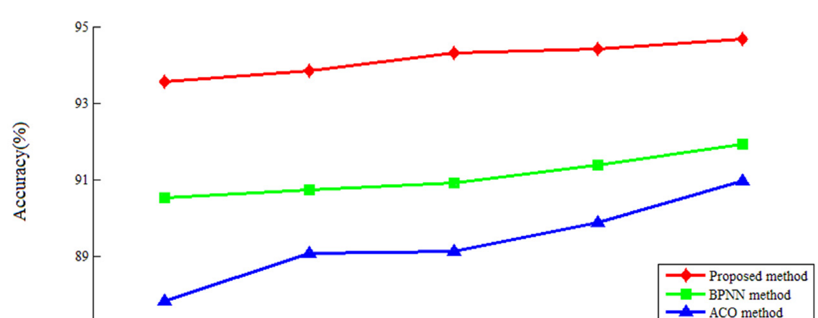
The purpose of the second experiment in this section was to assess the effects of the size of the training set on classification accuracy. Five subsets were extracted from the training set using a random sampling method. The number of samples in each subsets were 2500, 5000, 10,000, 20,000, and 50,000 pixels. The five subsets were used to train three classifiers separately and an identical testing set was used to evaluate the classification accuracy. It should be noted that the optimal parameters were selected in the second experiment to achieve the best results, i.e., the feature dimensionality was set to seven for the ACO method, and eight for the proposed and BPNN methods.

The results presented in Fig. 5 prove that the level of classification accuracy is influenced by the size of training set, regardless of what classifier is chosen. With regard to the proposed and BPNN methods, the accuracy improved as the size of the training set increased. For the ACO method, the accuracy increased from the first subset to the second. Then, the accuracy remained approximately equivalent with the second and third subsets, then, increased again such that the highest accuracy was obtained by the fifth subset. Therefore, the overall trend of the accuracy for the ACO method was also upward. On the other hand, the growth rates of the accuracy for the three algorithms were slight; only a 1.1% (from 93.6% to 94.7%) increment of accuracy for the proposed method when the number of training samples increased from 2500 to 50,000, and 1.4% (from 90.5% to 91.9%) and 3.1% (from 87.8% to 90.9%) increments were demonstrated by the BPNN and ACO methods, respectively. The following three tables present the confusion matrices of the three algorithms with optimal parameters.

The comparison of classification results for the three algorithms is presented in Tables 2–4. The proposed method had an overall accuracy of 94.7% and a Kappa coefficient of 0.927. The total accuracy of the BPNN method was 91.9% with a Kappa coefficient of 0.889. The total accuracy and the Kappa coefficient obtained by the ACO method were 90.9% and 0.875, respectively. The comparative results suggest that the proposed method outperformed the other two methods in classification accuracy. Fig. 6 presents the classification results of the study area using the three algorithms.

4.2. Evaluation of computational efficiency

In this section, the computational efficiency of the three algorithms was evaluated by two groups of experiments. The first experiment was to assess the effects of training set size on computational efficiency; the second was designed to study the relationship between computational efficiency and testing set size. In the first experiment, five training sets with different sizes and the same testing set were employed. The training sets were composed of 2500, 5000, 10,000, 20,000, and 50,000 pixels. The testing set was composed of 200,000 pixels. The same training set and five testing sets with different sizes were employed in the second experiment. The training set composed of 2500 pixels; the number of pixels in each of the testing sets were 10,000, 20,000, 40,000, 80,000, and 200,000 pixels. The overall time consumption



	2500	5000	10000	20000	50000
Proposed method	93.6	93.8	94.3	94.4	94.7
BPNN method	90.5	90.7	90.9	91.4	91.9
ACO method	87.8	89.0	89.0	89.9	90.9

Size of training set

Fig. 5. Effects of training set size on classification accuracy.

Table 2
Classification accuracy assessment on the proposed method.

Class	Cropland	Fallow	Forest	Residential area	Water	User's accuracy (%)
Cropland	18,609	0	1561	1256	81	86.5
Fallow	0	35,961	0	1094	0	97.0
Forest	2686	0	61395	389	38	95.2
Residential area	1408	1646	324	65,032	0	95.1
Water	9	0	176	0	8335	97.8
Producer's accuracy (%)	82.0	95.6	96.8	96.0	98.6	
	Total accuracy = 94.7%	Kappa coefficient = 0.927				

Table 3
Classification accuracy assessment on the BPNN method.

Class	Cropland	Fallow	Forest	Residential area	Water	User's accuracy (%)
Cropland	15,835	41	2737	2803	91	73.6
Fallow	158	32,839	158	3729	171	88.6
Forest	2426	5	61,283	774	20	95.0
Residential area	874	1142	382	65,971	41	96.4
Water	0	0	590	0	7930	93.1
Producer's accuracy (%)	82.1	96.5	94.1	90.0	96.1	
	Total accuracy = 91.9%	Kappa coefficient = 0.889				

Table 4
Classification accuracy assessment on the ACO method.

Class	Cropland	Fallow	Forest	Residential area	Water	User's accuracy (%)
Cropland	14,253	0	4540	2587	127	66.3
Fallow	0	35,046	1	2008	0	94.6
Forest	1550	0	62,092	689	177	96.3
Residential area	2641	3027	536	62,206	0	90.9
Water	0	0	300	3	8217	96.4
Producer's accuracy (%)	77.3	92.0	92.0	92.2	96.4	
	Total accuracy = 90.9%	Kappa coefficient = 0.875				

of the experiment was chosen as the criterion to measure computational efficiency. A lower time consumption represented a higher computational efficiency.

The overall experiment time consumption of the three methods with different sizes of training set and the 200,000 pixel testing set are presented in Fig. 7. The results suggest that the training set size has a significant influence on the computational efficiency. The time consumptions of the three algorithms increased with an increased training set size. The BPNN method demonstrated the highest time consumption of the three algorithms. It required approximately ten hours to complete the classification process when 50,000 pixels were used as the training set. Moreover, the computational efficiency of the BPNN method would dropped sharply with the increase of the training set size. The ACO method was significantly faster than the BPNN method when the same training set was used. It required less than 1 h to classify the testing set with the largest training set. The proposed method performed the best of the three algorithms for computational efficiency; it required the minimum time for the experiment.

Fig. 8 presents the overall time consumption of the three algorithms with the 2500 pixel training set and different sizes of testing set. The results indicate that the training set size had a

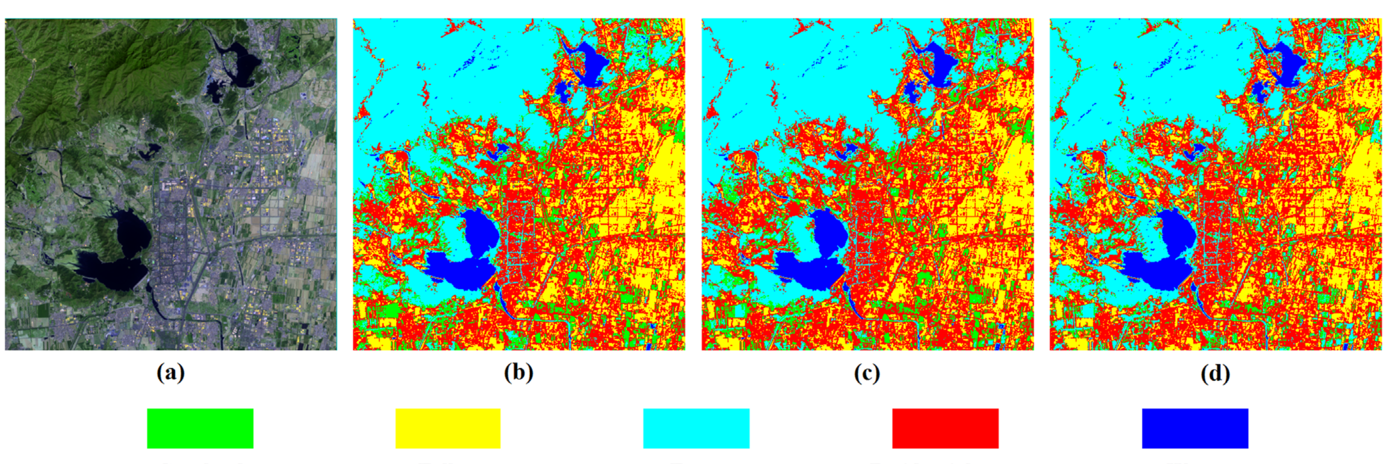
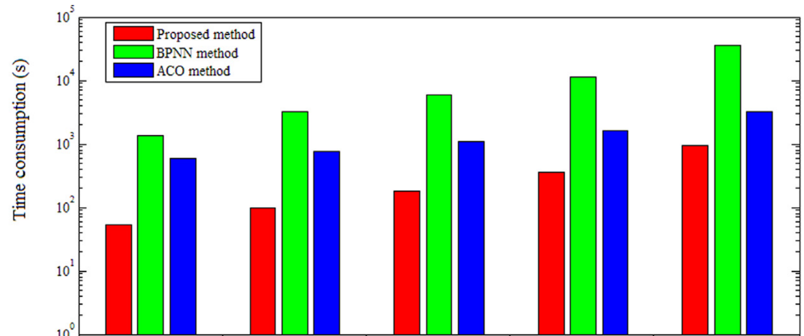


Fig. 6. (a) The original remote sensing image. (b) The classification result obtained by the proposed method. (c) The classification result obtained by the BPNN method. (d) The classification result obtained by the ACO method.



	2500	5000	10000	20000	50000
Proposed method	54.2	98.4	183.9	353.0	963.1
BPNN method	1347.4	3212.1	5994.9	11432.7	36403.6
ACO method	589.9	759.1	1102.8	1610.4	3196.5

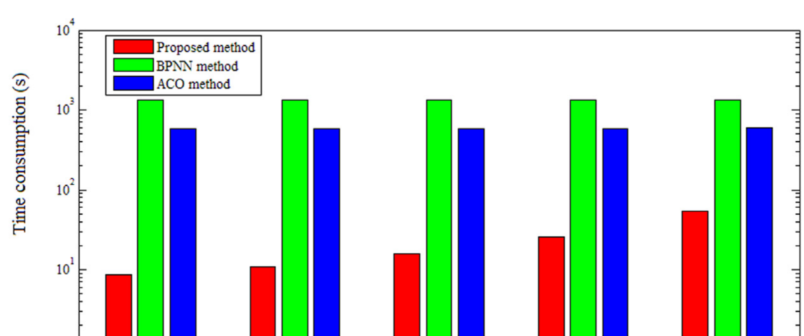
Size of training set

Fig. 7. Effects of training set size on computational efficiency.

different impact on the computational efficiency of the three algorithms. The time consumption of the proposed method was approximately linear, upwards from the first testing set to the fifth. The time consumptions of the BPNN and ACO methods also grew linearly with the size of the testing set, however, the growth rates were marginal. Therefore, it can be concluded that the testing set size had a strong influence on the computational efficiency of the proposed method; it had a minimal influence on the computational efficiency of the BPNN and ACO methods.

The results in Figs. 7 and 8 indicate that the size of the data set had a different impact on the computational efficiency of the three algorithms. Because the KNN model is employed as the classification model of the proposed method, the similarity between every testing sample and every training sample must be calculated during the experiment. Therefore, both training set size and testing set size had a substantial influence on the computational efficiency of the proposed method. For the BPNN method, increasing iterations were required to train the BPNN model with the growth of the training set size. Therefore, the time consumption of the training phase continued to increase. In the testing phase, the testing data were input into the model and the class labels were output by the model. Compared with the training phase, the testing phase required a minimal amount of time to classify the testing data. Hence, the computational efficiency was primarily influenced by the training set size and only marginally influenced by the testing size for the BPNN method. Further, the computational efficiency of the ACO method was also mainly affected by the training set size and only slightly influenced by the testing size. The training strategy of the ACO method is to identify classification rules and remove the samples that are covered by the rules from the training set. Then, the testing data are labeled with class information according to the classification rules in the testing phase. Similar to the BPNN method, the testing phase of the ACO method required significantly less time than the training phase.

The comparison of the three methods on computational efficiency with the same size of data set indicates that the proposed method outperformed the BPNN and ACO methods considerably. The main reason is that both the BPNN and ACO methods require excessive time constructing the feature space in the training phase because they require iterative training procedures to represent the features. Conversely, the proposed method employs VSM to construct the feature space and hence, does not require iterations to represent the features. The time consumption in the training phase is significantly reduced. Even though the proposed method requires more time than the other two algorithms in the testing phase, the loss of computational efficiency is compensated by the fast training procedure.



	10000	20000	40000	80000	200000
Proposed method	8.6	10.9	15.6	26.0	54.2
BPNN method	1343.1	1343.3	1343.8	1344.7	1347.4
ACO method	585.0	585.3	585.8	586.8	589.9

Size of testing set

Fig. 8. Effects of testing set size on computational efficiency.

5. Conclusion

A new method for remote sensing image classification was successfully implemented in this research work. To extract features from the remote sensing data, a discretization algorithm based on information entropy was employed. Then, VSM was used to represent the features and construct the feature space. Finally, because it is one of the most effective classification algorithms for VSM, a KNN model was employed to classify the testing data set. The main novelty of the proposed method is applying a classical text classification algorithm to remote sensing image classification. The combination of VSM and KNN is widely used in text classification and high classification accuracy can be achieved using this method (Yang, 1999; Soucy and Mineau, 2005; Xia and Du, 2011). However, this method is seldom applied to remote sensing image classification because it is difficult for VSM to process continuous values. Using the proposed method, the band values are effectively transformed into discrete values such that VSM can be applied to represent the features of the remote sensing data.

The experiments in this study aimed to evaluate the applicability of the proposed method in remote sensing image classification and compared the performance of the proposed method with two other algorithms, the BPNN and ACO methods. The assessment of applicability included not only classification accuracy but also computational efficiency. The experimental results identified specific conclusions. First, the classification accuracy of the proposed method was influenced by the feature dimensionality and training set size. A higher accuracy could be achieved by a higher feature dimensionality or a larger training set size. Secondly, the computational efficiency of the proposed method was influenced by the size of the training and testing sets. The computational efficiency of the proposed method decreased as the data set size increased. The results also indicated that the proposed method improved both classification accuracy and computational efficiency compared with the BPNN and ACO methods.

Acknowledgment

The research is supported by National Nature Science Foundation of China (NSFC) (Nos. 60771065 and 51378365).

Appendix A. Supplementary data

Supplementary data associated with this paper can be found in the online version at http://dx.doi.org/10.1016/j.cageo.2015.12.015.

References

Benediktsson, J., Swain, P.H., Ersoy, O.K., 1990. Neural network approaches versus statistical methods in classification of multisource remote sensing data. IEEE Trans. Geosci. Remote Sens. 28 (4), 540–552.
Bischof, H., Schneider, W., Pinz, A.J., 1992. Multispectral classification of landsat-images using neural networks. IEEE Trans. Geosci. Remote Sens. 30 (3), 482–490.
Bruzzone, L., Prieto, D.F., Serpico, S.B., 1999. A neural-statistical approach to multitemporal and multisource remote-sensing image classification. IEEE Trans. Geosci. Remote Sens. 37 (3), 1350–1359.
Cover, T., Hart, P., 1967. Nearest neighbor pattern classification. IEEE Trans. Inf. Theory 13 (1), 21–27.
Dai, Q., Liu, J., 2007. Application of ant colony optimization (aco) algorithm to remote sensing image classification. In: International Symposium on Multispectral Image Processing and Pattern Recognition, pp. 67881A1–67881A6.
Dorigo, M., Maniezzo, V., Colorni, A., 1996. Ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybern. Part B: Cybernet. 26 (1), 29–41.
Friedl, M.A., Brodley, C.E., 1997. Decision tree classification of land cover from remotely sensed data. Remote Sens. Environ. 61 (3), 399–409.
Gethers, M., Oliveto, R., Poshyvanyk, D., Lucia, A.D., 2011. On integrating orthogonal information retrieval methods to improve traceability recovery. In: 2011 27th IEEE International Conference on Software Maintenance (ICSM), pp. 133–142.
Han, M., Zhu, X., Yao, W., 2012. Remote sensing image classification based on neural network ensemble algorithm. Neurocomputing 78 (1), 133–138.
Heermann, P.D., Khazenie, N., 1992. Classification of multispectral remote sensing data using a back-propagation neural network. IEEE Trans. Geosci. Remote Sens. 30 (1), 81–88.
Jovanovic, R., Tuba, M., 2013. Ant colony optimization algorithm with pheromone correction strategy for the minimum connected dominating set problem. Comput. Sci. Inf. Syst. 10 (1), 133–149.
Kavzoglu, T., 2009. Increasing the accuracy of neural network classification using refined training data. Environ. Model. Softw. 24 (7), 850–858.
Kerroum, M.A., Hammouch, A., Aboutajdine, D., 2010. Textural feature selection by joint mutual information based on Gaussian mixture model for multispectral image classification. Pattern Recognit. Lett. 31 (10), 1168–1174.
Kesorn, K., Poslad, S., 2012. An enhanced bag-of-visual word vector space model to represent visual content in athletics images. IEEE Trans. Multimed. 14 (1), 211–222.
Kim, H., Howland, P., Park, H., 2005. Dimension reduction in text classification with support vector machines. J. Mach. Learn. Res. 6, 37–53.
Liu, X., Li, X., Liu, L., He, J., Ai, B., 2008. An innovative method to classify remote-sensing images using ant colony optimization. IEEE Trans. Geosci. Remote Sens. 46 (12), 4198–4208.
McIver, D., Friedl, M., 2002. Using prior probabilities in decision-tree classification of remotely sensed data. Remote Sens. Environ. 81 (2), 253–261.
Melgani, F., Bruzzone, L., 2004. Classification of hyperspectral remote sensing images with support vector machines. IEEE Trans. Geosci. Remote Sens. 42 (8), 1778–1790.
Miller, D.M., Kaminsky, E.J., Rana, S., 1995. Neural network classification of remote-sensing data. Comput. Geosci. 21 (3), 377–386.
Pal, M., Mather, P.M., 2003. An assessment of the effectiveness of decision tree methods for land cover classification. Remote Sens. Environ. 86 (4), 554–565.
Pal, M., Maxwell, A.E., Warner, T.A., 2013. Kernel-based extreme learning machine for remote-sensing image classification. Remote Sens. Lett. 4 (9), 853–862.
Parpinelli, R.S., Lopes, H.S., Freitas, A., 2002. Data mining with an ant colony optimization algorithm. IEEE Trans. Evol. Comput. 6 (4), 321–332.
Salton, G., Wong, A., Yang, C.-S., 1975. A vector space model for automatic indexing. Commun. ACM 18 (11), 613–620.
Shao, Y., Lunetta, R.S., 2012. Comparison of support vector machine, neural network, and cart algorithms for the land-cover classification using limited training data points. ISPRS J. Photogram. Remote Sens. 70, 78–87.
Soucy, P., Mineau, G.W., 2005. Beyond tfidf weighting for text categorization in the vector space model. In: International Joint Conference on Artificial Intelligence, vol. 5, pp. 1130–1135.
Strahler, A.H., 1980. The use of prior probabilities in maximum likelihood classification of remotely sensed data. Remote Sens. Environ. 10 (2), 135–163.
Tan, S., 2005. Neighbor-weighted k-nearest neighbor for unbalanced text corpus. Expert Syst. Appl. 28 (4), 667–671.
Tso, B.C., Mather, P.M., 1999. Classification of multisource remote sensing imagery using a genetic algorithm and Markov random fields. IEEE Trans. Geosci. Remote Sens. 37 (3), 1255–1260.
Wilkinson, G.G., 2005. Results and implications of a study of fifteen years of satellite image classification experiments. IEEE Trans. Geosci. Remote Sens. 43 (3), 433–440.
Xia, T., Du, Y., 2011. Improve vsm text classification by title vector based document representation method. In: 2011 6th International Conference on Computer Science and Education (ICCSE), pp. 210–213.
Xie, H., Cheng, H.Z., Niu, D.X., 2005. Discretization of continuous attributes in rough set theory based on information entropy. Chinese J. Comput. 28 (9), 1570–1574.
Xu, M., Watanachaturaporn, P., Varshney, P.K., Arora, M.K., 2005. Decision tree regression for soft classification of remote sensing data. Remote Sens. Environ. 97 (3), 322–336.
Yang, Y., Chute, C.G., 1994. An example-based mapping method for text categorization and retrieval. ACM Trans. Inf. Syst. 12 (3), 252–277.
Yang, H., Du, Q., Chen, G., 2012. Particle swarm optimization-based hyperspectral dimensionality reduction for urban land cover classification. IEEE J. Select. Top. Appl. Earth Observ. Remote Sens. 5 (2), 544–554.
Yang, Y., 1999. An evaluation of statistical approaches to text categorization. Inf. Retr. 1, 69–90.
Zhong, Y., Zhang, L., 2012. An adaptive artificial immune network for supervised classification of multi-hyperspectral remote sensing imagery. IEEE Trans. Geosci. Remote Sens. 50 (3), 894–909.