ChinaRxiv

AI translation · View original & related papers at
chinarxiv.org/items/chinaxiv-201812.00067

Postprint of a Real-Time Classification Algorithm for Remote Sensing Images Based on Ant Colony Optimization and Independent Feature Sets

Authors: Zhao Fang, Suo Yan, Peng Ziran

Date: 2018-12-13T00:00:00+00:00

Abstract

To improve the real-time classification accuracy and efficiency of remote sensing images, a real-time classification algorithm for remote sensing image sets based on ant colony optimization algorithm and independent feature sets is proposed. First, wavelet domain features and color features of remote sensing images are extracted and composed into feature vectors. Then, the ant colony optimization algorithm is employed to optimize the feature space, independently selecting significant feature sets for each class, thereby reducing the dimensionality of each sub-feature space. Finally, an extreme learning machine classifier is independently trained for each class, thereby achieving classification of remote sensing image sets. Simulation experiments were conducted based on publicly available remote sensing image datasets, and the results demonstrate that the proposed algorithm achieves high classification accuracy and high computational efficiency.

Full Text

Preamble

Real-time Classification Algorithm for Remote Sensing Images Based on Ant Colony Optimization and Independent Feature Sets

Zhao Fang[1], Suo Yan[2], Peng Ziran[3]
(1. Institute of Computer & Information Engineering, Xinxiang University, Xinxiang Henan 453003, China;
2. Xinlian College, Henan Normal University, Xinxiang Henan 453003, China;
3. School of Information Science & Engineering, Central South University, Changsha 410083, China)

Abstract: To improve the accuracy and efficiency of real-time classification for remote sensing images, this paper proposes a real-time classification algorithm that integrates ant colony optimization with independent feature sets. The algorithm first extracts wavelet-domain and color features from remote sensing images to form feature vectors. It then employs ant colony optimization to optimize the feature space by independently selecting salient feature subsets for each class, thereby reducing the dimensionality of each subspace. Finally, an independent extreme learning machine classifier is trained for each class to achieve classification of the remote sensing image set. Simulation experiments conducted on a public remote sensing image dataset demonstrate that the proposed algorithm achieves high classification accuracy while maintaining superior computational efficiency.

Keywords: artificial intelligence; feature extraction; remote sensing image; computational efficiency; ant colony optimization algorithm; extreme learning machine

0 Introduction

Remote sensing technology represents a critical research area in military reconnaissance, missile early warning, military mapping, ocean surveillance, and meteorological observation. Ground detection and reconnaissance through remote sensing images hold substantial research value [1]. In mineral exploration and petroleum engineering applications, preliminary surface surveys and geological structure analysis using satellite remote sensing images have become essential technical approaches [2]. Furthermore, in robotic exploration of desert regions, acquiring remote sensing images constitutes a crucial task for exploration robots [3]. However, communication capabilities between exploration robots and satellites are limited. Transmitting remote sensing image sets in real-time to workstations would impose enormous communication burdens [4]. Therefore, performing real-time preprocessing of remote sensing image sets onboard exploration robots or satellites not only alleviates communication loads but also enhances ground positioning effectiveness [5,6].

Remote sensing images contain rich detail information and exhibit high feature dimensionality. Conventional feature extraction schemes typically yield weak feature representation capability and severe information loss, adversely affecting subsequent image processing [7]. Reference [8] proposed a dimensionality reduction approach that extracts spectral features from pixels and uses convolutional neural networks to extract spatial features from image patches, concatenating these two feature types before classification with a support vector machine. While this method effectively improved classification accuracy, its time efficiency remained low. Reference [9] introduced a high-resolution remote sensing image road extraction method based on salient features and GVF Snake, which calculates saliency maps by fusing color contrast and spatial statistical features, using the maximum saliency value as the initial seed point for GVF Snake. Although this approach simultaneously enhanced computational efficiency and detection precision, its reliance on the GVF Snake seed growth algorithm limited its applicability to targets with prominent contours.

Wavelet features can extract detailed information from remote sensing images [10]. This paper considers both wavelet-domain and color features of remote sensing images. To ensure low processing time for remote sensing image classification, effective feature reduction constitutes another important research problem. Currently, successful feature optimization schemes primarily include ant colony optimization [11], particle swarm optimization [12], and genetic algorithms [13], among which ant colony optimization has demonstrated superior performance. Consequently, this paper adopts ant colony optimization to optimize feature sets for each class. Classifier time efficiency also significantly influences remote sensing image classification performance. While support vector machines are commonly used in remote sensing image classification [14], our experiments reveal that extreme learning machines offer slightly better computational efficiency than SVMs. Therefore, this paper employs extreme learning machines as classifiers.

In summary, to enable real-time classification processing of remote sensing image sets at measurement terminals, it is essential to effectively reduce algorithmic computation time while maintaining classification accuracy. This paper proposes a real-time classification algorithm for remote sensing image sets based on ant colony optimization and independent feature sets. The algorithm extracts wavelet and color features from remote sensing images, optimizes feature quantities using ant colony optimization, and trains an independent extreme learning machine classifier for each image class. Simulation experiments based on a public remote sensing image dataset demonstrate that the proposed algorithm not only ensures high classification accuracy but also effectively improves computational efficiency, facilitating real-time image preprocessing at remote sensing measurement terminals.

---

1 Feature Extraction

Feature extraction transforms complex image datasets into feature vectors that represent large-scale data, thereby reducing storage and computational costs. In our feature extraction experiments, feature vectors are extracted for each pixel from its surrounding window. Rock features in ground surface remote sensing images are represented using texture descriptors from wavelet transforms, with statistical parameters in the wavelet domain serving as pixel descriptors. Other surface features are represented using color feature descriptors. Our feature descriptor comprises four types of features: wavelet-domain coefficient features, dominant color descriptor (DCD) features, local color histogram features, and color statistical features.

1.1 Wavelet Features

First, a window is established for each pixel in the input image, and a one-level two-dimensional wavelet decomposition is performed on this window. The Euclidean norm is then applied to the rows and columns of wavelet coefficients in the LH and HL components [15]. Finally, the mean and variance of the Euclidean norms are calculated to obtain eight wavelet feature values.

1.2 Color Features

Color features are low-level features that are insensitive to transformations such as rotation and scaling. Since remote sensing images contain significant noise and many images exhibit rotation, scaling, and other deformations, color features effectively address these challenges. This paper employs three color features: dominant color descriptor (DCD) features, local color histograms, and color statistical features.

1.2.1 Dominant Color Descriptor DCD is a color descriptor recognized by the MPEG-7 standard. DCD partitions image colors into several regions called “coarse partitions,” assuming all points within the same partition are similar. The partition center is calculated as the mean value of all pixel colors within that partition, as expressed in the following equation:

$$\text{Center} = \frac{1}{|P_i|} \sum_{p \in P_i} C(p)$$

where P_i represents the i-th partition. This paper extracts DCD features in the RGB color domain, assuming each pixel belongs to a partition and replacing the pixel’ s color value with its partition center value, thereby quantizing the image’ s colors. For each pixel, the DCD value of its window is calculated, and these DCD values form a feature vector. [Figure 1: see original paper] illustrates an original remote sensing image and its corresponding DCD image in RGB color space.

1.2.2 Local Color Histogram Features A histogram represents the discrete statistical probability density of an image. Remote sensing images are typically in RGB format with each color channel ranging from [0,255]. Considering all 256 bins for each histogram would result in high computational complexity. This paper quantizes each color channel into 8 equal bins and calculates the local histogram for each pixel’ s window. Consequently, each image pixel is mapped to a 24-dimensional vector.

1.2.3 Color Statistical Features The mean and standard deviation of each RGB channel are considered as color statistical features for each pixel.

2 Classifiers

2.1 Support Vector Machine (SVM) Classifier

Remote sensing images contain numerous pixel classes. Extracted features are input to the classifier to train model parameters. For each pixel class, a one-versus-all binary SVM is trained. Assuming pixels from the primary class as positive samples and all other class pixels as negative samples, the primary class typically has far fewer pixels than other classes, leading to an imbalanced SVM model biased toward negative samples. To address this issue, k samples are randomly selected from the negative sample set to serve as the negative training set for the SVM, thereby achieving balanced positive and negative samples. [Figure 2: see original paper] illustrates the balanced SVM training method.

2.2 Extreme Learning Machine (ELM) Classifier

ELM was originally proposed for single-hidden-layer feedforward neural networks (SLFN). For generalized SLFNs, ELM can be expressed as:

$$f_L(x) = \sum_{i=1}^{L} \beta_i h_i(x)$$

where β_i represents the weight between the i-th hidden node and output node, and $h_i(x)$ denotes the output of the i-th hidden node. The hidden node output function $h_i(x)$ is typically a nonlinear piecewise continuous function such as Sigmoid, Hardlimit, Gaussian, or multiquadric functions.

The objective of feedforward neural networks is to achieve minimal training error. ELM simultaneously considers minimal training error and minimal output weight norm, as smaller weight norms yield better generalization performance. ELM can be formulated as the following optimization problem:

$$\min_{\beta} \sigma_1 \|\beta\|_p + \sigma_2 \|H\beta - T\|_q$$

where σ_1 and σ_2 are positive constants, $p, q = 0, 0.5, 1, 2, ...$, H represents the hidden layer output matrix for h equations, and T is the target matrix of training data. This paper employs ELM to train pixel feature vector sets and subsequently classify feature vectors of other pixels. The ELM classifier is implemented using MATLAB' s ELM toolbox.

---

3 Feature Selection

Feature selection is a discrete optimization problem that selects m features from n available features. The total number of possible subsets is calculated as:

$$\sum_{s=0}^{n}\binom{n}{s}=2^{n}$$

where n denotes the dimensionality of the feature vector and s represents the size of the current feature subset.

3.1 Feature Selection Using Ant Colony Optimization

Ant colony optimization (ACO) is an iterative, probability-based heuristic algorithm that simulates the natural foraging behavior of ant colonies. Ants traverse nodes in a graph, depositing pheromones on each traversed path and selecting paths with higher pheromone concentrations. The shortest path in the graph accumulates more pheromones, increasing its selection probability. Pheromones also evaporate over time. The feature selection problem is modeled as a shortest path selection problem in a graph, where nodes represent features and edges represent the next selected feature [16]. After traversal, the ant colony selects the optimal feature subset. In this approach, pheromones and heuristic values are associated with features (nodes). The probability of ant k selecting feature i at time t is calculated as:

$$P_i^k(t)=\begin{cases}\frac{[\tau_i(t)]^\gamma[\eta_i]^\delta}{\sum_{u\in J_k}[\tau_u(t)]^\gamma[\eta_u]^\delta}, & \text{if } i\in J_k\\ 0, & \text{otherwise}\end{cases}$$

where J_k represents the set of currently unvisited features, $\tau_i(t)$ and η_i denote the pheromone value and heuristic value associated with feature i, and γ and δ are weights for pheromone and heuristic information, respectively. The pheromone deposited by ant k on feature i at time t is calculated as:

$$\Delta\tau_i^k(t)=\begin{cases}\frac{\phi\cdot H(S_k(t))+\psi\cdot(FC-|S_k(t)|)}{FC}, & \text{if } i\in S_k(t)\\ 0, & \text{otherwise}\end{cases}$$

where FC represents the total number of features, $S_k(t)$ and $|S_k(t)|$ denote the feature subset discovered by ant k at iteration t and its length, respectively. $H(S_k(t))$ represents the classifier performance evaluation result for subset $S_k(t)$, and ϕ and ψ are weights balancing classifier performance and feature subset length. The pheromone update method for nodes is:

$$\tau_i(t+1)=(1-\rho)\tau_i(t)+\sum_{k=1}^{m}\Delta\tau_i^k(t)+\Delta\tau_i^g(t)$$

where m is the number of ants, ρ is the pheromone evaporation rate, and g represents the current best ant. This formulation ensures that the best ant has a greater influence on pheromone updates than other ants, concentrating the colony on optimal solutions in subsequent iterations. [Figure 3: see original paper] illustrates the ACO-based feature selection scheme.

3.2 Our Feature Selection Scheme

Selecting the most relevant features not only reduces feature vector dimensionality and classifier computational cost but also improves system classification accuracy. In our pixel classification system, feature selection is performed after the feature extraction stage.

3.2.1 Single Feature Set for All Classes (First Feature Selection Scheme) The ACO-based feature selection scheme is a wrapper-based approach. In traditional ACO feature selection algorithms, each ant selects optimal features based on path efficiency. For a feature vector with n features, the second feature requires $n-1$ model trainings, the third feature requires $n-2$ trainings, and in the worst case, each ant must complete $n(n-1)/2$ model trainings per iteration, resulting in extremely high computational cost. This paper proposes a novel scheme that maintains high classification performance while reducing computational cost. In the original ACO algorithm, each iteration requires evaluating all unvisited features, which is computationally expensive. Experimental results from various literature demonstrate that this step is unnecessary for all ants.

To address this issue, we partition features into 13 groups as shown in . The table includes: LH_meanL and LH_meanH (wavelet LH band L and H component norm means), HL_meanH and HL_meanL (wavelet HL band H and L component norm means), DCDRiR, DCDRiG, and DCDRiB (dominant colors for R, G, B channels), Rm&Rs, Gm&Gs, and Bm&Bs (means and standard deviations for R, G, B channels), and Ri, Gi, Bi (histogram bins for R, G, B channels).

The advantage of feature grouping is that each ant only needs to evaluate one feature per group. Since features within each group have similar effects on classification accuracy, training all features in each group is unnecessary. For example, if LH_meanL is selected from group 1 in iteration k, then LH_stdL becomes a candidate solution in iteration $k+1$, ensuring all features in the group have selection opportunities.

In the first feature selection scheme, feature selection is performed simultaneously for all pixel classes by maximizing classification accuracy across all pixels to extract the optimal feature subset. Consequently, all classes share the same feature subset.

The steps of the first feature selection method are: a) Set ant colony size, map each feature to a graph node, assign random pheromone levels to each node, and set maximum iterations and termination conditions. b) Partition the feature set into groups. c) Each ant creates a solution, starting traversal from an initial solution (initial node) and marking it as "visited." d) Each ant randomly selects one feature from each feature group and stores it in queue α. Using equation (5), calculate probabilities for all features in α, marking the highest-probability feature as "visited." e) If the ant fails to satisfy the threshold condition (equation (8)), return to step d):

$$\exp(-\phi \cdot FN/FN_0) \cdot \exp(\omega \cdot \text{F-measure}) \geq A$$

where FN is the number of features currently selected by the ant colony, N is the total number of features, and ϕ and ω are parameters controlling the effects of feature size and F-measure with $\phi + \omega = 1$. Eventually, all ants successfully obtain their respective feature subsets.

Three pheromone update rules are considered: (a) each ant deposits pheromones on features; (b) pheromones evaporate over time; (c) the best ant has a decisive effect on pheromone updates, allowing other ants to follow high-quality paths. f) All ants in the colony execute steps c) through e). g) Each ant deposits pheromones on features along its path, marking nodes as "visited." The pheromone update method is calculated as:

$$\Delta\text{pher}(i,k) \propto \text{Acc\_model}(k) \cdot \left[\beta \cdot \frac{FC - FC(k)}{FC} + (1-\beta)\right]$$

where $\Delta\text{pher}(i,k)$ is the pheromone deposited by ant k on feature i, $\text{Acc\_model}(k)$ represents the classification accuracy of the model trained with ant k' s selected features, FC and $FC(k)$ are the total number of features and the number of features in ant k' s path, respectively. Parameters α and β control the effects of classification accuracy and feature subset length with $\alpha + \beta = 1$. h) Identify the ant with the highest F-measure value. i) Update pheromones. j) Remove existing ants and generate new ants randomly. k) If termination conditions are met, proceed to step l); otherwise, return to step c). l) The path with the maximum F-measure value represents the optimal solution.

[Figure 4: see original paper] shows the flowchart of the first feature selection scheme.

3.2.2 Different Feature Sets for Different Classes (Second Feature Selection Scheme) The first feature selection scheme selects an optimal feature set shared by all seven pixel classes. However, different pixel classes may have varying relevance to the same features. The second feature selection scheme extracts an optimized feature subset for each pixel class individually. Although this method requires longer computation time, it operates offline, ensuring real-time performance for the online program. [Figure 6: see original paper] illustrates the flowchart of the second feature selection scheme.

This approach establishes an optimal feature subset for each pixel class, ensuring each class has the most relevant features to improve overall pixel classification accuracy. The steps of the second feature selection scheme are: a) Execute steps b) through d) for each pixel class. b) Run the first feature selection scheme. c) Calculate the optimal feature subset for each class individually. d) Train a separate classifier for each pixel class.

4 Experimental Environment and Parameter Settings

The Mars Exploration Rover Spirit dataset serves as the benchmark, containing 94,257 images of size 512×512. [Figure 7: see original paper] shows three example images from this dataset. Following reference [17], the dataset is divided into seven classes as shown in [Figure 8: see original paper], using their classification results as prior knowledge.

The experimental environment consists of an Intel CPU at 2.26 GHz with 6 GB RAM. Each pixel' s window size is 21×21. Wavelet decomposition is computed for each window in grayscale images, with row and column norms calculated to obtain eight wavelet features per pixel (means and standard deviations). For color features, the dominant color quantity is set to 8, yielding 24 features across R, G, and B channels. Means and standard deviations of R, G, and B channels constitute six color statistical features. Using four bins from R, G, and B histograms produces 12 additional features. Consequently, each pixel has a 50-dimensional feature vector.

Seven ELM classifiers correspond to the seven classes, each representing one surface type. During training for each classifier, the corresponding surface type serves as positive samples while other pixels serve as negative samples. shows the positive and negative sample quantities for all seven classes, with 60% of pixels used for training and 40% for testing.

KNN, GA, and our algorithm were tested under different parameter settings to evaluate feature selection and pixel classification performance. The SVM classifier uses the Sequential Minimal Optimization (SMO) algorithm with a maximum of 20,000 iterations and kernel cache constraint of 1000. The multilayer perceptron range is [-0.01, 0.01] with a linear Gaussian kernel where $\sigma = 4$.

Several standard classification performance metrics evaluate our algorithm: precision, recall, F-measure, and accuracy. F-measure is defined as:

$$\text{F-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

where Precision denotes precision and Recall denotes recall rate.

5 Experimental Results

5.1 Results of the First Feature Selection Scheme

To compare classifier effectiveness, experiments analyze three classifiers: ELM, SVM, and KNN, using 13, 17, and 18 features respectively. [Figure 9: see original paper] shows the classification performance achieved by the three classifiers, demonstrating that ELM achieves the highest performance.

5.2 Second Feature Selection Scheme

The second feature selection algorithm extracts an optimized feature subset for each classifier individually. Experiments compare ELM and SVM classifiers, with results shown in [Figure 10: see original paper]. The results indicate that ELM uses slightly more features than SVM. [Figure 11: see original paper] presents classification accuracy results, showing that ELM outperforms SVM.

5.3 Algorithm Time Efficiency

Time efficiency is a critical performance metric for real-time remote sensing image processing algorithms. This paper considers both training and testing phase efficiencies. [Figure 12: see original paper] shows experimental results for training and testing times across three classifiers. While KNN requires no training, its classification accuracy is significantly lower than SVM and ELM. Overall, ELM with a linear kernel achieves the lowest computation time.

5.4 Remote Sensing Image Segmentation Results

Remote sensing image classification enables image segmentation. [Figure 13: see original paper] shows segmentation examples, visually demonstrating that our algorithm achieves good classification accuracy for remote sensing images.

* * *

6 Conclusion

Remote sensing images contain rich detail information with high feature dimensionality. Conventional feature extraction schemes often produce weak feature representation and severe information loss, negatively impacting subsequent processing. To enable classification processing at remote sensing image acquisition terminals, this paper proposes a real-time classification algorithm based on ant colony optimization and independent feature sets. The algorithm extracts wavelet and color features, optimizes feature quantities using ant colony optimization, and trains independent extreme learning machine classifiers for each image class. Experimental results demonstrate that the algorithm effectively ensures classification accuracy while significantly reducing processing time, facilitating real-time image preprocessing at remote sensing measurement terminals.

* * *

References

[1] Cheng Gong, Han Junwei, Lu Xiaoqiang. Remote sensing image scene classification: benchmark and state of the art [J]. Proceedings of the IEEE, 2017, 105(10): 1865-1883.

[2] Zhao Quanhua, Liu Dong, Li Xiaoli, et al. Fuzzy segmentation algorithm for remote sensing images using inclusion degree and membership degree [J]. Journal of Image and Graphics, 2017, 22(7): 988-995.

[3] Li Junlin, Zhang Liming, Si Xiaolong, et al. Scanning measuring system of stray light for optical remote sensing satellite [J]. Infrared and Laser Engineering, 2017, 46(9): 242-247.

[4] Li Deren, Shen Xin, Li Dilong, et al. On civil-military integrated space-based real-time information service system [J]. Geomatics and Information Science of Wuhan University, 2017, 42(11): 1501-1505.

[5] Romero Adriana, Gatta Carlo, Camps-Valls Gustau. Unsupervised deep feature extraction for remote sensing image classification [J]. IEEE Trans on Geoscience & Remote Sensing, 2016, 54(3): 1349-1362.

[6] Liu Dawei, Han Ling, Han Xiaoyong. High spatial resolution remote sensing image classification based on deep learning [J]. Acta Optica Sinica, 2016, 36(4): 0428001.

[7] Wang Li, Yang Zheng, Li Yang, et al. Retrieval technology of remote sensing image database based on feature aggregation [J]. Laser Journal, 2016, 37(6): 78-81.

[8] Zeng Rui, Chen Duansheng. Concatenating two deep learned spectral-spatial features hyperspectral remote sensing imagery classification [J]. Journal of Chinese Computer Systems, 2018, 39(2): 396-400.

[9] Wang Fengping, Wang Weixing, Xue Baiyu, et al. Road extraction from high-spatial-resolution remote sensing image by combining GVF Snake with salient features [J]. Acta Geodaetica et Cartographica Sinica, 2017, 12(12): 1978-1985.

[10] Wu Yiquan, Tao Feixiang, Cao Zhaoqing. Remote sensing image classification based on Log-Gabor wavelet and Krawtchouk moments [J]. Geomatics and Information Science of Wuhan University, 2016, 41(7): 861-867.

[11] Kashef S, Nezamabadi-Pour H. An advanced ACO algorithm for feature subset selection [J]. Neurocomputing, 2015, 147(1): 271-279.

[12] Zhang Yong, Gong Dunwei, Zhang Wanqiu. Feature selection of unreliable data using an improved multi-objective PSO algorithm [J]. Neurocomputing, 2015, 171(C): 1281-1290.

[13] Alexandre E, Cuadra L, Salcedo-Sanz S, et al. Hybridizing extreme learning machines and genetic algorithms to select acoustic features in vehicle classification applications [J]. Neurocomputing, 2015, 152(C): 354-362.

[14] Peng Yanfei, Li Jia. Remote sensing image retrieval based on SVM and genetic algorithm [J]. Journal of Chinese Computer Systems, 2016, 37(4): 875-880.

[15] Zhang Dan, Wang Yude, Feng Wei. A fusion feature retrieval algorithm based on wavelet feature contribution rate [J]. Laser Journal, 2018, 1(1): 110-113.

[16] Forsati R, Moayedikia A, Jensen R, et al. Enriched ant colony optimization and its application in feature selection [J]. Neurocomputing, 2014, 142(142): 354-371.

[17] Shang Changjing, Barnes D. Fuzzy-rough feature selection aided support vector machines for Mars image classification [J]. Computer Vision & Image Understanding, 2013, 117(3): 202-213.

Note: Figure translations are in progress. See original paper for figures.

Source: ChinaXiv —Machine translation. Verify with original.