

Cross-Scene Hyperspectral Feature Selection via Hybrid Whale Optimization Algorithm With Simulated Annealing

Jianxi Wang, Minchao Ye ¹, *Member, IEEE*, Fengchao Xiong ², *Member, IEEE*, and Yuntao Qian ³, *Member, IEEE*

Abstract—Hyperspectral images (HSIs) include hundreds of spectral bands, which lead to Hughes phenomenon in classification task and decrease the classification accuracy. Feature selection can remove redundant and noisy features in the HSIs to overcome this phenomenon. In real applications, we may face a HSI scene with only a few labeled samples. Meanwhile, there are adequate labeled samples in a similar HSI scene. For example, they share the same land-cover classes. The shared information can be used to help the scene with a few labeled samples in feature selection. Traditional single-scene-based feature selection appears powerless in solving such problems. Cross-scene feature selection provides an attractive way to select feature subsets by simultaneously using the information from two HSI scenes. However, spectral distribution may change due to atmospheric conditions, yielding spectral shift. In order to tackle this problem, we propose a cross-domain algorithm based on hybrid whale optimization algorithm with simulated annealing (WOASA). The newly proposed algorithm is dubbed cross-domain WOASA (CDWOASA). CDWOASA simultaneously considers the separability of different land-cover classes and the consistency of selected features between two scenes, leading to discriminative and domain-invariant characters of selected feature subset. Moreover, since the original WOASA is not able to precisely control the dimension of selected features, we propose an improvement using a sorting strategy based on the fitness function value, thus making the output feature dimension precisely controlled. The experimental results on two cross-scene HSI datasets demonstrate the superiority of CDWOASA in cross-scene feature selection.

Index Terms—Cross-domain WOASA, cross-scene feature selection, hyperspectral images.

Manuscript received October 28, 2020; revised December 29, 2020; accepted January 28, 2021. Date of publication February 2, 2021; date of current version February 22, 2021. This work was supported in part by the National Natural Science Foundation of China under Grants 61701468, 62071421, and 62002169, in part by the National Major Program for Technological Innovation 2030-New Generation Artificial Intelligence under Grant 2018AAA0100500, and in part by the Jiangsu Provincial Natural Science Foundation of China under Grant BK20200466. (*Corresponding author: Minchao Ye.*)

Jianxi Wang and Minchao Ye are with the Key Laboratory of Electromagnetic Wave Information Technology and Metrology of Zhejiang Province, College of Information Engineering, China Jiliang University, Hangzhou 310018, China (e-mail: p1903085225@cjlu.edu.cn; yeminchao@cjlu.edu.cn).

Fengchao Xiong is with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China (e-mail: fcxiong@njust.edu.cn).

Yuntao Qian is with the College of Computer Science, Zhejiang University, Hangzhou 310027, China (e-mail: yuqian@zju.edu.cn).

Digital Object Identifier 10.1109/JSTARS.2021.3056593

I. INTRODUCTION

HYPERSPECTRAL images (HSIs) consist of many narrow and continuous bands and they contain both spatial and spectral information [1], [2]. A great challenge of HSI classification is the high feature dimension. Redundant and noisy features have a negative impact on classification. High feature dimension leads to huge data scale and takes a lot of calculation time [3]. Moreover, in real remote sensing applications, only very few training samples are available. The high dimension and small sample size problem leads to the Hughes phenomenon, which affects classification accuracy [4]. Dimensionality reduction has proven necessary in classification algorithms [5]. By eliminating redundant and noisy features, feature selection can reduce the high feature dimension, speed up the learning process, and simplify the classifier [6].

Feature selection selects a small subset from the full feature set that contains as much discrimination information as possible. Feature selection methods are grouped into three main categories: the filter method, the wrapper method, and the embedded method. In filter-based feature selection methods, criterion function is employed to evaluate the effectiveness of feature subsets based on statistical distance measures or information measures, e.g., information gain (IG) [7], mutual information [8], ReliefF [9], and so on. In wrapper-based feature selection methods, criterion function is employed to evaluate the effectiveness of feature subsets through a certain classifier, e.g., K-nearest neighbors (KNN) and support vector machine (SVM) [10]. In embedded methods, feature selection is in the process of training a classifier. Since wrappers are more likely to search for a feature subset which suits a classifier, wrappers have been widely invested and applied.

How to search for the optimal feature subset from the original feature set is the main problem that feature selection solves. If the dimension of the original feature set is N , then there would be 2^N feature subsets [11]. This is a typical NP-hard problem. The exhaustive method is inappropriate. Because when the dimension of features is high, the exhaustive method would take a lot of time. It is a good choice to use population intelligence with meta-heuristic search as a search strategy, since meta-heuristic algorithms have achieved remarkable results in the field of feature selection in recent years. Meta-heuristic algorithms solve optimization problems by mimicking biological or physical phenomena [12]. The meta-heuristic methods may not

be able to find the optimal solution, but they can find satisfactory results in limited time.

In [10], genetic algorithm (GA) was proposed to solve feature selection problem in a wrapper way. GA is more capable of avoiding getting stuck in local optima, since the mutation and crossover operators are used to generate the next solution [10]. Differential evolution (DE) algorithm [13], particle swarm optimization (PSO) algorithm [14], [15], gray wolf optimizer [16], ant lion optimizer [17], and whale optimization algorithm (WOA) [12] are also used to search for a subset of features. They are all population-based methods and have a strong ability of searching the feature space. In order to further improve the ability of exploiting the best solution, some meta-heuristic algorithms based on single solution, like local search and simulated annealing (SA) algorithm, are also proposed [18]. A natural idea for improving the performance of feature selection is to combine a population-based algorithm with a single-solution-based algorithm. This kind of algorithms is also named memetic algorithms. Hybrid WOA with SA (WOASA) algorithm [19] is one of them, which combines WOA and SA algorithms. WOASA had proved its superiority on 18 standard benchmark datasets from UCI repository [19]. In [20], hybrid WOA based on local search strategy was used for the permutation flow shop scheduling problem. The excellent performance of memetic algorithms is attributed to the parallel searching power and exploration of the best solution.

In the field of HSI, many methods have been proposed for feature selection. In [21], an embedded feature selection algorithm was proposed using SVMs to perform feature selection and classification at the same time. Yuan *et al.* [22] proposed a multitask sparsity pursuit framework for HSI feature selection. In [23], hybridization of GA and PSO (HGAPSO) algorithm had achieved a good performance on the well-known Indian Pines hyperspectral dataset. In HGAPSO, PSO was for local search, while GA was employed for global optimization [23]. In [24], high-dimensional model representation method was proposed for feature selection. In [25], a multiobjective feature selection approach was proposed for HSIs based on discrete sine cosine algorithm. Unsupervised methods are also proposed for feature selection. In [26], affinity propagation was employed to achieve the goal of unsupervised band selection. In [27], a robust multifeature spectral clustering method was proposed and had a good performance on HSI datasets.

Since labeling samples is labor-consuming, many HSI scenes contain a few or no labeled samples. As the number of HSIs increases, we can find similar HSI scenes, e.g., the classes of land-cover objects they record are similar and they are taken by the same sensor [28]. We can use shared information, which is the class-specified distribution of data, to help the scene with a few labeled samples in feature selection. Therefore, cross-scene feature selection is proposed to solve the feature selection of a HSI scene with limited labeled samples (namely target scene) with the help of a similar HSI scene with plenty of labeled samples (namely source scene). The main challenge faced by cross-scene feature selection is spectral shift [29]. Spectral shift refers to the changes in spectral distribution caused by atmospheric conditions, ground conditions, and so on. Even though two HSIs are captured by the same sensor and the classes

of land-cover objects are the same, spectral shift still exists. Due to the existence of spectral shift, we cannot directly use the labeled samples of the source scene to train a classifier. Hence, it is meaningful to take some effective measures to overcome spectral shift. In this case, the consistency of selected features between two scenes as well as the separability of different land-cover classes must be considered. Some methods like transfer learning [30] and cross-view learning have been proposed to solve this kind of problem. In [31], ReliefF was extended to a cross-scene version, and experimental results on two cross-scene datasets verified the superiority of the cross-scene feature selection method. In [32], a domain adaptation method named hyperspectral feature adaptation and augmentation was proposed to extract the transferable knowledge by learning a common subspace. In [33], a generalized similarity model is used to improve the accuracy of cross-domain feature matching. In [34], a novel dataset shift measure was proposed to compute the distance of the conditional distributions between the source and target domains in a reproducing kernel Hilbert space. In [35], a unsupervised domain adaptation approach based on the class centroid and covariance alignment was proposed to solve the classification problem of remote sensing images.

WOASA has strong abilities of exploring the feature space and exploiting the best solution. Furthermore, it has the advantages of fewer parameters and high flexibility. In WOASA, feature subsets are evaluated by fitness function which considers the classification accuracy and selected feature dimension. In this article, we introduce WOASA to solve cross-domain feature selection problem. The novel algorithm is named cross-domain WOASA (CDWOASA). The contributions of this article include the following.

- 1) Aiming at selecting a discriminative and domain-invariant feature subset, the original WOASA is adapted to its cross-domain version, i.e., CDWOASA. The newly developed fitness function in CDWOASA considers the accuracies achieved by target training samples and source training samples to ensure the separability of different land-cover classes and the consistency of selected features between two scenes.
- 2) Based on WOASA, a sorting-based strategy is proposed to make the dimension of the selected feature subset precisely controllable.

The rest of this article is organized as follows. Section II presents the basics of WOA, SA, and WOASA. Section III presents a cross-scene feature selection algorithm CDWOASA. In Section IV, the experimental results on two cross-scene HSI datasets are presented and analyzed, which show the superiority of the proposed CDWOASA. Finally, the conclusion is drawn in Section V.

II. RELATED WORKS

This section introduces WOA, SA, and WOASA in detail.

A. Whale Optimization Algorithm (WOA)

WOA is a meta-heuristic algorithm, which was first proposed by Mirjalili and Lewis in 2016 [12]. WOA mimics the bubble-net

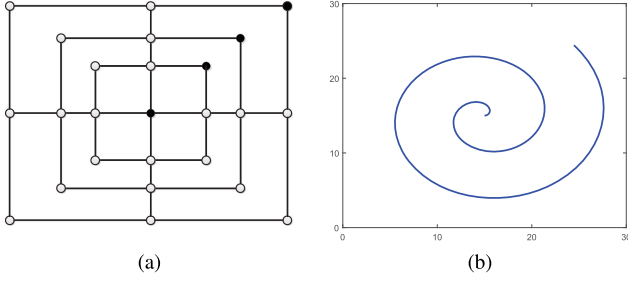


Fig. 1. Bubble-net feeding mechanism in WOA. (a) Shrinking encircling mechanism. (b) Spiral updating position.

foraging behavior of humpback whales, and the mathematical model of this special mechanism is shown in Fig. 1.

In order to mimic the encircling prey mechanism, WOA defines the target prey or one candidate solution which is close to the optimum as the current best search agent (solution). Then, the humpback whales update their positions to approach the current best search agent. It should be mentioned that the position of the current best search agent would be replaced if a better solution appears. This behavior can be represented by

$$\vec{D} = |C\vec{X}^*(t) - \vec{X}(t)| \quad (1)$$

$$\vec{X}(t+1) = \vec{X}^*(t) - A\vec{D} \quad (2)$$

where t indicates the current iteration. \vec{X} is the position vector. \vec{X}^* means the best search agent obtained so far. \vec{D} is the absolute difference between $C\vec{X}^*$ and \vec{X} . In addition, A and C are coefficients that are given by

$$A = 2ar - a \quad (3)$$

$$C = 2r \quad (4)$$

where a decreases linearly from 2 to 0 along with iterations (in both exploration and exploitation phases). r is a random number uniformly distributed in the interval $[0, 1]$. By adjusting the values of A and C , different positions around the best search agent relative to the current position can be achieved. The shrinking encircling behavior is achieved by decreasing the value of a in (3) according to

$$a = 2 - t \frac{2}{M} \quad (5)$$

where M is the maximum number of allowed iterations.

In order to model the spiral updating position mechanism, the absolute difference between \vec{X} and \vec{X}^* is calculated. Then, the position of the neighbor search agent is created by a spiral equation as

$$\vec{X}(t+1) = \vec{D}' e^{bl} \cos(2\pi l) + \vec{X}^*(t) \quad (6)$$

where $\vec{D}' = |\vec{X}^*(t) - \vec{X}(t)|$ and represents the absolute difference between \vec{X} and \vec{X}^* , and l is a random number in $[-1, 1]$. Moreover, b is a constant for defining the shape of the spiral and is set to 1 in this article. Both encircling prey and the spiral updating position have a probability of 50% to be chosen during

the optimization process as

$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - A\vec{D} & p < 0.5 \\ \vec{D}' e^{bl} \cos(2\pi l) + \vec{X}^*(t) & p \geq 0.5 \end{cases} \quad (7)$$

where p is a random number in $[0, 1]$.

In exploration phase, a random search agent is selected to update the position of the current search agent instead of using the current optimal search agent to improve the ability of exploring feature space. When the value of $|A|$ is greater than 1, the current search agent is forced to move far away from the best known search agent. Equations (8) and (9) are used to represent this mechanism

$$\vec{D}'' = |C\vec{X}_{\text{rand}} - \vec{X}| \quad (8)$$

$$\vec{X}(t+1) = \vec{X}_{\text{rand}} - A\vec{D}'' \quad (9)$$

where \vec{X}_{rand} is a random whale chosen from the current population. In this article, we assume that a , r , C , and A are scalars following [20].

B. Simulated Annealing (SA)

Algorithm 1: SA.

Input:

Number of iterations Itr_{SA}^{\max} .

Output:

Bestsol.

- 1: $T \leftarrow T_0$.
 - 2: Generate *Cursol* randomly.
 - 3: $Bestsol \leftarrow Cursol$.
 - 4: $\delta(Bestsol) \leftarrow \delta(Cursol)$.
 - 5: **for** $j = 1, 2, 3, \dots, Itr_{SA}^{\max}$ **do**
 - 6: Generate a *Newsol* randomly around *Cursol*.
 - 7: Calculate $\delta(Newsol)$.
 - 8: **if** $\delta(Newsol) > \delta(Bestsol)$ **then**
 - 9: $Cursol \leftarrow Newsol$.
 - 10: $Bestsol \leftarrow Newsol$.
 - 11: $\delta(Cursol) \leftarrow \delta(Newsol)$.
 - 12: $\delta(Bestsol) \leftarrow \delta(Newsol)$.
 - 13: **else**
 - 14: $\theta = \delta(Bestsol) - \delta(Newsol)$.
 - 15: Randomly generate a number P from 0 to 1.
 - 16: **if** $P > e^{-\frac{\theta}{T}}$ **then**
 - 17: $Cursol \leftarrow Newsol$.
 - 18: $Bestsol \leftarrow Newsol$.
 - 19: $\delta(Cursol) \leftarrow \delta(Newsol)$.
 - 20: $\delta(Bestsol) \leftarrow \delta(Newsol)$.
 - 21: **end if**
 - 22: **end if**
 - 23: $T \leftarrow 0.99T$.
 - 24: **end for**
 - 25: **return** *Bestsol*
-

SA is derived from the principle of solid annealing and was first proposed by Kirkpatrick *et al.* [18]. It is a random

optimization algorithm based on hill climbing method. Since a worse result is accepted with a certain probability in each iteration, SA can effectively avoid falling into the problem of local optima. An initial solution is randomly generated at the beginning of the algorithm. Then, a new solution (*Newsol*) is searched and evaluated around the current solution (*Cursol*) in each iteration. Once the fitness of a new solution is better than the fitness of current best solution (*Bestsol*), the current best solution is always replaced by the new solution. Otherwise, the new solution is accepted with the Boltzmann probability. So the probability of accepting the new solution can be represented as

$$P = \begin{cases} 1 & \delta(\text{Newsol}) \geq \delta(\text{Bestsol}) \\ e^{-\frac{\theta}{T}} & \delta(\text{Newsol}) < \delta(\text{Bestsol}) \end{cases} \quad (10)$$

where δ represents the fitness of a solution and θ is the difference between $\delta(\text{Newsol})$ and $\delta(\text{Bestsol})$. T represents temperature and decreases along with the iterations. In this work, the initial temperature (T_0) is defined as 0.1. Moreover, the current temperature is calculated as $T = 0.99 T$. It should be noted that the probability of accepting a worse solution declines along with the iterations to ensure convergence. And, the number of allowed iterations in SA algorithm is set to 30 in this article. The pseudo code of SA is shown in Algorithm 1.

C. Hybrid Whale Optimization Algorithm With Simulated Annealing (WOASA)

WOASA is a wrapper-based feature selection method which combines WOA and SA algorithms. SA algorithm is used to find the optimal solution after WOA is finished. WOASA employs a fitness function to evaluate each search agent in each iteration and then the desired feature subset is selected. In WOASA, current solution $\vec{X}(t)$ is used to represent the selected feature subset. For the elements in $\vec{X}(t)$ which are greater than 0.5, their corresponding features are selected. And, the other features are dropped. In order to consider the classification accuracy and selected feature dimension at the same time, fitness function is calculated as

$$\delta = \alpha\gamma + (1 - \alpha)\frac{W}{N} \quad (11)$$

where $\alpha \in [0, 1]$ balances the relationship between the importance of classification accuracy and selected feature dimension. γ is the error rate of classification with a certain classifier. And, W represents the number of selected features while N is the number of total features. It should be noted that by this definition of the fitness, a smaller fitness leads to a better solution.

Tournament selection is one of the most well-regarded selection strategies [36]. In tournament selection, two random search agents are selected at first. Then, a random number between 0 and 1 is generated. After that, this number is used to compare with a fixed probability (mostly is defined as 0.5). When the random number is greater than the fixed probability, the solution with higher fitness value is accepted. Otherwise the weak solution is accepted. Tournament selection gives the weak solution more chances to be selected. Since tournament selection improves the ability of exploring the feature space,

Algorithm 2: WOASA.

Input:

- Number of iterations M ,
- Number of search agents n .

Output:

- Position of the best search agent \vec{X}^* .

- 1: Generate initial population \vec{X} .
 - 2: Calculate the fitness of each agent using (11).
 - 3: Find the current best search agent \vec{X}^* .
 - 4: **for** $t = 1, 2, \dots, M$ **do**
 - 5: **for** $i = 1, 2, \dots, n$ **do**
 - 6: Randomly generate l, r , and p .
 - 7: Calculate parameters a, A , and C using (3)–(5).
 - 8: **if** $p < 0.5$ **then**
 - 9: **if** $|A| < 1$ **then**
 - 10: Employ mutation operation to get \vec{X}^{Mut} from \vec{X}^* (best solution) with rate y using (12).
 - 11: Apply crossover between \vec{X}^{Mut} and $\vec{X}(t)$ to get the new position of $\vec{X}(t+1)$ as output of crossover using (13).
 - 12: **else**
 - 13: Select a search agent \vec{X}_{rand} using tournament selection.
 - 14: Employ mutation operation to get \vec{X}^{Mut} from \vec{X}_{rand} (best solution) with rate y using (12).
 - 15: Apply crossover between \vec{X}^{Mut} and $\vec{X}(t)$ to get the new position of $\vec{X}(t+1)$ as output of crossover using (13).
 - 16: **end if**
 - 17: **else**
 - 18: Update the position of current solution using (6).
 - 19: **end if**
 - 20: **end for**
 - 21: SA (Algorithm 1) is used to find the best solution around the current best search agent.
 - 22: Check if all agents are in the search space.
 - 23: Calculate the fitness of each agent using (11).
 - 24: Once a better solution appears, update \vec{X}^* .
 - 25: **end for**
 - 26: **return** \vec{X}^*
-

randomly selecting a search agent to updating the position of next move is replaced by tournament selection.

In WOA, (2) and (9) are used to determine the next move from a randomly selected solution and the current best solution, respectively. To further improve the exploration of feature space, the mutation and crossover operators are employed to determine the position of next move instead of (2) and (9) in WOASA. The mutation rate y is calculated as

$$y = 0.9 + \frac{-0.9 \times (t - 1)}{M - 1} \quad (12)$$

where M is the maximum number of iterations, and t means the current number of iteration. And, the mutation rate y linearly

decreases from 0.9 to 0 as the number of iterations increases. The crossover operation is performed between the current solution $\vec{X}(t)$ and the resultant solution from the mutation operation. This operation is given as

$$\vec{X}(t+1)_d = \begin{cases} \vec{X}(t)_d & z < 0.5 \\ X^{\text{Mut}}_d & z \geq 0.5 \end{cases} \quad (13)$$

where X^{Mut} is the result from the mutation operation and $\vec{X}(t+1)$ is the newly generated solution. Moreover, z is a random number in $[0, 1]$ and $\vec{X}(t+1)_d$ is the d th dimension in $\vec{X}(t+1)$. The pseudo code of WOASA is shown in Algorithm 2. As seen from Algorithm 2, SA is employed when WOA is finished in every iteration. Moreover, it should be mentioned that the mutation operator is employed to generate a new solution in SA. Since the parameter a gradually decreases with the number of iterations, WOASA can achieve good convergence.

III. PROPOSED APPROACH

Algorithm 3: The Calculation of Fitness Function in CDWOASA.

Input:

Solution \vec{X} ,
The dimension of selected features W .

Output:

The fitness $\delta(\vec{X})$.

- 1: Sort the elements in \vec{X} in descending order.
 - 2: Select the top W features in \vec{X} .
 - 3: Train a classifier to calculate $Acc1$ using source training samples (see Table I).
 - 4: Train another classifier to calculate $Acc2$ using randomly selected half of target training samples (see Table I).
 - 5: Calculate $\delta(\vec{X})$ using (14).
 - 6: **return** $\delta(\vec{X})$
-

Traditional HSI feature selection algorithms are usually based on a single HSI scene with sufficient labeled samples. WOASA can be employed to such problems. However, when we face the cross-scene feature selection problem, WOASA is not applicable. So we introduce WOASA to solve cross-domain feature selection problem and make adaptations to it. The new proposed method is named cross-domain WOASA (CDWOASA). As a transfer learning algorithm, CDWOASA considers the information from source and target scenes at the same time. In order to effectively overcome the influence of spectral shift, the proposed cross-scene feature selection method considers two factors: 1) the consistency of the selected features between two scenes; and 2) the separability among different land-cover classes. Following this idea, the fitness function in CDWOASA considers two accuracies, namely $Acc1$ and $Acc2$. The calculations of $Acc1$ and $Acc2$ are listed in Table I. $Acc1$ can reflect the consistency of the selected features between two scenes while $Acc2$ can represent the separability among different land-cover classes on

target scene. The greater $Acc1$ and $Acc2$ are, the better selected features are. Based on Table I, the fitness function used in the proposed model is defined as

$$\delta = Acc1 + uAcc2 \quad (14)$$

where u is the parameter to balance the relationship between $Acc1$ and $Acc2$ and is set to 8 in this article. A greater u means that the separability among different land-cover classes on target scene is more important while a smaller u means that the consistency of the selected features between two scenes takes more attention. It should be noted that by this definition of the fitness, a bigger fitness leads to a better solution. The pseudo code of calculation of fitness function is shown in Algorithm 3. As seen from Algorithm 3, two classifiers are trained and a hybrid fitness function is calculated to evaluate one certain solution.

In WOASA, the fitness function is used to evaluate solutions considered both classification accuracy and the dimension of selected features. The parameter α can only roughly adjust the dimension of selected features, rather than accurately determine. Therefore, when facing datasets with different dimensions, parameter α may change greatly. However, we want to be able to precisely control the number of selected features. In this article, a sorting-based strategy is proposed to control the dimension of the selected feature subset. We sort the current solution $\vec{X}(t)$ in descending order. Then, the top W features are selected, where W is the dimension of selected features. Since sorting-based strategy is proposed to control the number of selected features, the dimension of the selected features can be determined in advance. Moreover, the term $(1 - \alpha)(W)/(N)$ in the fitness function (11) becomes a constant since W is fixed in CDWOASA. Hence, it is dropped in the fitness function (14).

In CDWOASA, search agents with M features are randomly generated using tournament selection at the beginning. It should be mentioned that SA algorithm is only called in each iteration to search the neighborhood near the current best solution in original WOASA. In order to improve the ability of exploiting the best solution, CDWOASA also adds SA as an operator when a random search agent is selected. An acceptable solution is generated after several iterations. The pseudo code of CDWOASA is shown in Algorithm 4. The flowchart of CDWOASA is shown in Fig. 2.

IV. EXPERIMENTS

A. Dataset

In order to test the performance of the proposed method, experiments are conducted on two real-world cross-scene HSI datasets. The first one is Indiana dataset. Indiana dataset was captured by AVIRIS sensor over the Northwest Tippecanoe County, Indiana. Both the source and target scenes have 220 bands and 400×300 pixels with seven different land-cover classes which are listed in Table II. The data cubes and ground truth maps are shown in Fig. 3. The other cross-scene HSI dataset is named Pavia, which was captured by the airborne hyperspectral sensor DAIS. Pavia dataset contains two scenes, namely Pavia University and Pavia Center. Pavia University dataset (sized $243 \times 243 \times 72$) is regarded as the source scene,

TABLE I
CALCULATIONS OF $Acc1$ AND $Acc2$ (OVERALL ACCURACIES)

	Training samples	Test samples
$Acc1$	Source training samples	Target training samples
$Acc2$	Randomly selected half of the target training samples	The remaining half of the target training samples

Algorithm 4: CDWOASA.

Input:

- Number of iterations M ,
- Number of search agents n ,
- Selected feature dimension W .

Output:

Position of the best search agent \vec{X}^* .

- 1: Generate initial population \vec{X} .
- 2: Calculate the fitness of each agent using (14) by applying Algorithm 3.
- 3: Find the current best search agent \vec{X}^* .
- 4: **for** $t = 1, 2, \dots, M$ **do**
- 5: **for** $i = 1, 2, \dots, n$ **do**
- 6: Randomly generate l, r , and p .
- 7: Calculate parameters a, A , and C using (3)–(5).
- 8: **if** $p < 0.5$ **then**
- 9: **if** $|A| < 1$ **then**
- 10: Employ mutation operation to get \vec{X}^{Mut} from \vec{X}^* (best solution) with rate y using (12).
- 11: Apply crossover between \vec{X}^{Mut} and $\vec{X}(t)$ to get the new position of $\vec{X}(t+1)$ as output of crossover using (13).
- 12: **else**
- 13: Select a search agent \vec{X}_{rand} using tournament selection.
- 14: Employ mutation operation to get \vec{X}^{Mut} from \vec{X}_{rand} (best solution) with rate y using (12).
- 15: Apply crossover between \vec{X}^{Mut} and $\vec{X}(t)$ to get the new position of $\vec{X}(t+1)$ as output of crossover using (13).
- 16: SA (Algorithm 1) is used to find the best solution around the current search agent.
- 17: **end if**
- 18: **else**
- 19: Update the position of current solution using (6).
- 20: **end if**
- 21: **end for**
- 22: SA (Algorithm 1) is used to find the best solution around the current best search agent.
- 23: Check if all agents are in the search space.
- 24: Calculate the fitness of each agent using (14) by applying Algorithm 3.
- 25: Once a better solution appears, update \vec{X}^* .
- 26: **end for**
- 27: **return** \vec{X}^*

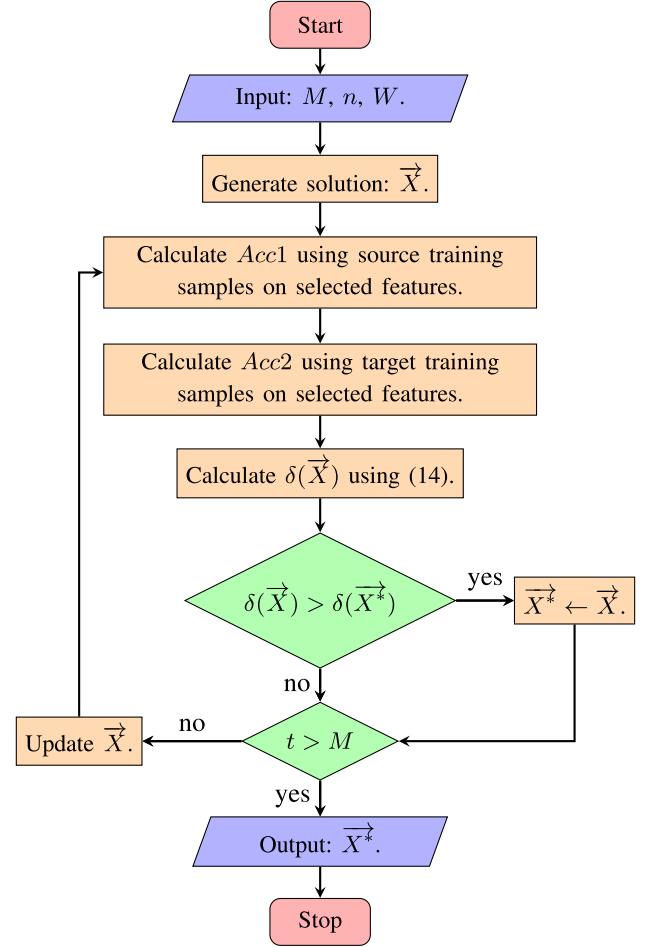


Fig. 2. Flow chart of CDWOASA.

TABLE II
NUMBER OF LABELED SAMPLES IN EACH LAND COVER CLASS WITHIN INDIANA SCENES

	Class		Labeled samples	
	Name	Source scene	Target scene	
1	Concrete/Asphalt	4867	2942	
2	Corn-CleanTill	9822	6029	
3	Corn-CleanTill-EW	11414	7999	
4	Orchard	5106	1562	
5	Soybeans-CleanTill	4731	4792	
6	Soybeans-CleanTill-EW	2996	1638	
7	Wheat	3223	10739	

while Pavia Center dataset (sized $400 \times 400 \times 72$) is adopted as target scene. The data cubes and ground truth maps are illustrated in Fig. 4. There are six land-cover classes shared by source and target scenes, which are listed in Table III.

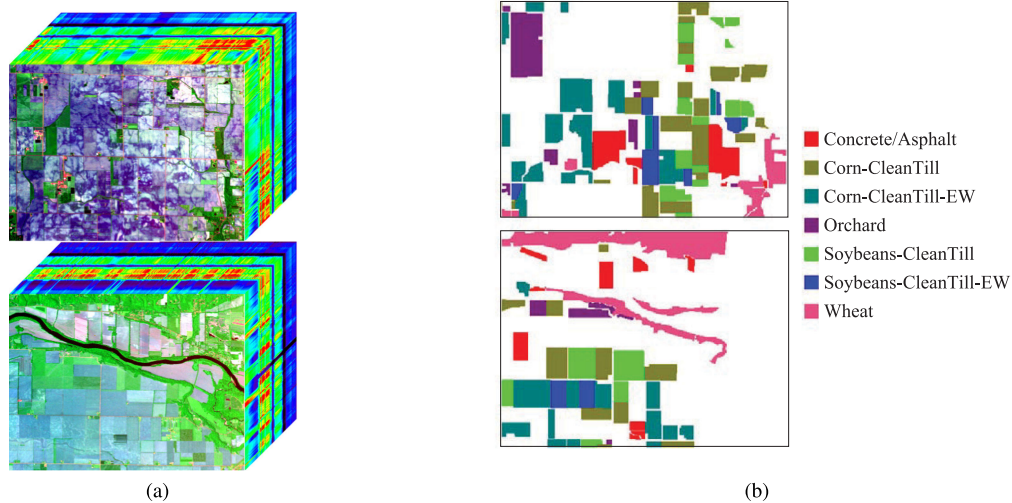


Fig. 3. Data cubes and ground truth maps of Indiana dataset. (a) Data cubes. (b) Ground truth maps. The upper one is the source scene, while the lower one is the target scene.

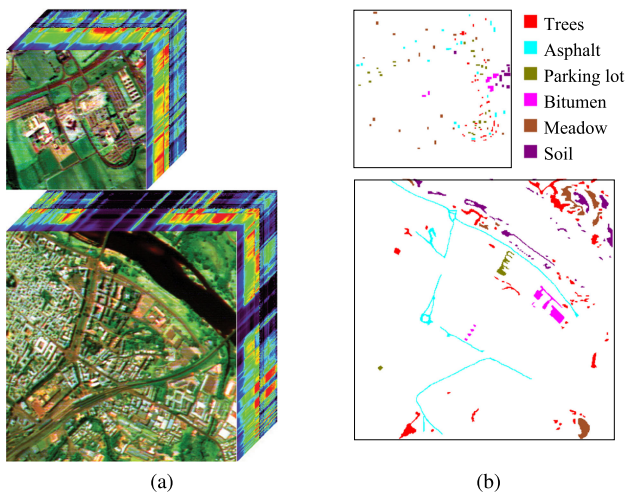


Fig. 4. Ground cubes and ground truth maps of Pavia dataset. (a) Data cubes. (b) Ground truth maps. The upper one is Pavia University (source scene), while the lower one is Pavia Center (target scene).

TABLE III
NUMBER OF LABELED SAMPLES IN EACH LAND COVER CLASS
WITHIN PAVIA SCENES

Class	Labeled samples		
	Name	Source scene	Target scene
1	Trees	266	2424
2	Asphalt	266	1704
3	Parking lot	265	287
4	Bitumen	206	685
5	Meadow	273	1251
6	Soil	213	1475

For both cross-scene datasets, we randomly select 200 labeled samples per class from source scene and five labeled samples per class from target scene as training samples. The rest of labeled samples in target scene are used as test samples. It should be mentioned that labeled samples in source scene are only used in the process of feature selection. The final classifier is trained

by the target training samples and makes prediction on the remaining labeled samples of target scene.

B. Parameter Settings

To validate the superiority of the proposed method, KNN (where $k = 1$) with the Euclidean distance is adopted as the classifier. In the proposed approach, there are two key parameters: the number of iteration M and the number of population n . M is set to 100, while n is set to 5 in the experiments. Furthermore, each algorithm is repeated ten times to ensure the reliability of the experimental results. Each method is evaluated by overall classification accuracy (OA), average accuracy (AA), and kappa coefficient (κ) on selected feature subset. In each comparison, the number of selected features is the same.

In order to validate the superiority of the proposed method, the following feature selection algorithms are compared.

- 1) Target-Domain ReliefF (TDRF): ReliefF is performed on the samples of target scene. In each iteration, for a randomly selected sample, the Euclidean distance is used to find the nearest neighbors in each class. Then, each feature is evaluated based on its impact on classification. Finally, the optimal feature subset is selected by ranking the weights of all features [9].
- 2) Cross-Domain ReliefF (CDRF): The labeled samples of both source and target scenes are used in CDRF algorithm. Like TDRF, CDRF selects a random sample in the target scene and finds the closest neighbors in each class in the target scene. Furthermore, CDRF also needs to use the Euclidean distance to find the closest neighbors in each class in source scene. When evaluating features, the impact on classification in both source scene and target scenes is considered. Finally, the weights of all features are ranked and the optimal subset is selected [31].
- 3) Target-Domain WOASA (TDWOASA): WOASA is performed on labeled samples of target scene [19].

TABLE IV
SUMMARY OF COMPARED METHODS

Algorithm	Source/target information that is used	Main parameters
TDRF	Target	<ul style="list-style-type: none"> Number of the nearest neighbors: $s = 1, 2, 3, 4$; Number of iterations: $M \in \{50, 100, \dots, 300\}$.
CDRF	Source and target	<ul style="list-style-type: none"> Number of the nearest neighbors: $s = 1, 2, 3, 4$; Number of iterations: $M \in \{50, 100, \dots, 300\}$.
TDWOASA	Target	<ul style="list-style-type: none"> Number of search agents: $n = 5$; Number of iterations: $M = 100$.
CDWOASA	Source and target	<ul style="list-style-type: none"> Number of search agents: $n = 5$; Number of iterations: $M = 100$.

TABLE V
ACCURACIES ON INDIANA AND PAVIA DATASETS ACHIEVED
BY TDWOASA AND CDWOASA

Dataset	Feature selection	OA	AA	κ	Feature dimension
Indiana	TDWOASA	0.5091	0.5023	0.4082	85
	CDWOASA	0.5145	0.5036	0.4131	85
Pavia	TDWOASA	0.8503	0.8510	0.8121	16
	CDWOASA	0.8708	0.8704	0.8373	16

The values in bold are the highest OA/AA/kappa achieved in each dataset.

4) Cross-Domain WOASA (CDWOASA): CDWOASA is the proposed algorithm in this article. CDWOASA is performed on both source and target scenes.

The main characteristic and parameters of above algorithms are summarized in Table IV. As seen from Table IV, we retrieve all combinations of s and M in CDRF and TDRF. Then, the highest accuracy is regarded as the final accuracy with a certain feature size. OA, AA, and κ are calculated, respectively. In the TDWOASA, OA and the number of selected features are used in fitness function. After selecting the optimal solution, AA, and κ of the optimal feature subset are calculated. In the CDWOASA, $Acc1$ and $Acc2$ are used in fitness function. After selecting the desired feature subset, the AA and κ of the desired feature subset are calculated.

C. Experimental Results

TDWOASA uses a parameter α (which is set to 0.01 in the experiments) in fitness function to make the dimension of selected features controllable, rather than precisely determined. In other words, the dimension of selected feature subset by TDWOASA is not *a priori*. So to make sure that the number of selected features is the same in each comparison, experiments are divided into two parts. One part is the comparison between CDWOASA and TDWOASA to validate the superiority of cross-domain feature selection method. Since the output feature dimension cannot be determined in advance in TDWOASA, while CDWOASA sets the number of selected features in advance, TDWOASA is performed first to determine the dimension of selected feature subset. Then, CDWOASA is performed with the same number of selected features. On the other hand, CDWOASA is compared with the other feature selection algorithms when the number of selected features is taken as $W \in \{5, 10, \dots, 25\}$, respectively.

First, we conduct a comparison between CDWOASA and TDWOASA. Table V shows the comparison between the classification accuracies of CDWOASA and TDWOASA. It can be seen from Table V that CDWOASA is more advantageous

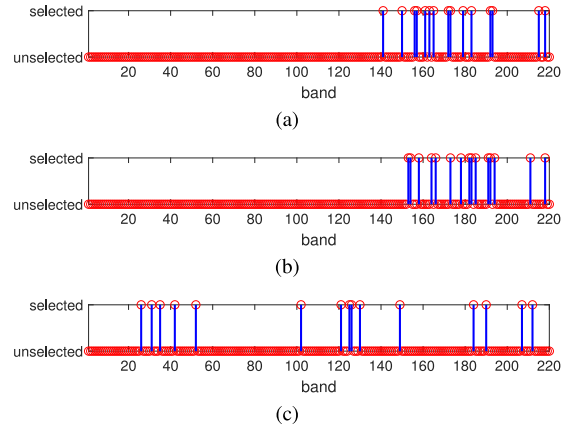


Fig. 5. Selected features on Indiana dataset. (a) TDRF. (b) CDRF. (c) CDWOASA.

than TDWOASA in both Indiana and Pavia datasets. The experimental results also prove that the performance of feature selection methods is more competitive when the information in source scene is added. Moreover, too many features are selected by TDWOASA which make TDWOASA no longer meaningful in feature selection. Hence, controlling the number of selected features is necessary and important.

Next is the comparison between CDWOASA and other algorithms in the Indiana dataset. When the number of selected features is 15, the selection results are shown in Fig. 5. It can be seen that the feature subset selected by the CDWOASA is very dispersed, while the feature subsets selected by the CDRF and TDRF algorithms contain many continuous bands. Since continuous bands may contain more redundant information, we believe that the feature subset selected by CDWOASA is better than CDRF and TDRF. Classification accuracies with respect to the number of selected features on Indiana dataset are shown in Fig. 6. As seen from Fig. 6, the classification accuracies of CDRF and TDRF are similar, and CDRF even achieves a little lower accuracies than TDRF. The results seem strange, which do not match with the original work [31]. Actually, it is caused by the sensitivity to normalization in CDRF. In [31], sample-wise normalization was applied, i.e., the norm of each spectral vector is first normalized to 1, while in this article, for fair comparison, we skip the sample-wise normalization on spectral vector, and only apply feature-wise Z-score normalization for all algorithms, which is more generally applied in classification problems. The proposed CDWOASA achieves significantly higher accuracies

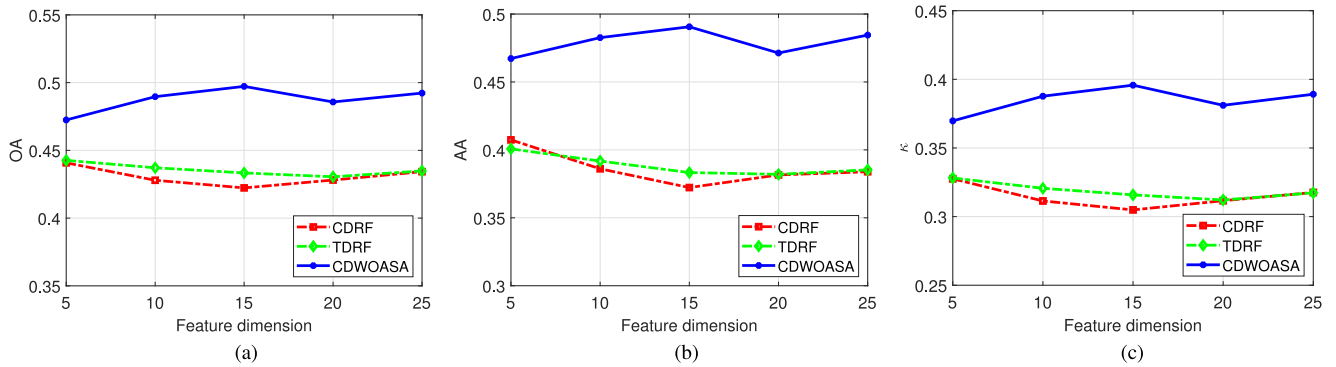


Fig. 6. Accuracies on Indiana dataset obtained by TDRF, CDRF, and CDWOASA. (a) OA. (b) AA. (c) κ .

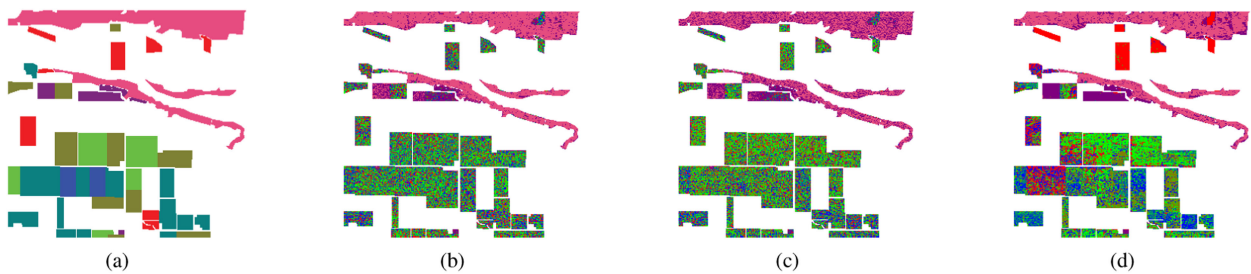


Fig. 7. Ground truth and classification maps of Indiana dataset. (a) Ground truth map. (b) The classification map over the selected features of TDRF. (c) The classification map over the selected features of CDRF. (d) The classification map over the selected features of CDWOASA.

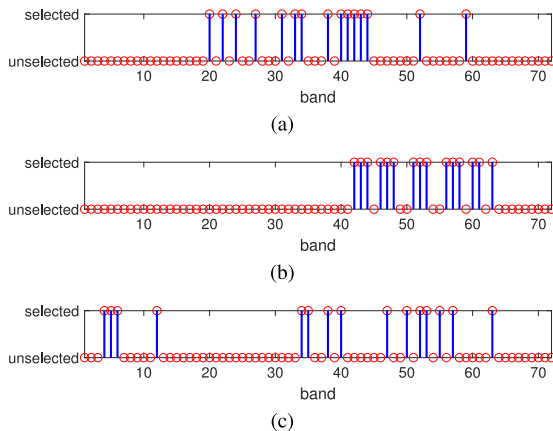


Fig. 8. Selected features on Pavia dataset. (a) TDRF. (b) CDRF. (c) CDWOASA.

than both CDRF and TDRF. The classification accuracies of CDWOASA gradually increase when the number of selected features increases within the range [5,15]. CDWOASA gets its highest accuracy when 15 features are selected. The ground truth and classification maps of Indiana dataset are shown in Fig. 7. Combining Figs. 3 and 7, we can find CDWOASA has advantages on the land-cover class Concrete/Asphalt.

In the Pavia dataset, when the number of selected features is 15, the selection results are depicted in Fig. 8. Fig. 8 reveals that TDRF and CDRF select many continuous bands, which may lead to feature redundancy. And the bands selected by CDWOASA

TABLE VI
ACCURACIES ON INDIANA AND PAVIA DATASETS

Dataset	Feature selection	OA	AA	κ
Indiana	TDRF	0.4357	0.3886	0.3188
	CDRF	0.4306	0.3862	0.3146
	CDWOASA	0.4875	0.4793	0.3846
Pavia	TDRF	0.8495	0.8462	0.8115
	CDRF	0.8371	0.8322	0.7955
	CDWOASA	0.8592	0.8573	0.8231

The values in bold are the highest OA/AA/kappa achieved in each dataset.

are very dispersed. Classification accuracies with respect to the number of selected features on Pavia dataset are shown in Fig. 9. Seen from Fig. 9, CDRF achieves lower accuracies than TDRF, the reason is the same as that given on Indiana dataset. It can also be recognized that CDWOASA has a great advantage compared to CDRF and TDRF when fewer features are selected. For example, when five features are selected, the classification accuracies of CDWOASA are much better than TDRF and CDRF. As the number of selected features increases, the classification accuracies of CDRF and TDRF gradually increase. But it can be observed that CDWOASA has certain advantages in most cases. Fig. 10 depicts the ground truth and classification maps of Pavia dataset. Combining Figs. 4 and 10, we can find CDWOASA has advantages on the land-cover classes Asphalt and Soil.

The mean accuracies across different feature dimensions are summarized in Table VI. It can be seen from Table VI that CDWOASA has absolute advantages compared to CDRF and TDRF. In Indiana dataset, the mean accuracies of CDRF and

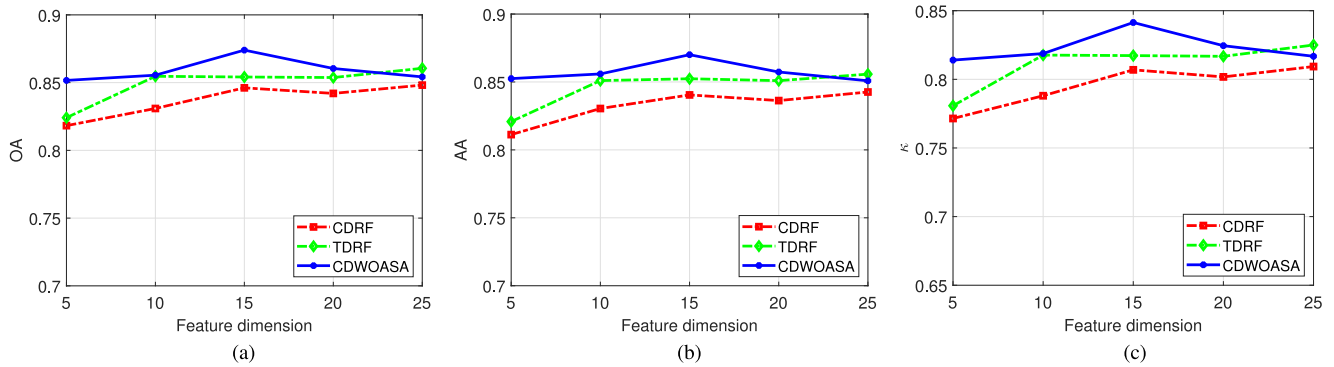


Fig. 9. Accuracies on Pavia dataset obtained by TDRF, CDRF, and CDWOASA. (a) OA. (b) AA. (c) κ .

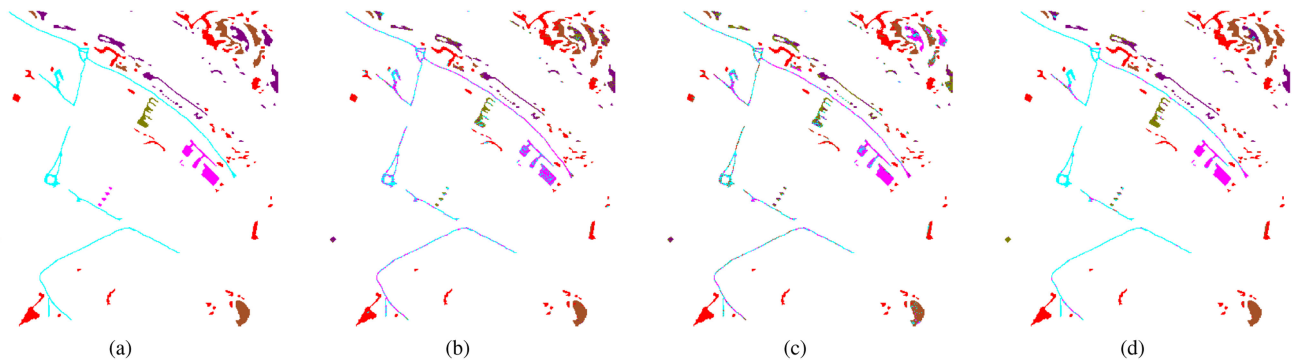


Fig. 10. Ground truth and classification maps of Pavia dataset. (a) Ground truth map. (b) The classification map over the selected features of TDRF. (c) The classification map over the selected features of CDRF. (d) The classification map over the selected features of CDWOASA.

TDRF are very similar and lower than CDWOASA. In detail, the OA of CDWOASA is 0.0569 higher than CDRF and is 0.0518 higher than TDRF. The AA and κ of CDRF and TDRF are also lower than that acquired by CDWOASA. In Pavia dataset, CDWOASA is better than CDRF and TDRF with 0.0221 and 0.0097 improvements in OA. It should be mentioned that CDWOASA has a strong exploitation ability when the number of selected features is small, which means that CDWOASA can achieve an excellent performance with small dimension of selected features. This is very meaningful in real applications.

V. CONCLUSION

This article proposes a novel cross-scene feature selection method, namely CDWOASA for HSI feature selection. In CDWOASA, the fitness function considers the classification accuracies achieved by target training samples and source training samples at the same time, ensuring the consistency of the selected features between two scenes and the separation of different land-cover classes. A sorting-based strategy is proposed to control the dimension of the selected feature subset. Moreover, the proposed method is much more competitive than two compared approaches, in particular when the dimension of selected features is relatively small. Experimental results on

two cross-scene datasets verify the superiority of the proposed method.

REFERENCES

- [1] A. Martinezuso, F. Pla, J. M. Sotoca, and P. Garciasveilla, "Clustering-based hyperspectral band selection using information measures," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 12, pp. 4158–4171, Dec. 2007.
- [2] J. Wang, J. Zhou, and W. Huang, "Attend in bands: Hyperspectral band weighting and selection for image classification," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 12, no. 12, pp. 4712–4727, Dec. 2019.
- [3] Q. Wang, Q. Li, and X. Li, "Hyperspectral band selection via adaptive subspace partition strategy," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 12, no. 12, pp. 4940–4950, Dec. 2019.
- [4] M. Pal and G. M. Foody, "Feature selection for classification of hyperspectral data by SVM," *IEEE Trans. Geosci. Remote Sens.*, vol. 48, no. 5, pp. 2297–2307, May 2010.
- [5] N. Renard, S. Bourennane, and J. Blanctalon, "Denoising and dimensionality reduction using multilinear tools for hyperspectral images," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 2, pp. 138–142, Apr. 2008.
- [6] B. Xue, M. Zhang, W. N. Browne, and X. Yao, "A survey on evolutionary computation approaches to feature selection," *IEEE Trans. Evol. Comput.*, vol. 20, no. 4, pp. 606–626, Aug. 2016.
- [7] C. Lee and G. G. Lee, "Information gain and divergence-based feature selection for machine learning-based text categorization," *Inf. Process. Manage.*, vol. 42, no. 1, pp. 155–165, 2006.
- [8] S. Gao, G. V. Steeg, and A. Galstyan, "Variational information maximization for feature selection," in *Proc. Neural Inf. Process. Syst.*, 2016, pp. 487–495.
- [9] M. Robniksikonja and I. Kononenko, "Theoretical and empirical analysis of ReliefF and RReliefF," *Mach. Learn.*, vol. 53, no. 1, pp. 23–69, 2003.

- [10] J. Huang, Y. Cai, and X. Xu, "A hybrid genetic algorithm for feature selection wrapper based on mutual information," *Pattern Recognit. Lett.*, vol. 28, no. 13, pp. 1825–1844, 2007.
- [11] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, 2003.
- [12] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, no. 95, pp. 51–67, 2016.
- [13] M. Z. Baig, N. Aslam, H. P. H. Shum, and L. Zhang, "Differential evolution algorithm as a tool for optimal feature subset selection in motor imagery EEG," *Expert Syst. Appl.*, vol. 90, pp. 184–195, 2017.
- [14] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimization for feature selection in classification: A multi-objective approach," *IEEE Trans. Syst. Man Cybern.*, vol. 43, no. 6, pp. 1656–1671, Dec. 2013.
- [15] S. Lin, K. Ying, S. Chen, and Z. Lee, "Particle swarm optimization for parameter determination and feature selection of support vector machines," *Expert Syst. Appl.*, vol. 35, no. 4, pp. 1817–1824, 2008.
- [16] S. A. Medjahed, T. A. Saadi, A. Benyettou, and M. Ouali, "Gray wolf optimizer for hyperspectral band selection," *Appl. Soft Comput.*, vol. 40, pp. 178–186, 2016.
- [17] M. Mafarja and S. Mirjalili, "Hybrid binary ant lion optimizer with rough set and approximate entropy reduces for feature selection," *Soft Comput.*, vol. 23, no. 15, pp. 6249–6265, 2019.
- [18] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.
- [19] M. Mirjalili and S. Mirjalili, "Hybrid whale optimization algorithm with simulated annealing for feature selection," *Neurocomputing*, vol. 260, pp. 302–312, 2017.
- [20] M. Abdelbasset, G. Manogaran, D. Elshahat, and S. Mirjalili, "A hybrid whale optimization algorithm based on local search strategy for the permutation flow shop scheduling problem," *Future Gener. Comput. Syst.*, vol. 85, pp. 129–145, 2018.
- [21] R. Archibald and G. Fann, "Feature selection and classification of hyperspectral images with support vector machines," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 4, pp. 674–677, Oct. 2007.
- [22] Y. Yuan, G. Zhu, and Q. Wang, "Hyperspectral band selection by multitask sparsity pursuit," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 2, pp. 631–644, Feb. 2015.
- [23] P. Ghamisi and J. A. Benediktsson, "Feature selection based on hybridization of genetic algorithm and particle swarm optimization," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 2, pp. 309–313, Feb. 2015.
- [24] G. Taskin, H. Kaya, and L. Bruzzone, "Feature selection based on high dimensional model representation for hyperspectral images," *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2918–2928, Jun. 2017.
- [25] Y. Wan, A. Ma, Y. Zhong, X. Hu, and L. Zhang, "Multiobjective hyperspectral feature selection based on discrete sine cosine algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3601–3618, May 2020.
- [26] S. Jia, Z. Ji, Y. Qian, and L. Shen, "Unsupervised band selection for hyperspectral imagery classification without manual band removal," *IEEE J. Sel. Top. Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 531–543, Apr. 2012.
- [27] W. Sun, G. Yang, and J. Li, "Robust multi-feature spectral clustering for hyperspectral band selection," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 3800–3803.
- [28] M. Ye, Y. Xu, H. Lu, K. Yan, and Y. Qian, "Cross-scene feature selection for hyperspectral images based on cross-domain information gain," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2018, pp. 4764–4767.
- [29] M. Ye, Y. Qian, J. Zhou, and Y. Y. Tang, "Dictionary learning-based feature-level domain adaptation for cross-scene hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 3, pp. 1544–1562, Mar. 2017.
- [30] B. Du, L. Zhang, D. Tao, and D. Zhang, "Unsupervised transfer learning for target detection from hyperspectral images," *Neurocomputing*, vol. 120, pp. 72–82, 2013.
- [31] M. Ye, Y. Xu, C. Ji, H. Chen, H. Lu, and Y. Qian, "Feature selection for cross-scene hyperspectral image classification using cross-domain ReliefF," *Int. J. Wavelets Multiresolution Inf. Process.*, vol. 17, no. 5, 2019, Art. no. 1950039.
- [32] J. Shen, X. Cao, Y. Li, and D. Xu, "Feature adaptation and augmentation for cross-scene hyperspectral image classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 4, pp. 622–626, Apr. 2018.
- [33] Y. Xu, J. Hu, K. Wattanachote, K. Zeng, and Y. Gong, "Sketch-based shape retrieval via best view selection and a cross-domain similarity measure," *IEEE Trans. Multimedia*, vol. 22, no. 11, pp. 2950–2962, Nov. 2020.
- [34] C. Persello and L. Bruzzone, "Kernel-based domain-invariant feature selection in hyperspectral images for transfer learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 5, pp. 2615–2626, May 2016.
- [35] L. Ma, M. M. Crawford, L. Zhu, and Y. Liu, "Centroid and covariance alignment-based domain adaptation for unsupervised classification of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 4, pp. 2305–2323, Apr. 2019.
- [36] G. Sanchita and D. Anindita, *Evolutionary Algorithm Based Techniques to Handle Big Data*. Berlin, Germany: Springer International Publishing, 2016.



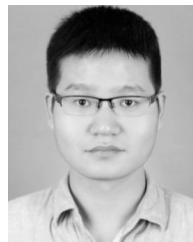
Jianxi Wang received the B.E. degree in automation from the Zhongyuan University of Technology, Zhengzhou, China, in 2019. He is currently working toward the master's degree in control engineering with China Jiliang University, Hangzhou, China.

His research interests include hyperspectral image processing and machine learning.



Minchao Ye (Member, IEEE) received the B.E. degree from Sichuan University, China, in 2010, and the Ph.D. degree from Zhejiang University, China, in 2016, both in computer science and technology.

Since 2016, he has been with the College of Information Engineering, China Jiliang University, Hangzhou, China, where he is currently an Associate Professor in Computer Science and Technology. His research interests include hyperspectral image processing, machine learning, and pattern recognition.



Fengchao Xiong (Senior Member, IEEE) received the B.E. degree in software engineering from Shandong University, Jinan, China, in 2014, and the Ph.D. degree in computer science and technology from Zhejiang University, Hangzhou, China, in 2019.

He visited Wuhan University from 2011 to 2012 and Griffith University from 2017 to 2018. He is currently a Lecturer with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, China. His research interests include hyperspectral image processing, machine

learning, and pattern recognition.

Dr. Xiong has received the Best Student Paper Award in DICTA 2020.



Yuntao Qian (Member, IEEE) received the B.E. and M.E. degrees in automatic control from Xi'an Jiaotong University, Xi'an, China, in 1989 and 1992, respectively, and the Ph.D. degree in signal processing from Xidian University, Xi'an, China, in 1996.

From 1996 to 1998, he was a Postdoctoral Fellow with the Northwestern Polytechnical University, Xi'an, China. Since 1998, he has been with the College of Computer Science, Zhejiang University, Hangzhou, China, where he became a Professor in 2002. From 1999 to 2001, and in 2006, 2010, 2013,

2015–2016, and 2018, he was a Visiting Professor with Concordia University, Hong Kong Baptist University, Carnegie Mellon University, the Canberra Research Laboratory of NICTA, Macau University, and Griffith University. His research interests include machine learning, signal and image processing, pattern recognition, and hyperspectral imaging.

Dr. Qian is an Associate Editor of the IEEE JOURNAL OF SELECTED TOPICS IN APPLIED EARTH OBSERVATIONS AND REMOTE SENSING.