Multimedia Tools and Applications (2024) 83:26901–26927
https://doi.org/10.1007/s11042-023-16638-6

Band selection using hybridization of particle swarm optimization and crow search algorithm for hyperspectral data classification

Ram Nivas Giri[1] · Rekh Ram Janghel[1] · Saroj Kumar Pandey[2]

Received: 4 May 2023 / Revised: 18 July 2023 / Accepted: 23 August 2023 /
Published online: 5 September 2023

Abstract
A Hyperspectral image (HSI) contains numerous spectral bands, providing better differentiation of ground objects. Although the data from HSI are very rich in information, their processing presents some difficulties in terms of computational effort and reduction of information redundancy. These difficulties stem mainly from the fact that the HSI consists of a large number of bands along with some redundant bands. Band selection (BS) is used to select a subset of bands to reduce processing costs and eliminate spectral redundancy. BS methods based on a metaheuristic approach have become popular in recent years. However, most BS methods based on a metaheuristic approach can get stuck in the local optimum and converge slowly due to a lack of balance between exploration and exploitation. In this paper, three BS methods are proposed for HSI data. The first method applies Crow Search Algorithm (CSA) for BS. The other two proposed methods, HPSOCSA_SP and HPSOCSA_SLP, are based on the hybridization of Particle Swarm Optimization (PSO) and CSA. The purpose of these hybridizations is to balance exploration and exploitation in a search process for optimal band selection and fast convergence. In hybridization techniques, PSO and CSA exchange informative data at each iteration. HPSOCSA_SP split the population into two equal parts. PSO is applied to one part and CSA to the other. HPSOCSA_SLP selects half of the top-performing members based on fitness. PSO and CSA are applied to the selected population sequentially. Our proposed models underwent rigorous testing on four HSI datasets and showed superiority over other metaheuristic techniques.

Keywords Hyperspectral image · Dimension reduction · Band selection · Particle swarm optimization · Crow search algorithm

✉ Ram Nivas Giri
rngiri.phd2021.it@nitrr.ac.in

Rekh Ram Janghel
rrjanghel.it@nitrr.ac.in

Saroj Kumar Pandey
sarojpandey23@gmail.com

[1] Department of Information Technology, National Institute of Technology, Raipur (C.G.) 492010, India
[2] Department of Computer Engineering and applications, GLA university Mathura, Mathura, India

1 Introduction

The state of the globe is continuously changing now than ever. Assessing these changes and their effect is imperative for protecting the planet. Hyperspectral remote sensors record comprehensive spectral responses from ground objects over hundreds of contiguous, narrow electromagnetic spectrum bands. The pixels of the obtained hyperspectral image (HSI) can be visualized as high-dimensional vectors containing values corresponding to the spectral reflectance. The high-dimensional values of each pixel help to determine minute spectral differences by which ground entities can be distinguished [1]. The HSI is successfully used in various fields, such as land use, exploration of minerals, detection of water pollution, and so on [2]. The classification or distinction of objects is the main aim of almost all HSI applications. In the HSI classification, a class category is determined for each pixel. Although HSI is particularly rich in information, the HSI classification faces several challenges. The HSI classification requires considerable computing overhead and high memory expenses due to the large number of spectral bands in the HSI data [3]. Additionally, collecting labelled samples for the ground truth image is costly, labour-intensive, and time-consuming. Therefore, the HSI dataset has a limited number of labelled samples. The high dimensionality and limited number of labelled samples of HSI data lead to the "curse of dimensionality" challenge. This challenge states that, for a fixed, limited number of training samples, the classification accuracy diminishes as the dimensionality of the data increases [4]. Moreover, the neighbouring bands in the HSI usually have a strong correlation with each other. In other words, the HSI has redundant data [5]. Data redundancy can make models unstable [6]. Because of these difficulties, dimensionality reduction (DR) becomes a necessary pre-processing step for HSI applications. The approaches of DR eliminate unnecessary bands and preserve crucial data [7].

The DR approaches are divided into two classes: feature extraction (FE) and feature selection (also referred to as band selection) (BS). The FE method maps the data from high-dimensional to low-dimensional with specific constraints. The principal component analysis (PCA) [8] and independent component analysis (ICA) [9] are examples of FE. The data produced by the FE methods lost its physical significance since they changed the original band properties [10]. The BS strategy determines a subset of the original bands that provide relevant data.

Various BS methods are available in the literature [1, 7]. The BS methods can be categorized into filter, wrapper, and embedded. The filter method determines the relevance of each band based on some criteria and select bands based on this relevance. It is independent to the training process. In the filter method, higher emphasis is placed on analyzing the merits of individual bands, and the benefits of the combination of bands are not considered [11].Wrapper methods start with a subset of bands and evaluate their importance using a prediction algorithm. This process is repeated for different subsets until the optimal subset is reached [12].In embedded feature selection, the selection process is integrated into the model learning procedure [11].

For highly correlated features (bands), a subset of bands with better classification performance can be obtained by incorporating the benefits of a combination of bands in the band selection process [11, 13]. Wrapper methods consider the advantages of combining bands. These methods involve an exhaustive or heuristic search strategy to identify a subset of bands from the original data [14]. Exhaustive search generates every possible subset of features to identify the best subset. Although this goal is desirable, search is an NP-hard combinatorial problem [15].

Metaheuristic techniques are being used to solve combinatorial problems more efficiently. These can investigate many potential band subgroups to identify a nearly ideal

feature subgroup within a reasonable timeline [16]. Exploration (or diversification) and exploitation (or intensification) are the two primary concerns in metaheuristic search algorithms. These two must be precisely balanced in the algorithm to obtain optimal results in a reasonable amount of time [17]. Exploration is the potential of the metaheuristic algorithm to search new and diverse regions of the solution space, and exploitation is the potential of the metaheuristic algorithm to use the best solutions so far and refine them locally. For HSI band selection, several nature-inspired metaheuristics have already been thoroughly investigated. These include the genetic algorithm (GA) [11], particle swarm optimisation (PSO) [18], grey wolf optimisation (GWO) [19], artificial bee colony (ABC) [20], wind-driven optimisation (WDO) [21], whale optimisation [22] and others.

The self-adaptive differential evolution method is used for BS [23]. In [18], a BS scheme is presented utilizing PSO, where the objective function has been created under the constraint of the minimal predicted abundance covariance. The PSO-based approach for the BS, which also chooses the appropriate number of bands, has been presented [24]. Another PSO-based BS method for HSI target detection was investigated in [25]. To devise a hybrid BS strategy, the authors merged a GA and PSO [26]. The GA is used for the BS, where the objective function has been formed using a weighted combination of entropy and image gradient [13]. A modified Lévy flight-based GA variant is used to avoid becoming stuck in local optima [27]. A binary form of Cuckoo Search (CS) has been employed for wrapper BS [28]. Authors of [20] have applied improved subspace decomposition and the artificial bee colony for the BS. A wind-driven optimization (WDO) model has been refined using PSO, and its use for band selection has been studied in [21]. The authors have presented a hybrid BS approach combining WDO and CS [29]. Gray Wolf Optimization (GWO), which is an algorithm inspired by the nature of grey wolves, has been used for BS [19]. Recently, a modified GWO algorithm has been presented for the BS, where chaotic operations are applied for indexing the wolves [30]. A modified discrete gravitational search algorithm creates band subsets by adhering to a requirement that increases the information conveyed by each element in a subset and reduces duplicated data between groups [31]. The capability of an ant colony algorithm for band selection has been extended using a pre-filter and a dynamic information update technique [32]. In [33], the potential of the moth-flame optimization for the BS has been investigated. The membrane whale optimization and the wavelet support vector machine ensemble technique for the BS are presented in [34].

However, these algorithms also have their limitations. The metaheuristic search algorithms for BS may get trapped in the local optimum, making it difficult to reach the global optimum. More redundant and irrelevant bands can also lead to more local optima in the solution space. These local optimum points can significantly reduce the effectiveness of metaheuristic algorithms to attain the global optimum. In addition, metaheuristic algorithms can converge slowly because of local optimum points. When exploitation and exploration are effectively balanced, the algorithm produces optimal results within a reasonable timeframe.

Thus, an investigation should be performed by combining two or more metaheuristic algorithms to balance exploration and exploitation. A hybrid metaheuristic algorithm integrates two metaheuristic strategies with the objective that the merits of both can be attained.

Crow search algorithm (CSA) and PSO are population-based meta-heuristic algorithms with different search procedures. PSO is motivated by the collective behaviour of flocks, whereas the behaviour of crow groups drives the crow search algorithm (CSA). They update the individuals in the population differently from each other to explore the search space. PSO emphasizes exploitation, whereas CSA places more focus on exploration.

Therefore, this paper proposes wrapper-based BS methods using the hybridization of PSO and CSA.

In hybridization methods, PSO and CSA exchange valuable information at each iteration to incorporate the strengths of both in the search process. These methods aim to speed up convergence and select the best band set for classification performance. The main contributions of this paper are as follows:

1) To the best of our knowledge, first time applying the CSA for band selection from the HSI data.
2) This paper proposes two hybrid models: HPSOCSA_SP and HPSOCSA_SLP, which are based on the hybridization of PSO and CSA. The goal of these hybridizations is that the competencies of PSO and CSA can be achieved in a hybrid search process
3) In HPSOCSA_SP, split the population into two equal parts. PSO is applied to one part and CSA to another.
4) In HPSOCSA_SLP, selects half of the top-performing members based on fitness. PSO and CSA are applied to the selected population sequentially.
5) This paper thoroughly examines the capability of proposed models on four benchmark HSI datasets in terms of class accuracy, average accuracy, overall accuracy, kappa score, precision, recall and F1 score.
6) The proposed methods achieved good results when compared with contemporary metaheuristic approaches. It achieved better classification accuracy with fewer bands.

The rest of the paper is organized into three sections. Section 2 includes the description of the proposed model. Section 3 presents the experimental results and comparison. Section 4 concludes this paper.

2 Methodology

This section explains the technological background of PSO and CSA, followed by descriptions of the proposed band selection methods.

2.1 Particle swarm optimization

PSO is a population-based metaheuristic optimization technique [35]. The collective behaviour of bird flocks forms the basis for the operating principle of the PSO [36]. In PSO, a group of particles (prospective solutions) forms a population. In pursuit of the optimal answer, particles move in the search area at a particular velocity. The best-known positions of each particle and the best-known of the group are used to determine the new locations of the particles [37].

Consider that n bands need to be selected, and the position of particle keeps the indices of the selected bands. The position vector (size n) of particle i at iteration itr is represented by $x_{pso}^{i,itr}$. The velocity of the i^{th} particle in iteration itr is determined by the following expression:

$$v^{i,itr+1} = wv^{i,itr} + c_1 r_1 \left(x_{pB}^{i} - x_{pso}^{i,\ itr} \right) + c_2 r_2 \left(x_{gBest} - x_{pso}^{i,\ itr} \right) \tag{1}$$

where x_{gBest} is the current optimum position, x_{pB}^{i} is the personal best position of i^{th} particle and w is the inertia weight. The participation of the local and global best is governed by cognitive factor ($c1$) and social factor ($c2$), respectively, and r_1 and r_2 are random numbers between 0 and 1 [25, 38].

Based on the changed velocity, the position of the i^{th} particle is determined by the following:

$$x_{pso}^{i,itr+1} = x_{pso}^{i,\ itr} + v^{i,\ itr+1} \tag{2}$$

The i^{th} particle updates its personal best using Eq. (3)

$$x_{pB}^{i,itr+1} = \begin{cases} x^{i,itr+1} & if\ Fitness\left(x^{i,itr+1}\right)\ is\ better\ than\ Fitness\left(x_{pB}^{i}\right) \\ No\ Change & Otherwise \end{cases} \tag{3}$$

2.2 Crow search algorithm

CSA is a population-based metaheuristic optimization method [39]. The fundamental principle of CSA is based on the behaviour of crow flocks. Crows live in groups. They explore food spots and memorize the best ones they find. When searching for new food sites, crows frequently follow another crow to see its food site and steal from it. In addition, when a crow detects that another crow is following it, it will flee to a random location rather than its food source [40].

Consider a d-dimensional search space and N number of crows. At iteration itr, the position of each crow i is denoted by a vector$x^{i,itr} = \left[x_1^{i,itr}, x_2^{i,itr}, \ldots x_d^{i,itr}\right]$. Each crow represents a potential solution for the concern. A variable X depicts the location of all crows as shown in Eq. (4).

$$X = \begin{bmatrix} x_1^1 & x_2^1 & \cdots & x_d^1 \\ x_1^2 & x_2^2 & \ldots & x_d^2 \\ \cdots & \cdots & \ldots & \cdots \\ x_1^N & x_2^N & \cdots & x_d^N \end{bmatrix} \tag{4}$$

The fitness of each crow's position is estimated using an objective function. Each crow has a memory for the position that is currently the fittest. At iteration itr, the memory of each crow i is denoted by a vector $M^{i,itr} = \left[M_1^{i,itr}, M_2^{i,itr}, \ldots M_d^{i,itr}\right]$. A matrix, MEM (which kept the memory of all crows), was represented as:

$$MEM = \begin{bmatrix} M_1^1 & M_2^1 & \cdots & M_d^1 \\ M_1^2 & M_2^2 & \cdots & M_d^2 \\ \cdots & \cdots & \cdots & \cdots \\ M_1^N & M_2^N & \cdots & M_d^N \end{bmatrix} \tag{5}$$

In other words, the MEM variable keeps the location of the foods that each crow has hidden.

The i^{th} crow generates a new location at any iteration itr by following to the j^{th} crow (selected randomly). Two states are probable in this situation:

State 1: **j^{th} Crow is unaware that i^{th} Crow is following it.**

$$x^{i,\ itr+1} = x^{i,\ itr} + r_i \times FL^{i,\ itr} \times \left(M^{j,\ itr} - x^{i,\ itr}\right) \tag{6}$$

where flight length (length covered by crow in a single ride) is represented using FL, and variable r_i is a random value between 0 and 1.

State 2: **j^{th} Crow is aware that i^{th} Crow is following it.**

$$x^{i,itr+1} = Any\ Random\ Position \tag{7}$$

In CSA, a variable called awareness probability (AP) is used, which measures the likelihood that a crow will be conscious that it is being pursued. On the basis of AP Eqs. (6) and (7) are merged as followed:

$$x^{i,itr+1} = \begin{cases} x^{i,itr} + r_i \times FL^{i,itr} \times \left(M^{j,itr} - x^{i,itr}\right) & when\ r_j \geq AP^{j,itr} \\ Any\ Random\ Position & Otherwise \end{cases} \tag{8}$$

The i^{th} crow updates its memory using Eq. (9)

$$M^{i,itr+1} = \begin{cases} x^{i,itr+1} & if\ Fitness\left(x^{i,itr+1}\right)\ is\ better\ than\ Fitness\left(M^{i,itr}\right) \\ No\ Change & Otherwise \end{cases} \tag{9}$$

where *Fitness* () represents fitness value.

2.3 Proposed methodology

Assume that the HSI data cube is represented using $H^{D1,\ D2,\ D3}$, in which *D1* and *D2* represent the spatial dimensions and *D3* the spectral. In other words, *H* has a *D3* number of bands, and each band has a $D1 \times D2$ number of pixels. *L* represents the count of labelled samples obtained from the ground truth of *H*. The band selection algorithm aims to find a subset of the bands *n* (from [*1, D3*]) that provides high classification accuracy.

This paper proposes three methods for band selection:

1) CSA-based BS.

2) Hybridization of PSO and CSA by split population (HPSOCSA_SP) for BS,

3) Hybridization of PSO and CSA by select population (HPSOCSA_SLP) for BS.

Figure 1 shows the workflow of the process, which includes preprocessing, BS based on the proposed method, classifiers based on selected bands, and performance evaluation.

2.3.1 Preprocessing

This part consists of two steps: mapping and initialization. An input 3-dimnesional HSI data cube is converted into a 2D matrix in mapping. Data cube *H* has been transformed into a 2-dimensional matrix (called a "Feature") of size $L \times D3$ .The initialization step involves initializing the parameters of different aspects of the applied metaheuristic algorithm, such as search space boundary, population size (*N*), and maximum iteration (itr_{Max}).

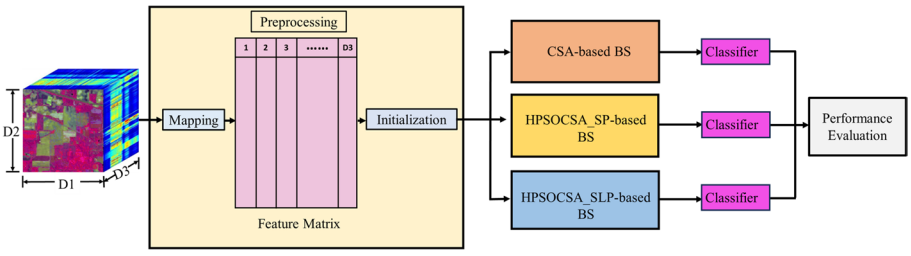

Fig. 1 Workflow of the proposed process

Objective function In this paper, the classification error of the classifier is used as the objective function. In this case, the objective function is used as a minimization function. Based on the objective function fitness of each member of the population is calculated. The classification error (*CE*) is computed using on the basis of overall accuracy (*OA*) Eqs. (10)-(11).

$$OA = \frac{\sum_{i=1}^{r} t_i}{T} \tag{10}$$

$$CE = 1 - OA \tag{11}$$

where r represents the no of classes in the HSI data, t_i is the no of samples of the i^{th} class that are correctly classified ($i = 1, 2\ldots r$), and T is the total no of labeled samples.

2.3.2 CSA-based BS

Suppose X (size $N \times n$) and *Fit* (size N) store the position and corresponding fitness value of each crow of the population, respectively. In addition, the best-known position of each population member and the fitness of that position are maintained in *MEM* (size $N \times n$) and *FitM* (size N), respectively. The proposed CSA-based BS starts with random population initialization in such a way that the size of the position matrix must be N x n, where N is the number of crows (population size) and n is the number of bands to be selected. The position of the crow keeps the indices of the selected bands. Each crow represents a population member, and their corresponding memory positions denote a possible solution (a subset of selected bands). Algorithm 1 includes the pseudo-code of the CSA-based BS.

Algorithm 1 Crow Search Algorithm based HSI band selection

Input: Feature matrix (size *L* × *D3),* Label vector (size *L),* and value of *n.*
Output: a subset of *n* bands from *[1, D3]*

Steps:

1. **[Initialization of variables]**
   LB=1; UB=D3 ; // Search space boundary
   AP (Awareness Probability); *FL* (Flight length);
   N (Population Size); itr_{Max} (Maximum iteration); *itr = 0;*
2. **[Initialize position matrix *X* of size *N* × *n*]**
   For *i =1 to N*
   value = Randperm*(D3)*; // generate *D3* unique random number between *[1, D3]*
   X(i, :) = value *(1 : n)*; // assign first n values to *X(i)*
   End
3. Compute the fitness of the position of each crow and assign the result to *Fit* (size *N)*
4. **[Initialize memory matrix *MEM* of size *N* × *n*]**
5. *MEM=X*;
6. **[Initialize fitness of memory FitM]**
   FitM=Fit;
7. Find the optimal position of memory based on the fitness value.
8. **While** (*itr* < itr_{Max})
   // for each crow of population
   For *i =1 to N*
   Pick a crow (say, *j*) at random to pursue.
   If *(r_j > AP)*
   Modify position $x^{i,itr+1}$ according to Eq. (8)
   Else
   $x^{i,itr+1} = Any\_Random\_Position$ // in the search space
   End If
   End For
   Examine the feasibility of new location
   Compute fitness of the new position of each crow and update vector Fit
   Update memory matrix *MEM* according to Eq. (9)
8. **End While**
9. Assign the best position of the memory based on the fitness value to the n .

2.3.3 Hybrid approaches involving PSO and CSA for BS

Metaheuristic techniques begin by exploring local optima and then move towards global optima to solve optimization problems [41]. For that, balancing exploration and exploitation to reach global optima during iteration is crucial. From Eqs. (1)-(2), it can be observed that in PSO, the movement of particles depends on their best position and the group's best position. This aspect assists in exploiting existing areas. It has a limited ability to explore unknown areas in search space. Equation (8) shows that in CSA, when a crow becomes aware of others following it, it moves to a random position in the search space. As a result, CSA is more effective in exploring new areas.

Two hybrid approaches (HPSOCSA_SP and HPSOCSA_SLP) based on PSO and CSA are proposed for band selection. In the proposed hybrid methods, the exploitation capability of PSO and the exploration capability of CSA are involved in the search process. PSO and CSA exchange information via population sharing to find an optimal solution at a reasonable time.

HPSOCSA_SP randomly split the population into two equal parts. In this approach, half of the members are processed by PSO and the rest by CSA in each iteration. The population members processed by PSO may or may not be processed by CSA in the next iteration. Similarly, population members processed by CSA may or may not be processed by PSO in the next iteration. HPSOCSA_SLP selects half the top-performing members based on fitness and sequentially applies PSO and CSA in each iteration.

Thus, HPSOCSA_SP is more random than HPSOCSA_SLP regarding the information sharing between PSO and CSA through population members.

HPSOCSA_SP The proposed HPSOCSA_SP randomly generates the initial population. The entire population is then split into two equal parts. Participants in each part are randomly selected. PSO is applied to one portion, and CSA is applied to another. Figure 2 shows the workflow of HPSOCSA_SP, which includes population splitting, PSO process, CSA process, and population integration.

The population is initialized randomly during preprocessing. Then, population splitting, PSO, CSA, and population integration form a new population for the next generation.

Population splitting In this step, the entire population is divided into two equal parts of size N/2 (N1) (Part A and Part B). The PSO and CSA progress on Part A and Part B, respectively. Suppose *idx, idx1,* and *idx2* are three vectors where *idx* = {randomly generate N integer number in [*1, N*]}; *idx1* = {first half of *idx*}*;* and *idx2* = {second half of *idx2}.*In the proposed method, the memory of the crow is treated as the same as the particle's best-known position because both perform the same role.

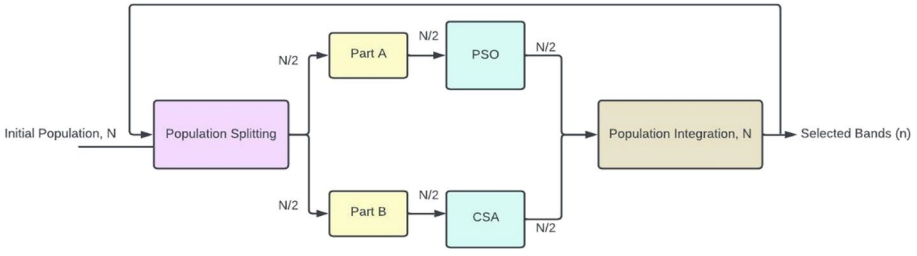

Fig. 2 Work flow of the HPSOCSA_SP

Part a The position matrix (x_{pso}) for Part A is derived from X by using the following:

$$x_{pso} = X[idx1]_{N1\times n} \tag{12}$$

The velocity matrix, (v_{pso}) for Part A is determined from v using the following:

$$v_{pso} = v[idx1]_{N1\times n} \tag{13}$$

The fitness of position matrix (Fit_{pso}) for Part A is determined from Fit using the following:

$$Fit_{pso} = Fit[idx1]_{N1} \tag{14}$$

The personal best position matrix (x_{pBest}) for Part A is determined from X_{pB} using the following:

$$x_{pBest} = X_{pB}[idx1]_{N1\times n} \tag{15}$$

The fitness of personal best position ($FitP_{pso}$) for Part A is determined from $FitP$ using the following:

$$FitP_{pso} = FitP[idx1]_{N1} \tag{16}$$

Part B The position matrix (x_{csa}) for Part B is determined from X using the following:

$$x_{csa} = X[idx2]_{N1\times n} \tag{17}$$

The velocity matrix (v_{csa}) for Part B is determined from v using the following:

$$v_{csa} = v[idx2]_{N1\times n} \tag{18}$$

The fitness of position matrix (Fit_{csa}) for Part B is determined from Fit using the following:

$$Fit_{csa} = Fit[idx2]_{N1} \tag{19}$$

The memory matrix (M_{csa}) for Part B is determined from X_{pB}using the following:

$$M_{csa} = X_{pB}[idx2]_{N1\times n} \tag{20}$$

The fitness of memory matrix ($FitM_{csa}$) for Part B is determined from $FitP$ using the following:

$$FitM_{csa} = FitP[idx2]_{N1} \tag{21}$$

PSO process The PSO process updates the population Part A. This process updates the values of x_{pso}, v_{pso}, Fit_{pso}, x_{pBest}, and $FitP_{pso}$ based on PSO.

CSA process The CSA process updates the population Part B. This process updates the values of x_{csa}, Fit_{csa}, M_{csa} and $FitM_{csa}$. The values of v_{csa} stay the same after this process.

Population integration In this step, the updated Part A and Part B are combined into a single population size N. The value of X, v, Fit, X_{pB}, and $FitP$ are updated using the following:

$$X = \begin{bmatrix} x_{pso} \\ x_{csa} \end{bmatrix}_{N\times n} \tag{22}$$

$$v = \begin{bmatrix} v_{pso} \\ v_{csa} \end{bmatrix}_{N \times n} \tag{23}$$

$$Fit = \begin{bmatrix} Fit_{pso} \\ Fit_{csa} \end{bmatrix}_{N} \tag{24}$$

$$X_{pB} = \begin{bmatrix} x_{pBest} \\ M_{csa} \end{bmatrix}_{N \times n} \tag{25}$$

$$FitP = \begin{bmatrix} FitP_{pso} \\ FitM_{csa} \end{bmatrix}_{N} \tag{26}$$

After integration, update the global best solution (X_{Best}) based on fitness value of X_{pB}.

In brief, HPSOCSA_SP involves preprocessing, followed by a sequential repetition of population-splitting, PSO, CSA, and population integration itr_{Max} times. After the whole process, the values of X_{Best} represent the indices of the selected bands. Algorithm 2 includes the pseudo-code of the HPSOCSA_SP. The flowchart of HPSOCSA_SP is presented in Fig. 3.

Algorithm 2 HPSOCSA_SP based HSI band selection

Input: Feature matrix of size $L \times D3$, Label vector of size L, and number of band (n)
Output: subset of n bands from $[1, D3]$

1. **[Initialization of variables]**
   $LB=1$; $UB=D3$; // Search space boundary
   N (Population Size); itr_{Max} (Maximum iteration); $N1=N/2$ (half population size); $itr=0$.
   PSO parameters: c_1, c_2 and w
   CSA parameters: AP (Awareness Probability); FL (Flight length);
2. **[Initialize position matrix X of size $N \times n$]**
   For $i = 1$ to N
   $value$ = Randperm$(D3)$; // generate $D3$ unique random number between $[1, D3]$
   $X(i, :)$ = value $(1 : n)$; // assign first n values to $X(i)$
   End
3. **[Initialize velocity matrix v of size $N \times n$]**
   $v = X$;
4. Compute the fitness of the position and assign the result to Fit (a vector of size N)
5. **[Initialize Personal best fitness $FitP$ and personal best position X_{pB} for each Particle]**
   $FitP = Fit$; //vector of size N
   $X_{pB} = X$; // matrix of size $n \times N$
6. Based on fitness value, determine initial solution X_{Best}
7. **While** ($itr < itr_{Max}$)
   [The entire population is split into two equal parts].
   Determine x_{pso}, v_{pso}, Fit_{pso}, x_{pBest} and $FitP_{pso}$ based on Eq. (12)-(16)
   Find global best x_{gBest} on the basis of optimal value of $FitP_{pso}$ from x_{pBest}
   For i 1 **to** $N1$ // for PSO
   Modify $v_{pso}^{i,itr}$ and $x_{pso}^{i,itr}$ using Eq. (1) and (2), respectively.
   Perform boundary correction
   Compute the fitness of the updated position and update Fit_{pso}
   Update x_{pBest} and x_{gBest}.
   END
   Determine x_{csa}, v_{csa}, Fit_{csa}, M_{csa} and $FitM_{csa}$ based on Eq. (17)-(21)
   For i 1 **to** $N1$ // for CSA
   Pick a crow (say, j) at random to pursue // value of j lie between [$1, N1$]
   Modify $x_{csa}^{i,itr}$ using Eq. (8)
   Examine the feasibility of new location
   Compute fitness of the new position of the crow and update vector Fit_{csa}
   Update memory matrix M_{csa} according to (9)
   END
   // Integration of two solutions
   Form X , v, Fit, X_{pB}, $FitP$ using Eqs. (22)-(26)
   Find the global best solution X_{Best}, based on fitness value of X_{pB}
8. **End While**
9. **Display:** values of X_{Best} (indices of the selected of bands).

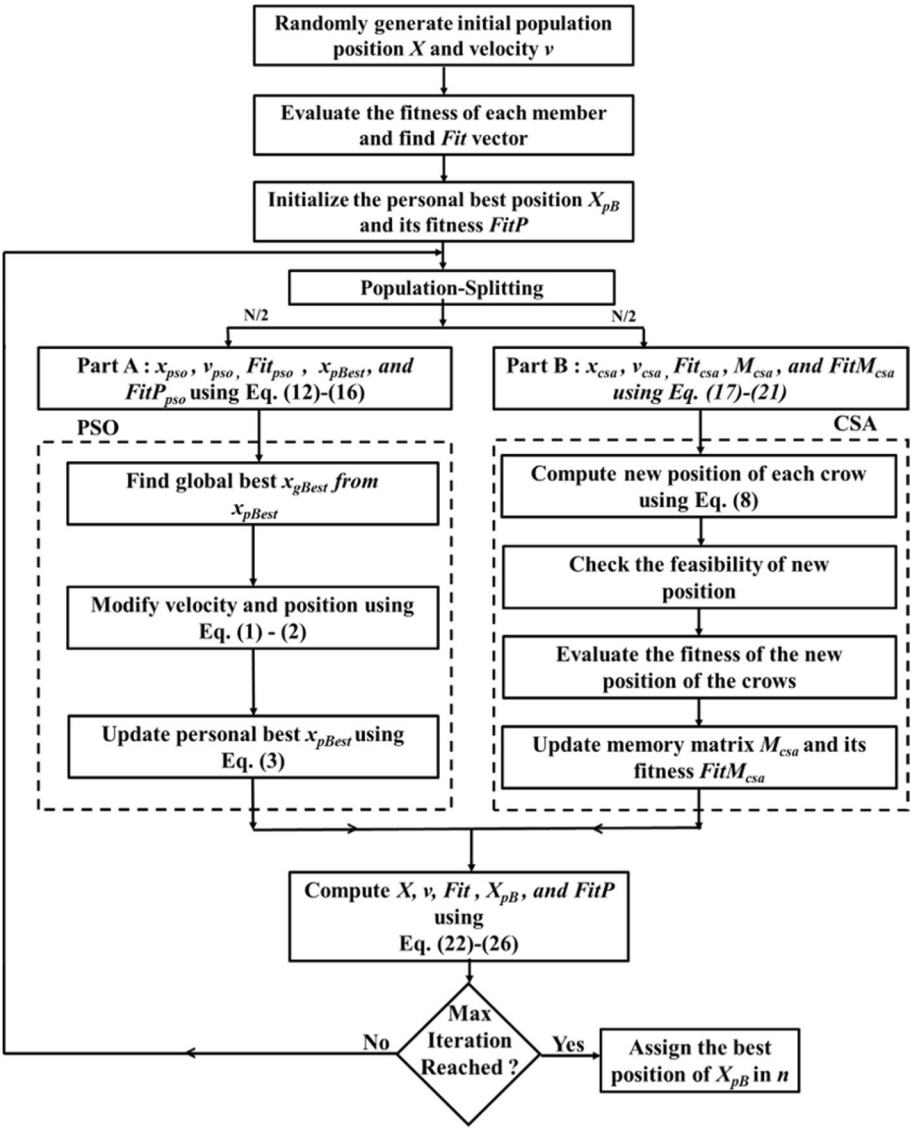

Fig. 3 Flowchart of HPSOCSA_SP process

HPSOCSA_SLP Figure 4 illustrates the workflow of HPSOCSA_SLP. The process of HPSOCSA_SLP consists of four steps: selection, PSO process, CSA process, and population-integration.

The population is randomly initialized during the preprocessing stage. A new population for the upcoming generation is created using selection, PSO, CSA, and population integration. The details of these steps are given below.

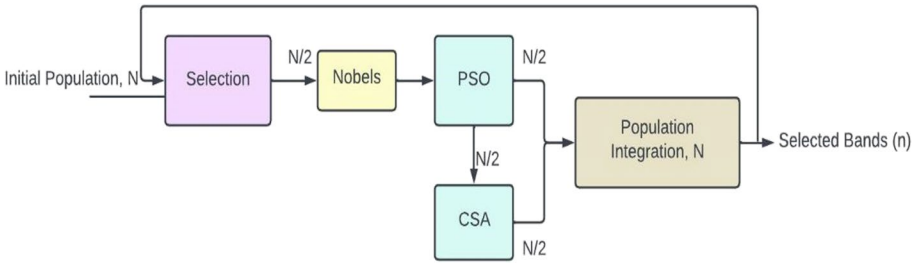

Fig. 4 Work flow of the HPSOCSA_SLP

Selection This step selects *N1* of the top-performing members. For that, based on the fitness value of every member of the population, the members are ordered (sorted) so that the first member has the best fitness value, and the last member has the worst. The first half of the population is picked. These selected members of the population are viewed as nobles.

The fitness of personal best vector, *FitP* ordered using the following:

$$\left[FitP_{order}, idx3\right] = sort[FitP] \tag{27}$$

where $FitP_{order}$ includes the sorted fitness value of personal-known, while $idx3$ has the corresponding indices of *FitP* vector. As in this proposed model, the first half of the sorted population is to be selected for the PSO operation, so another vector called *idx4* is maintained, which keeps the first *N1* values of *idx3*.

Using *idx4*, position matrix *X*, fitness vector *Fit*, velocity matrix *v,* the personal best-known matrix X_{pB}, and *FitP* are arranged as follows:

Nobles

$$x_{pso} = X[idx4]_{N1\times n} \tag{28}$$

$$Fit_{pso} = Fit[idx4]_{N1} \tag{29}$$

$$v_{pso} = v[idx4]_{N1\times n} \tag{30}$$

$$x_{pBest} = X_{pB}[idx4]_{N1\times n} \tag{31}$$

$$FitP_{pso} = FitP[idx4]_{N1} \tag{32}$$

PSO process In this step, PSO updates particles of the selected population, creating the PSO-generated population. The updated values of X_{pso}, Fit_{pso}, v_{pso}, X_{pB_pso}, and $FitP_{pso}$ are also stored in x_{csa}, Fit_{csa}, v_{csa}, M_{csa}, and $FitM_{csa}$, respectively, as this population becomes the input for the CSA process.

CSA process In this step, the members of PSO-generated population are updated by CSA. In other words, the values of x_{csa}, Fit_{csa}, M_{csa}, and $FitM_{csa}$ are updated. After this process, the values of v_{csa} remain unchanged. The resulting population is termed the CSA-generated population.

Population integration This step involves integrating the PSO-generated population and the CSA-generated population into a single population (of size *N*). The position matrix *X*,

velocity matrix v, fitness vector Fit, personal-best X_{pB}, and fitness of personal-best $FitP$ updated using Eqs. (22)-(26).

In brief, HPSOCSA_SLP involves preprocessing, followed by a sequential repetition of selection, PSO, CSA, and population integration itr_{Max} times. After the whole process, the values of X_{Best} represent the indices of the selected bands. Algorithm 3 includes the pseudo-code of the HPSOCSA_SLP. The flowchart of HPSOCSA_SLP is presented in Fig. 5.

3 Result and discussion

3.1 Dataset

The effectiveness of the proposed band selection methods was evaluated using four benchmark HSI datasets Indian Pines (IP), Kennedy space centre (KSC), Botswana (BW), and Pavia University. Table 1 lists the details of the mentioned HSI datasets [42].

Input: Feature matrix of size $L \times D3$, Label vector of size L, and number of band (n)
Output: subset of n bands from $[1, D3]$

[Initialization of variables]
  $LB=1$; $UB=D3$; // Search space boundary
  N (Population Size); itr_{Max} (Maximum iteration); $N1=N/2$ (half population size)
  PSO parameters: c_1, c_2 and w
  CSA parameters: AP (Awareness Probability); FL (Flight length);
2. [Initialize position matrix X of size $N \times n$]
  For $i = 1$ *to* N
    $value$ = Randperm$(D3)$; // generate $D3$ unique random number between $[1, D3]$
    $X(i, :)$ = value $(1 : n)$; // assign first n values to $X(i)$
  End
3. [Initialize velocity matrix v of size $N \times n$]
  $v = X$;
4. Compute the fitness of the position and assign the result to Fit (a vector of size N)
5. [Initialize Personal best fitness $FitP$ and personal best position X_{pB} for each Particle]
  $FitP = Fit$; //vector of size N
  $X_{pB} = X$; // matrix of size $n \times N$
6. Based on fitness value, determine initial solution X_{Best}
7. While ($itr < itr_{Max}$)
  Select N1 of the top-performing members based on fitness value
  Determine x_{pso}, Fit_{pso}, v_{pso}, x_{pBest} and $FitP_{pso}$ based on Eqs. (27)-(32)
  Find global best x_{gBest} on the basis of optimal value of $FitP_{pso}$ from x_{pBest}
  For i 1 **to** $N1$ // for PSO
    Modify $v_{pso}^{i,itr}$ and $x_{pso}^{i,itr}$ using Eq. (1) and (2), respectively.
    Perform boundary correction
    Compute the fitness of the updated position and update Fit_{pso}
    Update x_{pBest} and x_{gBest}.
  END
  Assign the values of X_{pso}, Fit_{pso}, v_{pso}, $X_{pB\ pso}$, and $FitP_{pso}$ to X_{csa}, Fit_{csa}, v_{csa}, M_{csa}, and $FitM_{csa}$, respectively.
  For i 1 **to** $N1$ // for CSA
    Pick a crow (say, j) at random to pursue // value of j lie between [$1, N1$]
    Modify $x_{csa}^{i,itr}$ using Eq. (8)
    Examine the feasibility of new location
    Compute fitness of the new position of the crow and update vector Fit_{csa}
    Update memory matrix M_{csa} and $FitM_{csa}$ according to (9)
  END
  // Integration of two solutions
  Form X , v, Fit, X_{pB}, $FitP$ using Eq. (22)-(26)
  Update global best solution X_{Best} based on optimal value of $FitP$ from X_{pB}
8. End While
9. Display: values of X_{Best} (indices of the selected of bands) .

Algorithm 3 HPSOCSA_SLP based HSI band selection

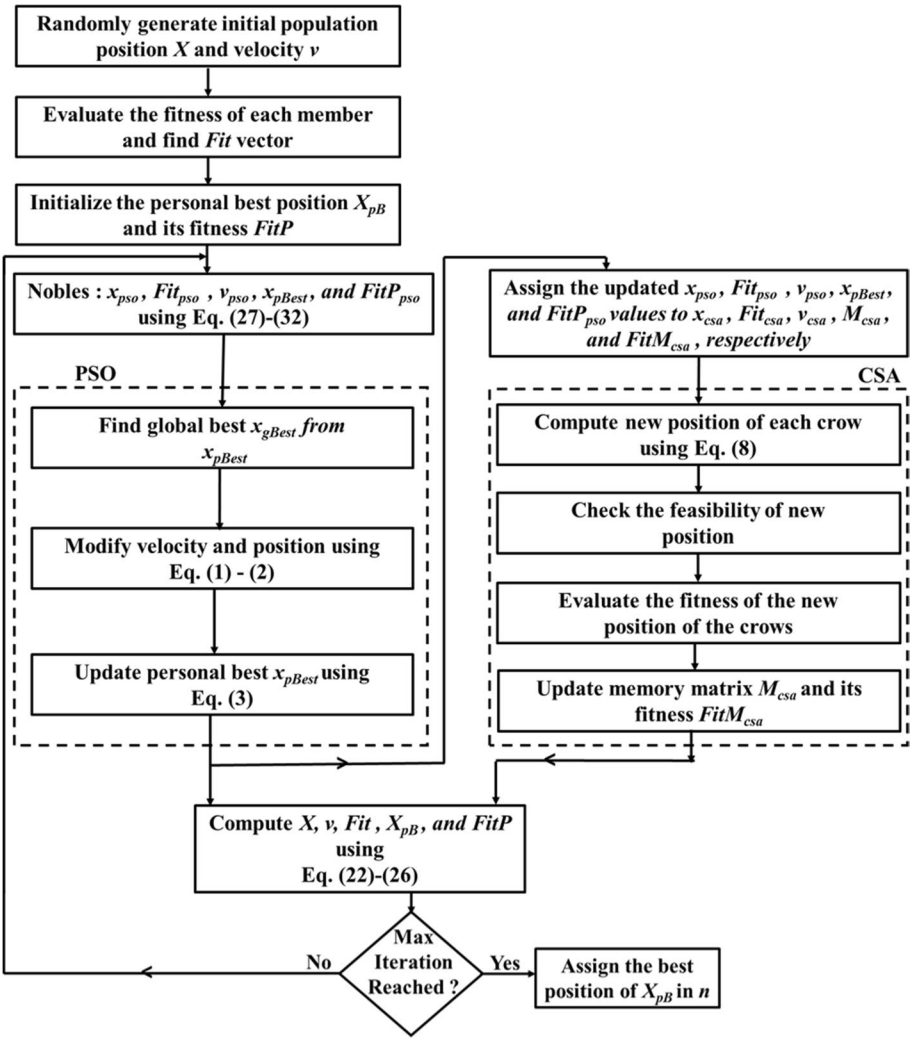

Fig. 5 Flowchart of HPSOCSA_SLP process

Table 1 Summarized details HSI datasets

HSI Data	Indian Pine	KSC	Botswana	Pavia University
Number of spectral band (After correction)	200	176	145	103
Size of band (in pixel)	145×145	512×614	1476×256	610×340
Wavelength (in nm)	400-2500	400-2500	400-2500	430-860
Spatial resolution (in meter)	20	18	30	1.3
Number of classes	16	13	14	9
Available labeled sample (*L*)	10,249	5211	3248	42,776
Sensor Name	AVIRIS	AVIRIS	Hyperion	ROSIS

Table 2 Parameters values of GA, PSO and CSA

Algorithm	Parameter	Value
GA	Population Size	20
	Crossover rate	0.9
	Mutation rate	0.01
	Maximum number of iteration	200
PSO	Population Size	20
	c_1	2
	c_2	2
	w	0.2
	Maximum number of iteration	200
CSA	Population Size	20
	AP	0.1
	FL	1.5
	Maximum number of iteration	200

3.2 Performance evaluation measure

The usefulness of the proposed methods has been evaluated using the following quantitative metrics: class accuracy (CA), average accuracy (AA), OA, Kappa coefficient, precision, recall, and F1 score[43–45].

$$C_i = \frac{t_i}{T_i} \tag{33}$$

$$AA = \frac{\sum_{i=1}^{r} C_i}{r} \tag{34}$$

$$Kappa = \frac{p_0 - p_c}{1 - p_c} \tag{35}$$

$$p_c = \frac{1}{T^2} \bullet \sum_{i=1}^{r} s_t \bullet s_c \tag{36}$$

$$Precision\ (P) = \frac{TP}{TP + FP} \tag{37}$$

$$Recall\ (R) = \frac{TP}{TP + FN} \tag{38}$$

$$F1\ score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{39}$$

where C_i is class accuracy of i^{th} class, t_i is the count of samples of i^{th} class that classified correctly, T_i is total count of samples in the i^{th} class, T is total count of labeled samples in the image, r is total count of classes, p_o represent observed agreement, p_c represent chance

agreement, s_t is true count of samples of i^{th} class, and s_c is count of samples classified as into the ith class [46]. *TP*, *FP*, *a*nd *FN* represent true positive, false positive and false negative, respectively.

3.3 Experimental setup

The experiments are executed on the Intel Xeon 2.2 GHz CPU having 64 GB RAM using MATLAB. For each dataset, the labeled pixels (samples) are divided into two parts: training data and a testing data. The training data consists of 10% of samples from each land-cover class chosen at random, with the rest making up the testing data. Then, the HSI with selected bands is classified using a support vector machine

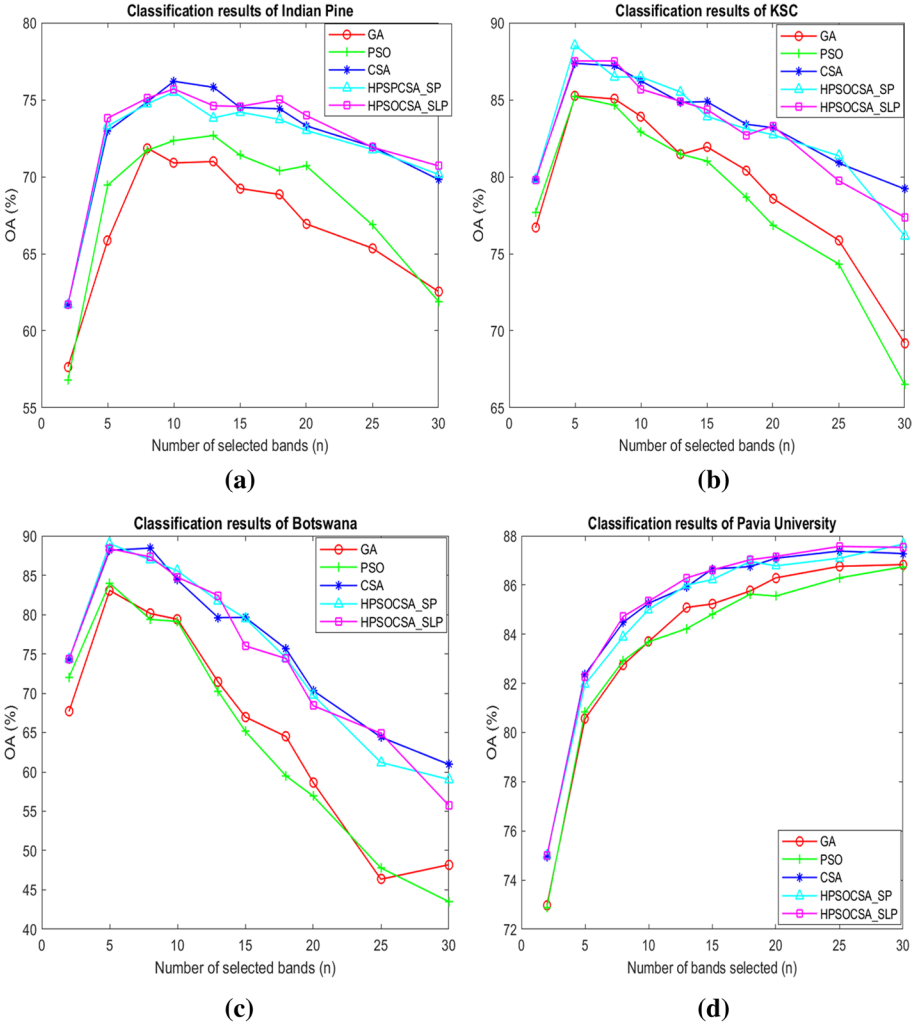

Fig. 6 Overall accuracy for a selected number of bands (**a**) Indian Pines (**b**) KSC (**c**) Botwana (**d**) Pavia University

Table 3 Classification results of Indian Pines for the ten bands

Class Label	GA	PSO	CSA	HPSOCSA_SP	HPSOCSA_SLP
Alfalfa	9.76	17.07	**24.39**	21.95	0
Corn-notill	65.91	69.65	71.05	71.67	**72.53**
Corn-mintill	43.78	42.57	51.41	**53.82**	48.13
Corn	8.92	16.43	29.58	**36.15**	21.96
Grass-pasture	74.94	60.46	75.40	80.46	**88.02**
Grass-trees	93	92.39	93.47	88.75	**95.74**
Grass-pasture-mowed	0	0	0	0	0
Hay-windrowed	86.54	94.43	94.42	92.33	**94.65**
Oats	0	0	0	0	0
Soybean-notill	60.30	55.15	63.54	67.31	**70.02**
Soybean-mintill	84.12	87.78	**88.78**	86.43	86.10
Soybean-clean	42.40	47.65	**60.98**	54.78	44.94
Wheat	86.96	87.50	82.16	76.22	**91.89**
Woods	93.59	96.14	95.52	95.61	**96.66**
Buildings-Grass-Trees-Drives	44.54	48.56	**53.89**	48.99	46.40
Stone-Steel-Towers	48.19	**49.40**	48.81	41.67	**49.40**
OA	70.90	72.35	**76.20**	75.51	75.70
AA	52.68	54.07	**58.34**	57.26	56.65
Kappa	0.6604	0.6771	**0. 7234**	0.7156	0.7180

The bold values represent the better performance of the competing methods

Table 4 Classification results of KSC for the ten bands

Class Label	GA	PSO	CSA	HPSOCSA_SP	HPSOCSA_SLP
Scrub	88.61	91.24	**95.76**	94.59	94.30
Willow swamp	**88.07**	69.72	72.94	78.90	84.86
Cabbage palm hammock	77.39	**82.61**	82.17	79.65	74.89
Cabbage palm/oak hammock	51.54	53.74	57.71	**68.28**	67.84
Slash pine	50.69	42.36	**65.28**	49.31	47.22
Oak/broadleaf hammock	46.86	47.34	**47.57**	43.69	40.29
Hardwood swamp	51.06	26.60	49.47	**60.64**	56.38
Graminoid marsh	83.72	81.40	**87.08**	79.38	78.61
Spartina marsh	81.84	77.56	86.14	**90.60**	90.38
Cattail marsh	**95.05**	90.66	85.44	91.18	91.18
Salt marsh	86.24	89.68	**91.51**	89.15	88.36
Mud flats	92.92	97.57	96.91	**98.01**	94.70
Water	98.68	98.92	99.88	**100**	**100**
OA	83.92	82.90	86.22	**86.48**	85.69
AA	76.36	73.03	78.30	**78.72**	77.62
Kappa	0.8207	0.8091	0.8462	**0.8491**	0.8403

The bold values represent the better performance of the competing methods

Table 5 Classification results of Botswana for 10 bands

Class Label	GA	PSO	CSA	HPSOCSA_SP	HPSOCSA_SLP
Water	99.59	**100**	**100**	**100**	**100**
Hippo grass	52.22	47.25	54.44	58.24	**62.64**
Floodplain grasses 1	85.33	85.33	90.22	**92.44**	90.22
Floodplain grasses 2	71.13	64.43	**80.83**	72.16	79.38
Reeds1	**82.23**	69.83	80.99	80.99	78.10
Riparian	65.29	**83.88**	77.37	76.86	74.38
Firescar	**100**	99.14	99.14	**100**	**100**
Island interior	73.08	83.52	79.23	94.51	**95.05**
Acacia woodlands	86.22	84.04	**89.40**	89.01	86.17
Acacia shrublands	80.72	70.98	**85.20**	78.57	83.04
Acacia grasslands	93.07	89.82	**96.73**	92.0	89.82
Short mopane	66.67	63.80	73.62	71.78	**75.46**
Mixed mopane	69.42	71.90	81.74	**95.04**	84.71
Expose soils	25.58	40.0	34.88	**50.59**	48.24
OA	79.40	79.13	84.40	**85.56**	84.74
AA	75.04	75.28	80.27	**82.30**	81.94
Kappa	0.7761	0.7731	0.8305	**0.8432**	0.8343

The bold values represent the better performance of the competing methods

(SVM) classifier. The CA, AA, OA and Kappa are computed using testing data. This experiment is repeated 10 times for each dataset and average performance is reported. In addition, the proposed methods are compared with those of two alternative metaheuristics based feature selection methods, GA and PSO. GA is a population-based metaheuristic algorithm [47]. The values of the parameters of the mentioned

Table 6 Classification results of Pavia for 10 bands

Class Label	GA	PSO	CSA	HPSOCSA_SP	HPSOCSA_SLP
Asphalt	84.48	85.35	**85.87**	85.70	84.46
Meadows	98.46	98.06	97.80	**98.51**	98.16
Gravel	52.67	**53.73**	52.86	53.10	52.36
Trees	84.81	88.14	85.82	**88.21**	85.10
Painted metal sheets	96.28	96.61	96.94	**97.69**	96.78
Bare Soil	51.01	51.04	**59.77**	56.06	58.18
Bitumen	13.28	19.88	26.04	25.81	**41.02**
Self-blocking bricks	87.27	82.23	**89.13**	85.18	87.75
Shadows	93.31	91.43	91.08	92.61	**93.90**
OA	83.70	83.69	85.26	85	**85.34**
AA	73.51	74.05	76.15	75.88	**77.52**
Kappa	0.7774	0.7779	0.8003	0.7962	**0.8011**

The bold values represent the better performance of the competing methods

Table 7 Precision, recall and F1 score of all mentioned dataset for 10 bands

Dataset	Method	Precision	Recall	F1-score
Indian Pines	GA	70.15	52.68	56.27
	PSO	69.96	54.07	58.05
	CSA	**73.07**	**58.34**	**62.53**
	HPSOCSA_SP	72.25	57.26	61.55
	HPSOCSA_SLP	65.85	56.65	59.39
KSC	GA	80.47	76.36	77.57
	PSO	80.97	73.03	74.55
	CSA	83.20	78.30	79.88
	HPSOCSA_SP	**84.39**	**78.72**	**80.51**
	HPSOCSA_SLP	83.90	77.62	79.37
Botswana	GA	86.38	75.04	77.27
	PSO	85.20	75.28	77.58
	CSA	88.85	80.27	82.32
	HPSOCSA_SP	88.91	**82.30**	**84.11**
	HPSOCSA_SLP	**88.95**	81.94	83.91
Pavia University	GA	84.61	73.51	75.92
	PSO	84.68	74.05	76.70
	CSA	86.17	76.15	78.73
	HPSOCSA_SP	**86.68**	75.88	78.58
	HPSOCSA_SLP	86.65	**77.52**	**80.45**

The bold values represent the better performance of the competing methods

algorithm are presented in Table 2. All parameters of PSO and CSO are included in HPSOCSA_SP and HPSOCSA_SLP.

3.4 Experimental results

In this part, the outputs of the proposed approach are presented. Figure 6 presents the value of OA (in %) of the proposed methods and other tested methods for a selected number of bands.

Figure 4 show that CSA, HPSOCSA_SP, and HPSOCSA_SLP outperform GA and PSO for all four used datasets. For the Indian Pines (in Fig. 6a), when the value of n was 10, the CSA achieved the best OA (76.20%). In addition, HPSOCSA_SLP outperforms the other competitors in the majority of instances. For the KSC dataset (in Fig. 6b), when the value of n was 5, the HPSOCSA_SP achieved the best OA (88.53%). Also, HPSOCSA_SP produced higher OA than others in most of cases for this dataset. When n was 5, HPSOCSA_SP achieved the best OA (89.02%) for the Botswana dataset shown in Fig. 6c. When n was 25, HPSOCSA_SLP achieved the best OA (87.57%) on the Pavia University dataset (Fig. 6d).

The classification results obtained by different methods ($n = 10$) are presented in Tables 3, 4, 5, and 6 for the Indian Pines, KSC and Botswana, respectively. Table 3 demonstrates that the proposed models (CSA-based, HPSOCSA_SP, and HPSOCSA_SLP) obtained OA values exceeding 75% for Indian Pines data. Also, it can be seen

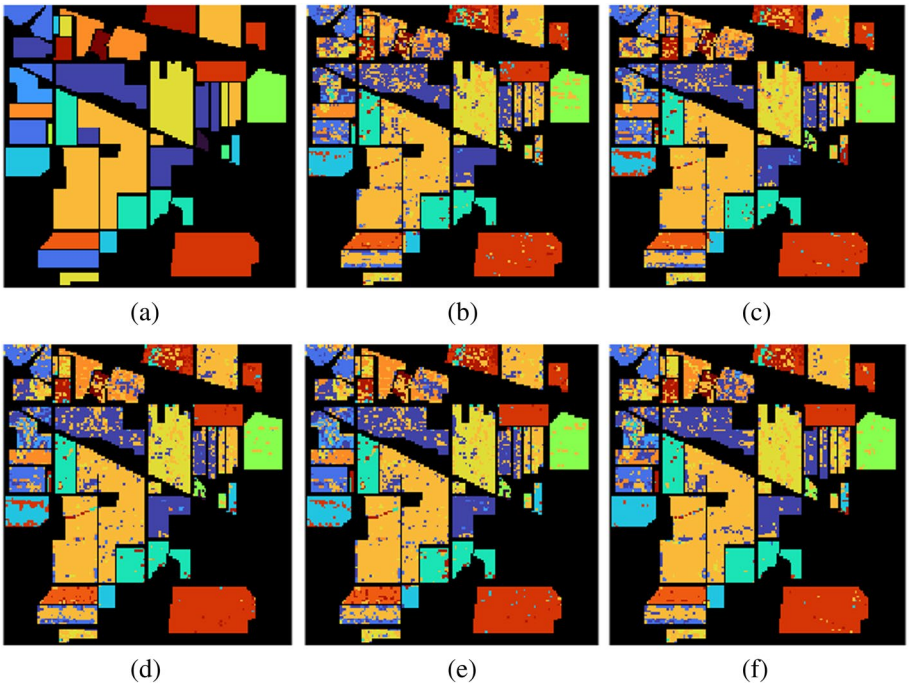

Fig. 7 Classification map for the Indian Pines dataset (**a**) Ground truth (**b**) GA (**c**) PSO (**d**) CSA (**e**) HPSOCSA_SP (**f**) HPSOCSA_SLP

that the CSA-based BS approach obtained the highest OA, AA, and Kappa value for Indian Pines. As illustrated in Table 4, the HPSOCSA_SP outperformed the other techniques for the KSC data, obtaining an OA of 86.48%, an AA of 78.73% and a Kappa score of 0.8491. Table 5 demonstrates that the proposed models achieved higher class-specific accuracy in most of the classes for the Botswana data. In addition, the HPSOCSA_SP performs better than other techniques, attaining an OA of 85.56%, an AA of 82.3%, and a Kappa score of 0.8432. Table 6 shows that the CSA-based, HPSOCSA_SP, and HPSOCSA_SLP methods outperform other methods for the Pavia University dataset. In addition, HPSOCSA_SLP achieved an OA of 85.34%, an AA of 77.52% and a Kappa score of 0.8011.

Table 7 illustrates the precision, recall and F1 score performance on all four datasets. The results show that the CSA achieved the highest precision, recall and F1 score of 73.07%, 58.34% and 62.53% on Indian Pine. HPSOCSA_SP obtained the highest precision (84.39%), recall (78.72%) and F1 score on KSC (80.51%). On the Botswana dataset, HPSOCSA_SP achieved the highest recall (82.30%) and F1 score (84.11%). HPSOCSA_SLP achieved the highest precision of 88.95%. Among the models tested on the Pavia, HPSOCSA_SLP had the highest recall (77.52%) and F1 score (80.45%), while HPSOCSA_SP achieved the highest precision of 86.68%.

The classification maps (CMs) created for the Indian Pines, KSC, Botswana, and Pavia datasets using different methods are displayed in Figs. 7, 8, 9, and 10, respectively.

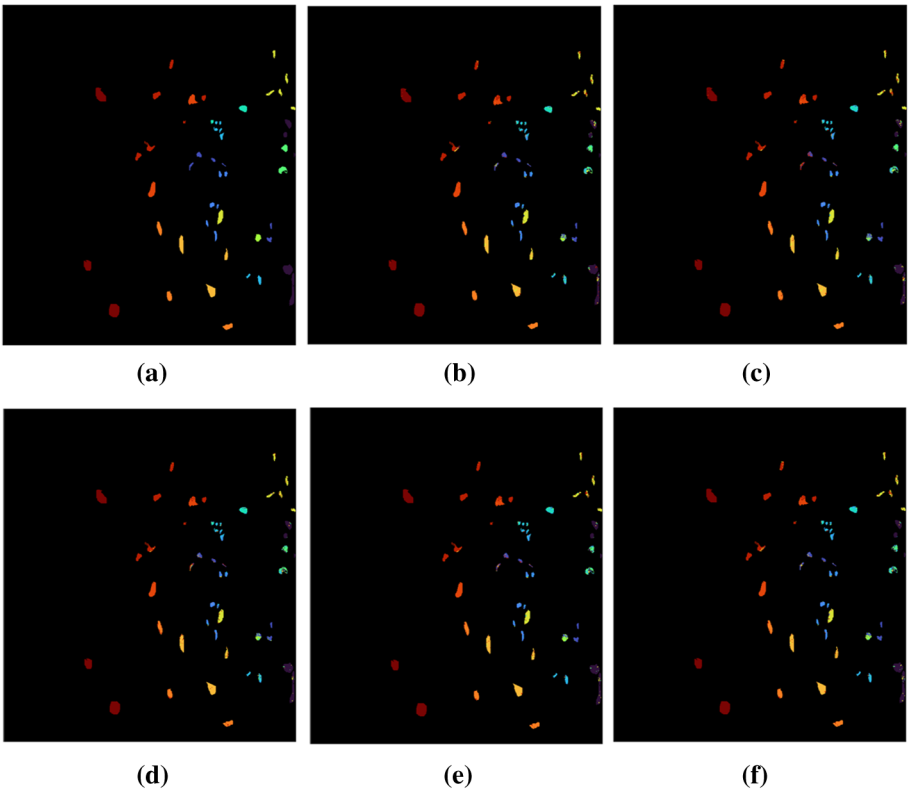

Fig. 8 Classification map for the KSC dataset (**a**) Ground truth (**b**) GA (**c**) PSO (**d**) CSA (**e**) HPSOCSA_SP (**f**) HPSOCSA_SLP

3.5 Convergence analysis

The convergence graphs using the proposed methods (CSA-based, HPSOCSA_SP, and HPSOCSA_SLP) and GA and PSO for the four mentioned datasets are presented in Fig. 11. In Fig. 11a, for Indian pines, HPSOCSA_SP converges earlier than other methods, and the CSA-based BS method reaches global solution. The convergence rate of HPSOCSA_SLP is better than GA, PSO and CSA. In addition, HPSOCSA_SP and HPSOCSA_SLP are closer to the reachable global optima. In Fig. 11b, the HPSOCSA_SLP algorithm quickly converged and obtained a near-global optimum for KSC data, while the CSA-based BS method converged late but eventually reached the global optimum. In Fig. 11c, the HPSOCSA_SLP method converges earlier for Botswana compared to other methods. Additionally, the CSA-based BS method successfully reaches a global solution. In Fig. 11d, the HPSOCSA_SLP algorithm quickly converged and obtained a near-global optimum for the Pavia data, while the CSA-based BS method converged later but eventually reached the global optimum.

Figure 11 shows that the CSA-based method achieves global optima on all four datasets. The HPSOCSA_SP method outperforms other methods in terms of convergence speed

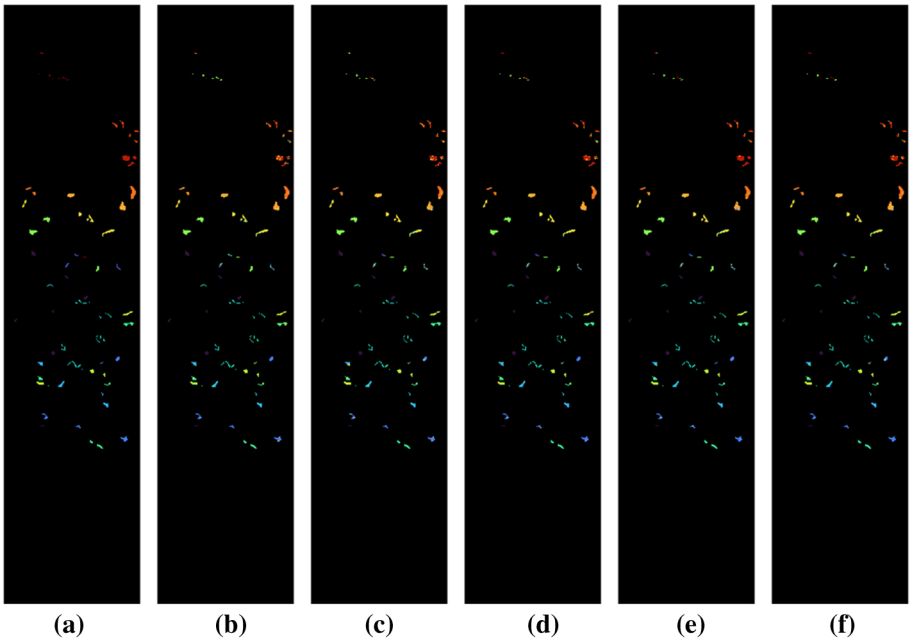

Fig. 9 Classification map for the Botswana (**a**) Ground truth (**b**) GA (**c**) PSO (**d**) CSA (**e**) HPSOCSA_SP (**f**) HPSOCSA_SLP

at Indian Pine. The HPSOCSA_SLP method demonstrates faster convergence than other methods at KSC, Botswana, and Pavia University.

3.6 Discussion

After analyzing Tables 3, 4, 5, 6 and 7, it can be concluded that the CSA-based algorithm performs the best on the Indian Pines dataset for OA, AA, Kappa score, precision, recall, and F1 score. On the other hand, HPSOCSA_SP is the best algorithm for the KSC dataset for the same metrics. In terms of recall, and F1 score, HPSOCSA_SP is the top-performing algorithm for Botswana, while HPSOCSA_SLP is the best for Pavia. In addition, HPSOCSA_SP and HPSOCSA_SLP achieve the highest overall accuracy on the Botswana and Pavia datasets, respectively. In addition, the highest OA on the Botswana and Pavia datasets is achieved through HPSOCSA_SP and HPSOCSA_SLP, respectively. On the other hand, the highest precision on the same datasets is achieved through HPSOCSA_SLP and HPSOCSA_SP. Thus, in most cases, the hybrid approach (HPSOCSA_SP or HPSOCSA_SLP) achieved the best result.

Based on the convergence graphs in Fig. 11, we can conclude that the CSA-based BS method reached the global optimum, but it took longer to converge. In addition, the HPSOCSA_SP method shows superior convergence speed at Indian Pine compared to other methods, while the HPSOCSA_SLP method demonstrates faster convergence than other methods at KSC, Botswana, and Pavia University. In most cases, the hybrid approaches,

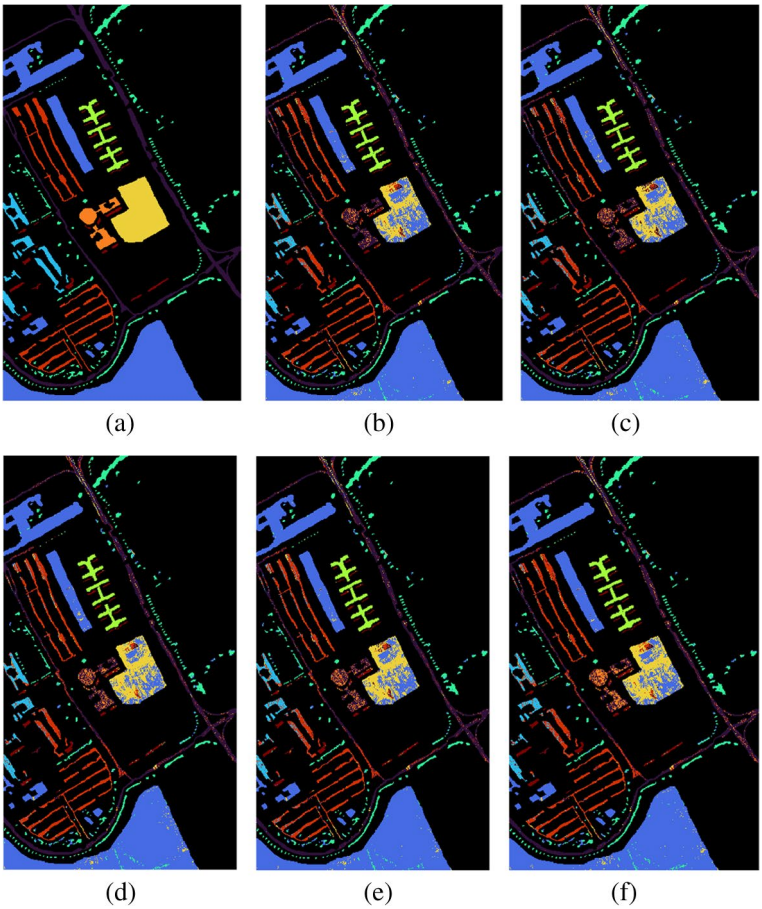

Fig. 10 Classification map for the Pavia (**a**) Ground truth (**b**) GA (**c**) PSO (**d**) CSA (**e**) HPSOCSA_SP (**f**) HPSOCSA_SLP

especially HPSOCSA_SLP, converge quickly compared to other methods. This suggests that information sharing between PSO and CSA via population sharing to balance the exploitation and exploration in each iteration works well.

4 Conclusion

The proposed research was conducted to find an optimal band subset with a high convergence rate for HSI data. In this direction, three BS methods based on metaheuristics for HSI have been applied, where the first method is based on CSA, and the other two (HPSOCSA_SP and HPSOCSA_SLP) are based on the hybridization of PSO and CSA. These three methods start with random population initialization. PSO and CSA have been hybridized to bring balance between exploration and exploitation into search

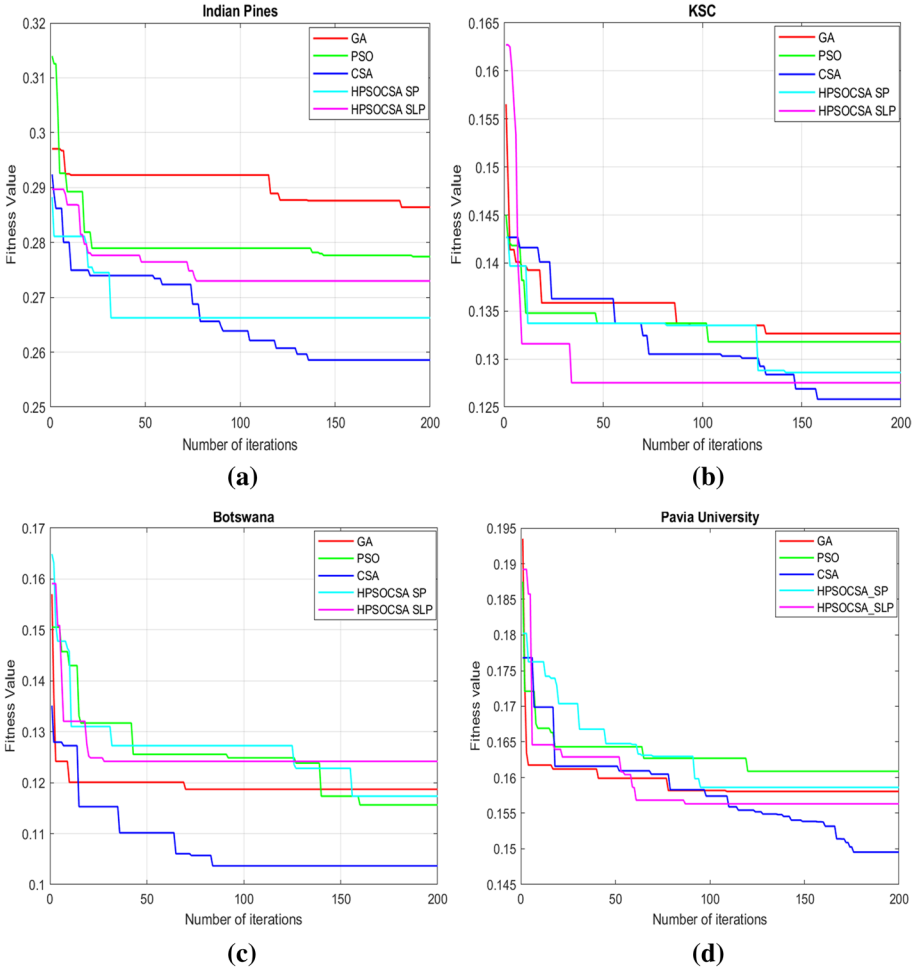

Fig. 11 Convergence graph of the proposed methods on (**a**) Indian Pines (**b**) KSC (**c**) Botwana (**d**) Pavia

progress. The population in HPSOCSA_SP is divided equally in half. PSO is used on one part, while CSA is used on the other. Based on fitness, HPSOCSA_SLP selects half of the best-performing members. The selected population receives successive applications of PSO and CSA. Then the resultant parts are integrated.

In the experiment, benchmark HSI datasets: Indian Pines, KSC, Botswana, and Pavia University have been used. CSA-based, HPSOCSA_SP, and HPSOCSA_SLP achieved OA values exceeding 75% for Indian Pines data ($n = 10$). On dataset KSC, HPSOCSA_SP obtains the OA of 86.48%, the AA of 78.73%, and the Kappa score of 0.8491. On dataset Botswana, HPSOCSA_SP attains the OA of 85.56%, the AA of 82.3%, and the Kappa score of 0.8432.On Pavia dataset, HPSOCSA_SLP attains the OA of 85.34%, the AA of 77.52.3%, and the Kappa score of 0.8011.The outcome demonstrates that proposed methods perform better than PSO and GA and offers steady outcomes. Compared to other methods CSA-based method obtains global optima on all four datasets. On Indian Pine, HPSOCSA_SP converges quickly compared to other methods. The

HPSOCSA_SLP method proves to outshine other methods in terms of faster convergence at KSC, Botswana, and Pavia University.

Data availability The data (outputs) generated or analyzed during this study are included in this manuscript.

Declarations

Conflict of interest The authors declare that they have no known competing financial or personal relationships that could be viewed as influencing the work reported in this paper.

Ethics approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Sun W, Du Q (2019) Hyperspectral band selection: a review. IEEE Geosci Remote Sens Mag 7(2):118–139. https://doi.org/10.1109/MGRS.2019.2911100
2. Gao H, Lin S, Li C, Yang Y (2019) Application of hyperspectral image classification based on overlap pooling. Neural Process Lett 49(3):1335–1354. https://doi.org/10.1007/s11063-018-9876-7
3. Zhang W, Li X, Zhao L (2018) A fast hyperspectral feature selection method based on band correlation analysis. IEEE Geosci Remote Sens Lett 15(11):1750–1754. https://doi.org/10.1109/LGRS.2018.2853805
4. Salimi A, Ziaii M, Amiri A, Hosseinjani M, Karimpouli S, Moradkhani M (2018) Using a feature subset selection method and support vector machine to address curse of dimensionality and redundancy in Hyperion hyperspectral data classification. Egypt J Remote Sens Sp Sci 21(1):27–36. https://doi.org/10.1016/j.ejrs.2017.02.003
5. Yan Y, Yu W, Zhang L (2022) A method of band selection of remote sensing image based on clustering and intra-class index. Multimed Tools Appl 81:22111–22128. https://doi.org/10.1007/s11042-021-11865-1
6. Groves P, Bajcsy P (2004) Methodology for hyperspectral band and classification model selection. 2003 IEEE Work Adv Tech Anal Remote Sensed Data, vol. 00, no. C, pp 120–128. https://doi.org/10.1109/WARSD.2003.1295183
7. Patro RN, Subudhi S, Biswal PK, Dell'acqua F (2021) A review of unsupervised band selection techniques. IEEE Geosci Remote Sens Mag 9(3):72–111. https://doi.org/10.1109/MGRS.2021.3051979
8. Licciardi G, Marpu PR, Chanussot J, Benediktsson JA (2012) Linear versus nonlinear PCA for the classification of hyperspectral data based on the extended morphological profiles. IEEE Geosci Remote Sens Lett 9(3):447–451. https://doi.org/10.1109/LGRS.2011.2172185
9. Villa A, Benediktsson JA, Chanussot J, Jutten C (2011) Hyperspectral image classification with independent component discriminant analysis. IEEE Trans Geosci Remote Sens 49(12 PART 1):4865–4876. https://doi.org/10.1109/TGRS.2011.2153861
10. Gao P, Wang J, Zhang H, Li Z (2019) Boltzmann entropy-based unsupervised band selection for hyperspectral image classification. IEEE Geosci Remote Sens Lett 16(3):462–466. https://doi.org/10.1109/LGRS.2018.2872358
11. Li S, Wu H, Wan D, Zhu J (2011) An effective feature selection method for hyperspectral image classification based on genetic algorithm and support vector machine. Knowl-Based Syst 24(1):40–48. https://doi.org/10.1016/j.knosys.2010.07.003
12. Kohavi R, John GH (1997) Wrappers for feature subset selection. Artif Intell 97(1–2):273–324. https://doi.org/10.1016/s0004-3702(97)00043-x
13. Paul A, Chaki N (2021) Dimensionality reduction of hyperspectral image using signal entropy and spatial information in genetic algorithm with discrete wavelet transformation. Evol Intell 14:1793–1802. https://doi.org/10.1007/s12065-020-00460-2
14. Wang M, Wu C, Wang L, Xiang D, Huang X (2019) A feature selection approach for hyperspectral image based on modified ant lion optimizer. Knowl-Based Syst 168:39–48. https://doi.org/10.1016/j.knosys.2018.12.031
15. Archibald R, Fann G (2007) Feature selection and classification of hyperspectral images with support vector machines. IEEE Geosci Remote Sens Lett 4(4):674–677. https://doi.org/10.1109/LGRS.2007.905116

16. Chaudhuri A, Sahu TP (2021) Feature selection using binary crow search algorithm with time varying flight length. Expert Syst Appl, vol. 168, https://doi.org/10.1016/j.eswa.2020.114288
17. Rashedi E, Rashedi E, Nezamabadi-pour H (2018) A comprehensive survey on gravitational search algorithm. Swarm Evol Comput 41(January):141–158. https://doi.org/10.1016/j.swevo.2018.02.018
18. Yang H, Du Q, Chen G (2012) Particle swarm optimization-based hyperspectral dimensionality reduction for urban land cover classification. IEEE J Sel Top Appl Earth Obs Remote Sens 5(2):544–554. https://doi.org/10.1109/JSTARS.2012.2185822
19. Medjahed SA, Ait Saadi T, Benyettou A, Ouali M (2016) Gray wolf optimizer for hyperspectral band selection. Appl Soft Comput J 40:178–186. https://doi.org/10.1016/j.asoc.2015.09.045
20. Xie F, Li F, Lei C, Yang J, Zhang Y (2019) Unsupervised band selection based on artificial bee colony algorithm for hyperspectral image classification. Appl Soft Comput J 75:428–440. https://doi.org/10.1016/j.asoc.2018.11.014
21. Sawant SS, Manoharan P (2019) New framework for hyperspectral band selection using modified wind-driven optimization algorithm. Int J Remote Sens 40(20):7852–7873. https://doi.org/10.1080/01431161.2019.1607609
22. Phaneendra Kumar BLN, Manoharan P (2021) Whale optimization-based band selection technique for hyperspectral image classification. Int J Remote Sens 42(13):5109–5147. https://doi.org/10.1080/01431161.2021.1906979
23. Ghosh A, Datta A, Ghosh S (2013) Self-adaptive differential evolution for feature selection in hyperspectral image data. Appl Soft Comput J 13:1969–1977. https://doi.org/10.1016/j.asoc.2012.11.042
24. Su H, Du Q, Chen G, Du P (2014) Optimized hyperspectral band selection using particle swarm optimization. IEEE J Sel Top Appl Earth Obs Remote Sens 7(6):2659–2670. https://doi.org/10.1109/JSTARS.2014.2312539
25. Xu Y, Du Q, Younan NH (2017) Particle swarm optimization-based band selection for hyperspectral target detection. IIEEE Geosci Remote Sens Lett 14(4):554–558. https://doi.org/10.1109/LGRS.2017.2658666
26. Ghamisi P, Benediktsson JA (2015) Feature selection based on hybridization of genetic algorithm and particle swarm optimization. IEEE Geosci Remote Sens Lett 12(2):309–313. https://doi.org/10.1109/LGRS.2014.2337320
27. Aghaee R, Momeni M, Moallem P (2022) Semisupervised band selection from hyperspectral images using levy flight-based genetic algorithm. IEEE Geosci Remote Sens Lett 19. https://doi.org/10.1109/LGRS.2022.3147272
28. Medjahed SA, Saadi TA, Benyettou A, Ouali M (2015) Binary cuckoo search algorithm for band selection in hyperspectral image classification. IAENG Int J Comput Sci 42(3):1–9
29. Sawant S, Manoharan P (2021) A hybrid optimization approach for hyperspectral band selection based on wind driven optimization and modified cuckoo search optimization. Multimed Tools Appl 80(2):1725–1748. https://doi.org/10.1007/s11042-020-09705-9
30. Wang M, Liu W, Chen M, Huang X, Han W (2021) A band selection approach based on a modified gray wolf optimizer and weight updating of bands for hyperspectral image. Appl Soft Comput, vol. 112, https://doi.org/10.1016/j.asoc.2021.107805
31. Tschannerl J et al (2019) MIMR-DGSA: unsupervised hyperspectral band selection based on information theory and a modified discrete gravitational search algorithm. Inf Fus 51:189–200. https://doi.org/10.1016/j.inffus.2019.02.005
32. Ding X et al (2020) An improved ant Colony algorithm for optimized band selection of hyperspectral remotely sensed imagery. IEEE Access 8:25789–25799. https://doi.org/10.1109/ACCESS.2020.2971327
33. Anand R, Samiaappan S, Veni S, Worch E, Zhou M (2022) Airborne Hyperspectral Imagery for Band Selection Using Moth–Flame Metaheuristic Optimization. J Imaging, vol. 8, no. 5, https://doi.org/10.3390/jimaging8050126
34. Wang M, Yan Z, Luo J, Ye Z, He P (2021) A band selection approach based on wavelet support vector machine ensemble model and membrane whale optimization algorithm for hyperspectral image. Appl Intell 51(11):7766–7780. https://doi.org/10.1007/s10489-021-02270-0
35. Eberhart R, Kennedy J (1995) A new optimizer using particle swarm theory. MHS'95. Proc. sixth Int. Symp. Micro Mach. Hum. Sci. Nagoya, Japan, pp 39–43, https://doi.org/10.1109/MHS.1995.494215
36. Samee NA et al (2022) Metaheuristic optimization through deep learning classification of COVID-19 in chest x-ray images. Comput Mater Contin 73(2):4193–4210. https://doi.org/10.32604/cmc.2022.031147
37. Hamadache I, Mellal MA (2021) Design optimization of a car side safety system by particle swarm optimization and grey wolf optimizer. In: Nature-Inspired Computing Paradigms in Systems. Academic Press, Elsevier Inc., pp 15–24. https://doi.org/10.1016/B978-0-12-823749-6.00006-4
38. Yang X-S, Chien SF, Ting TO (2015) Bio-inspired computation and optimization: an overview. Bio-inspired computation in telecommunications, pp 1–21. https://doi.org/10.1016/B978-0-12-801538-4.00001-X

39. Askarzadeh A (2016) A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. Comput Struct 169:1–12. https://doi.org/10.1016/j.compstruc.2016.03.001
40. Adamu A, Abdullahi M, Junaidu SB, Hassan IH (2021) An hybrid particle swarm optimization with crow search algorithm for feature selection. Mach Learn Appl 6:100108. https://doi.org/10.1016/j.mlwa.2021.100108
41. Hassan E, Shams MY, Hikal NA, Elmougy S (2023) The effect of choosing optimizer algorithms to improve computer vision tasks: a comparative study. Multimed Tools Appl 82(11):16591–16633. https://doi.org/10.1007/s11042-022-13820-0
42. Li Y, Zhang H, Xue X, Jiang Y, Shen Q (2018) Deep learning for remote sensing image classification: A survey. Wiley Interdiscip Rev Data Min Knowl Discov 8(6):1–17. https://doi.org/10.1002/widm.1264
43. Gamel SA, Hassan E, El-Rashidy N, Talaat FM (2023) Exploring the effects of pandemics on transportation through correlations and deep learning techniques. Multimed Tools Appl, no. 0123456789, https://doi.org/10.1007/s11042-023-15803-1
44. Hssayni EH, Joudar NE, Ettaouil M (2022) An adaptive drop method for deep neural networks regularization: estimation of DropConnect hyperparameter using generalization gap. Knowl-Based Syst, vol. 253, https://doi.org/10.1016/j.knosys.2022.109567
45. Hssayni EH, Joudar N-E, Ettaouil M (2022) A deep learning framework for time series classification using normal cloud representation and convolutional neural network optimization. Comput Intell, vol. 38, https://doi.org/10.1111/coin.12556
46. Wang C, Ma N, Ming Y, Wang Q, Xia J (2019) Classification of hyperspectral imagery with a 3D convolutional neural network and J-M distance. Adv Sp Res 64(4):886–899. https://doi.org/10.1016/j.asr.2019.05.005
47. Hssayni EH, Joudar NE, Ettaouil M (2022) Localization and reduction of redundancy in CNN using L 1-sparsity induction. J Ambient Intell Humaniz Comput, https://doi.org/10.1007/s12652-022-04025-2