Two-Stage Evolutionary Algorithm Based on Subspace Specified Searching for Hyperspectral Endmember Extraction

Cong Lei, Rong Liu, and Ye Tian

***Abstract*—In recent years, the introduction of multiobjective evolutionary algorithms (MOEAs) into the field of endmember extraction (EE) in hyperspectral unmixing has demonstrated a breadth of results that surpass those derived from single-objective-based methodologies. Despite these advancements, the adaptation of MOEAs to EE and the attainment of globally optimal solutions represent unresolved challenges meriting continued exploration. This study addresses two principal obstacles in MOEA-based EE: the notorious "curse of dimensionality" in high-dimensional optimization, and the difficulty in striking a balance between convergence and population diversity. We propose a two-stage, evolutionary-based EE algorithm, referred to as TSEA, designed to confront these issues. A novel solution space splitting strategy is incorporated into TSEA that efficiently mitigates the curse of dimensionality by strategically contracting the search space. This advantage is largely attributed to the significant reduction of invalid solutions achieved through the simple application of a clustering procedure. Furthermore, a two-stage optimization approach is employed to meticulously uphold the convergence and diversity of the population, aiming to attain the optimal solution within the realm of high-dimensional optimization. Empirical evidence from four real hyperspectral images demonstrates that the proposed TSEA outperforms other comparison multiobjective optimization algorithms. Thus, this study contributes to the ongoing discourse on the optimization and applicability of MOEAs in the context of EE.**

***Index Terms*—Endmember extraction (EE), hyperspectral remote sensing, multiobjective optimization.**

I. Introduction

HYPERSPECTRAL image (HSI) usually contains hundreds of observation channels, which makes it possible to address various application problems requiring very high discrimination capabilities in the spectral domain [1], [2], [3], [4]. One big challenge that contradicts the application of HSI is the inevitably prevalent mixed pixels due to the limitation of spatial resolution and heterogeneity of ground features [5].

Hyperspectral unmixing (HU) is an important technique for solving the mixed pixel problem [6], which takes aim at decomposing each mixed pixel into the constituent components (endmembers) and the corresponding coefficients (abundances). For this purpose, HU typically contains two tasks: endmember extraction (EE) and abundance estimation. The two tasks can be accomplished either conjointly or separately [7]. Representative approaches with the first manner involve the nonnegative matrix factorization-based algorithms [8], [9], deep learning-based algorithms [10], [11], [12], and unmixing algorithms considering spectral variability [13], [14], [15]. The second way estimates the abundance based on existent endmembers, frequently used are the sparse regression-based algorithms [16], [17], [18], and the multiple endmember spectral mixture analysis-based algorithms [19], [20]; endmembers can be either acquired from an existing library or automatically collected from the image [21], [22]. Image-based endmembers are more accessible and affordable than library-endmembers. EE algorithms mainly contain four categories: traditional pure pixel assumption-based algorithms [23], [24], [25], minimum volume-based algorithms [26], [25], [28], spatial preprocessing-based algorithms [29], [30], [31], and evolutionary computation-based algorithms [32], [33], where evolutionary computation-based algorithms belong to a newly developed branch for EE.

The main advantage of evolutionary computation-based EE algorithms lies in the powerful capability of evolutionary computation for solving nonconvex optimization problems, which enable the optimization of complex but robust model, thus handling the drawbacks within existing algorithms such as sensitivity to noises or outliers [34], and breaking limits of the application on real hyperspectral scenes.

Scholars have made attempts to apply evolutionary computation to solve various issues in the HU field, such as EE [35], [36], and sparse unmixing [37], [38], [39]. Works that appertain to EE can be divided into the following three categories.

1) *Single-objective EE*: At the outset, Zhang et al. [32] employed single-objective evolutionary algorithms, the discrete particle swarm optimization (DPSO) algorithm and the ant colony optimization (ACO) algorithm [33], to minimize the root-mean-square error (RMSE) between the original image and its reconstructed image, and

Manuscript received 22 August 2023; revised 16 October 2023; accepted 6 November 2023. Date of publication 17 November 2023; date of current version 4 December 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62201622 and Grant 62276001, in part by the Basic and Applied Research Project of Guangzhou Science and Technology Plan under Grant 2023A04J1988, in part by the Outstanding Young Program of Universities in Anhui Province under Grant 2022AH030013, and part by the GHfund B(202302024219). *(Corresponding authors: Rong Liu; Ye Tian.)*

Cong Lei and Rong Liu are with the School of Geography and Planning, Sun Yat-sen University, Guangzhou 510275, China (e-mail: leic@mail2.sysu.edu.cn; liurong25@mail.sysu.edu.cn).

Ye Tian is with the Information Materials and Intelligent Sensing Laboratory of Anhui Province, Institutes of Physical Science and Information Technology, Anhui University, Hefei 230601, China (e-mail: field910921@gmail.com).

Digital Object Identifier 10.1109/JSTARS.2023.3333955

achieved evidently improvement compared to traditional algorithms. This was the first time that the RMSE objective function was successfully used in EE, and it was demonstrated in the later work proposed by Liu et al. [34] that the RMSE objective function is robust to noise and outliers. Besides, the conventional objective function, the volume of the simplex composed of extracted endmembers, was also used in the single-objective evolutionary algorithms [40]. In the past decade, miscellaneous algorithms have been designed either to improve the efficiency or enhance the optimization results of single-objective EE [41], [42], [43], [44], [45], [46], [47], [48], [49], [50], [51], including a variety of evolutionary algorithms, such as PSO algorithms [32], [34], [40], [41], [42], ACO algorithms [33], [43], [44], [45], genetic algorithms [46], [47], and differential evolution (DE) algorithm [48].

2) *Multiobjective EE*: Due to complicated imaging conditions, real hyperspectral scenes may be corrupted with interferences. Researchers found that the reconstructed error and volume objective functions come to a disagreement, posing a tough dilemma to us about which objective function to choose. To better suit real-world applications, multiobjective EE was put forward to simultaneously optimize the two conflict objective functions, thus providing adjustable options to users. The multiobjective DPSO (MODPSO) algorithm [52] was originally proposed to produce tradeoff solutions between the reconstructed error and volume objective functions. Later on, several improved algorithms [53], [54], [55], [56] were developed with the purpose of reaching a better Pareto front (PF). For example, Cheng et al. [53] proposed a global-to-local evolutionary algorithm (GL-EA) where the global and local search is performed sequentially to extract the endmembers effectively; the improved MODPSO (IMODPSO) algorithm [54] was presented to enhance the results by utilizing the archive strategy. Tong et al. [56] proposed the $(\mu+\lambda)$ multiobjective DE algorithm based on ranking multiple mutations, which further improved the performance of IMODPSO.

3) *Endmember bundle extraction (EBE)*: EBE is a category of algorithms that consider spectral variability of endmembers caused by intra-class variability of materials or various imaging conditions [57], [58]. EBE uses multiple instances of spectra to represent an endmember class, which can result in better unmixing results than EE in scenes with spectral variability, although more complicated and time-consuming. Recently, multiobjective evolutionary algorithms (MOEAs) have been exploited in EBE [59], [60], and captured more endmember variabilities than existing representative algorithms, exhibiting considerable potential for application in EBE.

Sketching from the above researches, although evolutionary algorithms-based EE were ushered burgeoning in the past few years, the newly emerging multiobjective EE and EBE are still in the early stage. In this article, we focus on the study of multiobjective EE, revolving around two significant challenges. One challenge is the "curse of dimensionality" problem encountered in high-dimensional optimization [61], that is, the performance of the optimization algorithm deteriorates as the dimensionality of the search space increases. The other challenge is how to well maintain the convergence and the diversity of the population, so as to obtain the optimal solution using as few times of iterations as possible. Most existing MOEA-based EE methods use the same update rule to produce offspring and adopt the same environmental selection strategy during the whole optimization process. These algorithms are devoted to improving the convergence degree while ignoring the diversity, which may lead the population into local optima and result in premature convergence.

Dig deeply into the endmember combinations, in fact large number of combinations are invalid solutions to the EE problem. For example, if one component of the individual corresponds to an endmember of material #1, and another component of the individual also corresponds to an endmember of material #1, this will be a bad or invalid solution since we need to extract one endmember for one kind of material. Unfortunately, this kind of problem frequently happens during optimization because all components of the individual are possible to be assigned any values in the whole search space without limitation. Hence, it is inexpedient to find each optimal endmember in the whole search space, and strategy should be taken to narrow down the search space.

To solve the above problems, in this article, we proposed a two-stage EA (TSEA) based on subspace-specified searching for hyperspectral EE. Considering that bad endmember combinations contain the repetition of one or more endmember classes, TSEA restructures the original image based on clustering to allow each component of the individual search in the subspace constructed by valid solutions. First, the subspace limitation can avoid producing lots of invalid solutions; second, the new-built search space is considerably reduced than the original space, which can greatly benefit the search of optimal solutions. In addition, a two-stage optimization strategy is employed in TSEA to well maintain the exploration and exploitation of the population. In the first stage, TSEA aims at accelerating the convergence of the population by selecting two well-converged solutions as parents to generate an offspring and removing the solution with the worst convergence degree. In the second stage, TESA pays more attention to the population's diversity. One solution with a good convergence degree and another solution with good diversity are selected as parents, and a fine-tuned environmental selection strategy is adopted to enhance diversity.

The rest of this article is structured as follows. Section II introduces the linear spectral mixture model (LSMM) and the conception of multiobjective optimization briefly. In Section III, we present the framework and details of TSEA. The experimental results on four HSIs are displayed in Section IV and the discussion is given in Section V. Finally, Section VI concludes this article.

II. Related Works

A. Linear Spectral Mixture Model (LSMM)

The LSMM [6] considers each pixel in the image as a linear combination of the endmember spectra, and each endmember spectrum is weighted by its corresponding abundance. For a

pixel $\mathbf{y} = (y_1, y_2, \ldots, y_L)^{\mathrm{T}}$ in the HSI with L bands, it can be expressed as follows:

$$\mathbf{y} = \sum_{i=1}^{P} \mathbf{a}_i s_i + \mathbf{e} = \mathbf{A}\mathbf{s} + \mathbf{e} \tag{1}$$

where P is the number of endmembers, $\mathbf{a}_i$ and s_i represent the endmember vector, and the abundance of the ith endmember, respectively. $\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_P)$ is the endmember matrix with the size of $L \times P$. $\mathbf{s} = (s_1, s_2, \ldots, s_P)^{\mathrm{T}}$ is a $P \times 1$ abundance vector and $\mathbf{e}$ is an $L \times 1$ error vector which represents the observation noise and modeling error. Supposing that there are N pixels in the image, the $L \times N$ image matrix $\mathbf{Y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_N)$ can be written as follows:

$$\mathbf{Y} = \mathbf{A}\mathbf{S} + \mathbf{E} \tag{2}$$

where $\mathrm{S} = (s_1, s_2, \ldots, s_{\mathrm{N}})$ denotes the P×N abundance matrix, and $\mathrm{E} = (\mathrm{e}_1, \mathrm{e}_2, \ldots, \mathrm{e}_{\mathrm{N}})$ denotes the L×N error matrix.

B. Multiobjective Optimization

In multiobjective problems (MOPs), at least two objectives need to be optimized simultaneously [62]. Without loss of generality, an MOP can be written as follows for the minimization optimization:

$$\begin{aligned} &\min\ F(\mathbf{x}) = [f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})] \\ &\text{s.t. } \mathbf{x} \in \Omega \end{aligned} \tag{3}$$

where m denotes the number of objectives, $f_i(\mathbf{x})$ denotes the value of the ith objective function, $\mathbf{x}$ is the decision variable that represents a solution to the problem, and Ω represents the decision space. There is no solution that can minimize the m objective functions simultaneously because conflicts exist among the objectives. If one objective function reaches a smaller value, the value of another objective function may be larger. To solve the above problem, the domination relationship is used to establish an evaluation criterion for different solutions. A solution $\mathbf{x}$ is said to dominate $\mathbf{y}$ if and only if

$$\begin{cases} \forall i \in [1, 2, \ldots, m], f_i(\mathbf{x}) \le f_i(\mathbf{y}) \\ \exists j \in [1, 2, \ldots, m], f_j(\mathbf{x}) < f_j(\mathbf{y}) \end{cases}. \tag{4}$$

If a solution is not dominated by any other solutions in Ω, it is called a Pareto optimal solution or nondominated solution. The set of the Pareto solutions is called the Pareto set (PS) and the objective function values of the PS are called the PF. The main aim of the MOEAs is to find a set of nondominated solutions which can tradeoff different objectives best and make the result approach the real PF as close as possible.

III. TSEA

A. Objective Functions of TSEA

Following the existing MOEA-based EE algorithms, two LSMM-based objective functions are used in TSEA, namely, the maximization of volume and the minimization of RMSE between the original image and its reconstructed image. On ideal condition, the pixels that obtain the maximum volume are endmembers, and unmixing using optimal endmembers will result in minimum reconstructed error. However, real images are usually corrupted by noise and other perturbations. Maximum volume function can be easily affected by abnormal spectra and noise. The RMSE objective function is robust to abnormal spectra and noise, nevertheless mixed pixels can probably be mistakenly extracted as endmembers when the defined number of endmembers is smaller than the real number or spectral variability exists in the image. TSEA aims to optimize the two objective functions simultaneously. For convenience, the maximization of volume is translated into a minimization problem. The two functions are listed as follows:

$$\begin{aligned} &\min : F(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x})) \\ &f_1(\mathbf{x}) = \frac{1}{\mathrm{volume}(\mathbf{A}(\mathbf{x}))} = \frac{(P-1)!}{\left|\det \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \mathbf{a}_1 & \mathbf{a}_2 & \cdots & \mathbf{a}_P \end{bmatrix}\right|} \\ &f_2(\mathbf{x}) = \mathrm{RMSE}(\mathbf{Y}, \hat{\mathbf{Y}}) = \frac{1}{N}\sum_{i=1}^{N}\sqrt{\frac{1}{L}\|\mathbf{y}_i - \hat{\mathbf{y}}_i\|^2} \end{aligned} \tag{5}$$

where $\mathbf{x}$ is the decision variable, and each component represents the index of the extracted endmember in the image. $\mathbf{A}(\mathbf{x})$ is the endmember matrix, P denotes the number of endmembers, N denotes the total number of pixels, L denotes the number of bands, and $\boldsymbol{a}_i (i = 1, 2, \ldots, P)$ is a $(P-1)$-dimensional vector which is transformed from the original endmember spectral vector by minimum noise fraction [63]. $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ represent the original image and the reconstructed image, respectively. The abundance for reconstructing the image is calculated by the unconstrained least squares [52] using the following formulas:

$$\begin{aligned} &\hat{\mathbf{S}}_{ij} = \max\left(0, \left(\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\right)^{-1}\mathbf{A}^{\mathrm{T}}\mathbf{Y}\right)_{ij}\right), 1 \le i \le P, 1 \le j \le N \\ &\hat{\mathbf{Y}} = \mathbf{A}\hat{\mathbf{S}} \end{aligned} \tag{6}$$

where $\mathbf{A}$ is the endmember matrix, and $\hat{\mathbf{S}}$ is the estimated abundance matrix.

B. Framework of TSEA

The flowchart of TSEA is presented in Fig. 1. To effectively alleviate the "curse of dimensionality problem" and avoid producing large amounts of invalid solutions, TSEA elaborately constructs subspaces for each component of the decision variable. Instead of searching all endmembers from the indistinguishably intermixed original space, each component of the decision variable is limited to search in the corresponding subspace. In this way, each component of the decision variable is attached by a specific task of searching one kind of endmember, and large amounts of invalid endmember combinations are eliminated, which greatly benefits the optimization process. To well maintain the exploration and exploitation of the population, TSEA divides the evolution process into two stages and adopts different mating and environmental selection strategies at different stages. The first stage aims at improving the convergence and the second stage focus on improving the diversity of population. By

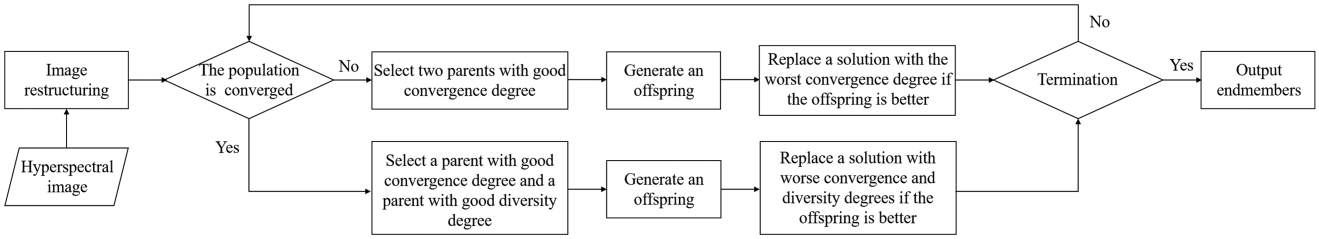


Fig. 1. Flowchart of TSEA.

Algorithm 1: Pseudocode of TSEA.

Input: $\mathbf{Y}\in R^{L\times N}$ (hyperspectral image), *NP* (population size), *P* (number of endmembers), *maxiter* (maximum number of iterations)
Output: **X** (offspring)
1 Restructure the image by method described in Section III-C;
2 Randomly initialize the population *pop*;
3 Calculate the nondominated front number of each solution in **X**, and the result is denoted by *Front*;
4 Calculate the convergence degree of each solution in **X** by (8), and the result is denoted by *Con*;
5 Calculate the diversity degree of each solution in **X** by (9), and the result is denoted by *Div*;
6 Set $t = 1$, $\mathbf{X} = \emptyset$;
7 **while** $t <$ *maxiter*
8 *stage* = DeterminingStage(**X**, *Front*);
9 [$parent_1$, $parent_2$] = MatingSelection(**X**, *stage*, *Front*, *Con*, *Div*);
10 Generate offspring **o** by (10) and (12);
11 $\mathbf{X} = \mathbf{X}\cup\mathbf{o}$;
12 Calculate *Front*, *Con,* and *Div* for **o**;
13 **X** = EnvironmentalSelection(**X**, **o**, *stage*, *Front*, *Con*, *Div*);
14 $t = t + 1$;
15 **end while**

using the two-stage evolution strategy, TSEA is able to obtain a well-converged population with good diversity.

The pseudocode of TSEA is displayed in Algorithm 1. First, pixels in the original image are clustered. The image is restructured according to the result of clustering and the vectors *lower* and *upper* represent the lower and upper bounds of the clusters. Then, the population **X** is randomly initialized under the constraints of *lower* and *upper*. TSEA adopts the integer encoding method. Each solution is a $1\times P$ vector and each element in the vector represents a candidate endmember. Assuming that there exist three kinds of endmembers in the HSI, the endmembers represented by the solution $\mathbf{x}$ = (17,55,96) are the 17th, 55th and 96th pixels of the image. The position of the *i*th solution $\mathbf{x}_i$ = ($\mathbf{x}_{i,1}$,$\mathbf{x}_{i,2}$,…,$\mathbf{x}_{i,P}$) is initialized as follows:

$$\mathbf{x}_{i,j} = \text{floor}\left(rand \times (upper_j - lower_j + 1)\right) + lower_j \quad (7)$$

where $\mathbf{x}_{i,j}$ defines the *j*th element of $\mathbf{x}_i$, *rand* is a random number in the range of [0,1], and $lower_j$ and $upper_j$ represent the *j*th elements of *lower* and *upper* respectively. The subspace restructuring strategy makes each component of $\mathbf{x}_i$ search in the corresponding subspace. Under the constraints of the specified subspaces, the *i*th solution $\mathbf{x}_i$ in the population can be written as

$$\mathbf{x}_i = (x_{i,1}, x_{i,2}, \ldots x_{i,P})$$
$$\text{s.t. } lower_j \le x_{i,j} \le upper_j, x_{i,j} \in Z, j = 1, 2, \ldots, P. \quad (8)$$

The front number *Front*, the convergence degree *Con*, and the diversity degree *Div* are then calculated. The front number *Front* is obtained by the nondominated sorting approach [64]. To measure the convergence degree of each solution, the criterion used in MOEAs is adopted and can be written as [65]

$$Con(\mathbf{x}) = f_1(\mathbf{x}) + f_2(\mathbf{x}) \quad (9)$$

where $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ denote the volume inverse and the RMSE value of solution **x**, respectively. The convergence degree of **x** is the sum of its objective function values. As for the calculation of diversity degree, TSEA adopts the method in SPEA2 [66] and it is defined as

$$Div(\mathbf{x}) = \|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{x}_1)\| + \varepsilon\|\mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{x}_2)\| \quad (10)$$

where $\mathbf{f}(\mathbf{x})$ denotes the vector composed of objective function values of **x**, $\|\mathbf{f}(\mathbf{x})-\mathbf{f}(\mathbf{x}_1)\|$ denotes the Euclidean distance between **x** and its closest neighbor $\mathbf{x}_1$ in the objective space, and $\|\mathbf{f}(\mathbf{x})-\mathbf{f}(\mathbf{x}_2)\|$ denotes the Euclidean distance between **x** and its second closest neighbor $\mathbf{x}_2$ in the objective space. ε is a tiny value, for example, 10^{-6}. $\varepsilon\|\mathbf{f}(\mathbf{x})-\mathbf{f}(\mathbf{x}_2)\|$ is used to distinguish the diversity degree of two very close solutions. If the distance between solution **x** and **y** is small enough, $\|\mathbf{f}(\mathbf{x})-\mathbf{f}(\mathbf{x}_1)\|$ and $\|\mathbf{f}(\mathbf{y})-\mathbf{f}(\mathbf{y}_1)\|$ will be the same value because each of them is the closest neighbor for the other. By introducing $\varepsilon\|\mathbf{f}(\mathbf{x})-\mathbf{f}(\mathbf{x}_2)\|$ and $\varepsilon\|\mathbf{f}(\mathbf{y})-\mathbf{f}(\mathbf{y}_2)\|$, there will be a slight difference between $Div(\mathbf{x})$ and $Div(\mathbf{y})$, which contributes to the selection of solutions with better diversity in the subsequent steps of TSEA. The basic idea of the diversity degree is that the farther the distance between solutions, the more diverse the distribution of population in the objective function space. If each solution is far away from its closest neighbor, the population will spread evenly in the objective space instead of gathering together, thus avoiding the deterioration of diversity.

In the main loop of TSEA, the stage of the evolution process is determined first. As displayed in Algorithm 2, if there exists any solution with front number larger than 1, the current stage is set to the first stage. Otherwise, the current stage is set to the second stage. In other words, when all the solutions are

Algorithm 2: Determining Stage.

Input: $\mathbf{X}$ (current population), *Front* (front number of each solution)
Output: *stage*
1 **if** $\max_{\mathbf{x}\in\mathbf{X}} Front(\mathbf{x}) > 1$
2 $stage = 1$;
3 **else**
4 $stage = 2$;
5 **end if**

nondominated, which means the population is well-converged, the second stage will be used. Then, two solutions are selected from the population to produce an offspring $\mathbf{o}$ by the simulated binary crossover (SBX) [67] and polynomial mutation [68]. In SBX, two parents $\mathbf{x}_1$ and $\mathbf{x}_2$ generate two offsprings $\mathbf{o}_1$ and $\mathbf{o}_2$ by

$$\begin{cases} \mathbf{o}_{1,i} = 0.5 \times [(1+\beta)\mathbf{x}_{1,i} + (1-\beta)\mathbf{x}_{2,i}] \\ \mathbf{o}_{2,i} = 0.5 \times [(1-\beta)\mathbf{x}_{1,i} + (1+\beta)\mathbf{x}_{2,i}] \end{cases} \tag{11}$$

where i represents the ith element of $\mathbf{x}$, and β is a random variable. For each element, β is determined by the user-defined distribution index γ dynamically in the following way:

$$\beta = \begin{cases} (2 \times rand)^{\frac{1}{\gamma+1}}, & rand \le 0.5 \\ (2 - 2 \times rand)^{\frac{-1}{\gamma+1}}, & \text{otherwise} \end{cases} \tag{12}$$

where *rand* is a random number in the range of [0,1] and γ is set to 20 based on existing research experiences [65]. After SBX, one solution is randomly selected from $\mathbf{o}_1$ and $\mathbf{o}_2$ to generate the offspring by performing polynomial mutation. The polynomial mutation can be expressed as follows:

$$\mathbf{x}'_i = \mathbf{x}_i + \delta\,(upper_i - lower_i) \tag{13}$$

where $upper_i$ and $lower_i$ represent the upper and lower bounds of the ith element, and δ is defined as (14) shown at the bottom of this page, where $\delta_1 = (\mathbf{x}_i - lower_i)/(upper_i - lower_i)$, $\delta_2 = (upper_i - \mathbf{x}_i)/(upper_i - lower_i)$, and η is a user-defined distribution index which is set to 20 according to experiences [65].

Afterward, the offspring $\mathbf{o}$ is added to the population, and the front number, the convergence degree, and the diversity degree of $\mathbf{o}$ are calculated. Last, TSEA selects *NP* solutions from the current *NP*+1 solutions in $\mathbf{X}$ by the environmental selection. The details of image restructuring, mating selection, and environmental selection will be presented in the following parts.

C. Image Restructuring

The restructuring process is illustrated in Fig. 2. Assuming that there exist three kinds of endmembers in the HSI. The input HSI is first clustered into three classes by the K-Means algorithm

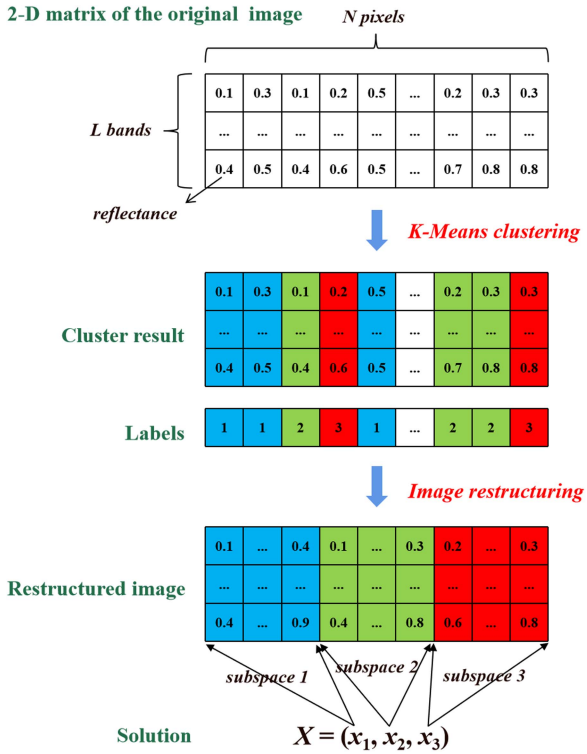


Fig. 2. Image restructuring process.

[69]. Notice that the number of clusters is determined by the number of endmembers, which is priori knowledge. All pixels are reordered according to their clustered labels in a way that pixels of the same cluster are arranged in adjacent columns in the image matrix. For each cluster, the minimum and maximum indices of columns are recorded in the boundary vector *lower* and *upper*. For example, if the HSI contains 100 pixels, where 20, 30, and 50 pixels are clustered into three clusters, respectively. After the image restructuring, the first 20 pixels belong to the first cluster, the next 30 pixels belong to the second cluster and the last 50 pixels belong to the third cluster. The *lower* vector is (1,21,51) and the *upper* vector is (20,50,100). The image restructuring process only changes the order of pixels. No change has been made to the image data.

D. Mating Selection in TSEA

In mating selection, two solutions with high convergence or diversity are picked up each time to generate offspring. Different selection criteria are used at different stages to cooperatively reach the optimal optimization goal. TSEA tends to select the well-converged solutions in the first stage, and the convergence degree and the population diversity are both considered in the second stage. As shown in Algorithm 3, mating selection starts

$$\delta = \begin{cases} [2 \times rand + (1 - 2 \times rand)\,(1 - \delta_1)]^{\frac{1}{\eta+1}}, & rand \le 0.5 \\ 1 - \left[2\,(1 - rand) + 2\,(rand - 0.5)\,(1 - \delta_2)^{\eta+1}\right]^{\frac{1}{\eta+1}}, & \text{otherwise} \end{cases} \tag{14}$$

Algorithm 3: Mating Selection.

Input: $\mathbf{X}$ (current population), *stage* (current stage), *Front* (front number of each solution), *Con* (convergence degree of each solution), *Div* (diversity degree of each solution)
Output: $parent_1$, $parent_2$ (two parents)
1 Randomly select four different solutions $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$, $\mathbf{x}_4$ from $\mathbf{X}$;
2 **if** $stage == 1$
3 **if** $Front(\mathbf{x}_1) < Front(\mathbf{x}_2)$ **then** $parent_1 = \mathbf{x}_1$;
4 **else if** $Front(\mathbf{x}_1) > Front(\mathbf{x}_2)$ **then** $parent_1 = \mathbf{x}_2$;
5 **else if** $Con(\mathbf{x}_1) < Con(\mathbf{x}_2)$ **then** $parent_1 = \mathbf{x}_1$;
6 **else** $parent_1 = \mathbf{x}_2$;
7 **end if**
8 **if** $Front(\mathbf{x}_3) < Front(\mathbf{x}_4)$ **then** $parent_2 = \mathbf{x}_3$;
9 **else if** $Front(\mathbf{x}_3) > Front(\mathbf{x}_4)$ **then** $parent_2 = \mathbf{x}_4$;
10 **else if** $Con(\mathbf{x}_3) < Con(\mathbf{x}_4)$ **then** $parent_2 = \mathbf{x}_3$;
11 **else** $parent_2 = \mathbf{x}_4$;
12 **end if**
13 **else** // $stage == 2$
14 **if** $Con(\mathbf{x}_1) < Con(\mathbf{x}_2)$ **then** $parent_1 = \mathbf{x}_1$;
15 **else** $parent_1 = \mathbf{x}_2$;
16 **end if**
17 **if** $Div(\mathbf{x}_3) > Div(\mathbf{x}_4)$ **then** $parent_2 = \mathbf{x}_3$;
18 **else** $parent_2 = \mathbf{x}_4$;
19 **end if**
20 **end if**

with the selection of four solutions $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$, and $\mathbf{x}_4$ from $\mathbf{X}$ without replacement. If $stage = 1$, the solution with a smaller front number in $\mathbf{x}_1$ and $\mathbf{x}_2$ is selected as $parent_1$. If $\mathbf{x}_1$ and $\mathbf{x}_2$ have the same front number, the one with better convergence degree will be selected. The same criterion is adopted for the selection of $parent_2$ from $\mathbf{x}_3$ and $\mathbf{x}_4$. If $stage = 2$, the one with a better convergence degree is picked up from $\mathbf{x}_1$ and $\mathbf{x}_2$. Similarly, as for $\mathbf{x}_3$ and $\mathbf{x}_4$, the one with a better diversity degree is selected.

E. Environmental Selection in TSEA

After generating offspring $\mathbf{o}$, the new offspring is added to the population, and one solution should be eliminated from the population through environmental selection. Algorithm 4 displays the environmental selection strategy. In the first stage, the population is not well converged and contains dominated solutions. In this situation, the solution with the worst convergence degree is eliminated. Specifically, we find the solutions with the largest front number in the population, then remove the one with the smallest convergence degree. In the second stage, all the solutions are nondominated and the population is well-converged. In this situation, the population only needs to be fine-tuned to improve diversity. As shown in line 6 to line 11 in Algorithm 4, the nearest neighbor of $\mathbf{o}$ named $\mathbf{w}$ in the objective space is selected. If $\mathbf{o}$ is better than $\mathbf{w}$ in terms of both convergence and diversity, then $\mathbf{w}$ is replaced by $\mathbf{o}$. Otherwise, the new offspring $\mathbf{o}$ will not be reserved.

Algorithm 4: Environmental Selection.

Input: $\mathbf{X}$ (current population), $\mathbf{o}$ (new offspring), *stage* (current stage), *Front* (front number of each solution), *Con* (convergence degree of each solution), *Div* (diversity degree of each solution)
Output: $\mathbf{X}$ (updated population)
1 **if** $stage == 1$
2 $\mathbf{X} = \{\mathbf{x} \in \mathbf{X} \mid Front(\mathbf{x}) = \max_{\mathbf{y} \in \mathbf{X}} Front(\mathbf{y})\}$
3 $\mathbf{w} = \text{argmax}_{\mathbf{x} \in \mathbf{X}} Con(\mathbf{x})$;
4 $\mathbf{X} = \mathbf{X} \backslash \{\mathbf{w}\}$;
5 **else** // $stage == 2$
6 $\mathbf{w} = \text{argmin}_{\mathbf{x} \in \mathbf{X}} \| \mathbf{f}'(\mathbf{o}) - \mathbf{f}'(\mathbf{x}) \|$;
7 **if** $Con(\mathbf{o}) \leq Con(\mathbf{w})$ **and** $Div(\mathbf{o}) \geq Div(\mathbf{w})$;
8 $\mathbf{X} = \mathbf{X} \backslash \{\mathbf{w}\}$;
9 **else**
10 $\mathbf{X} = \mathbf{X} \backslash \{\mathbf{o}\}$;
11 **end if**
12 **end if**

TABLE I
COMPUTATIONAL COMPLEXITY OF TSEA AND THE COMPARISON ALGORITHMS

Algorithm	Computational complexity
DSPO	$O(num\_pop \cdot LPN)$
IMODPSO	$O(num\_pop^2 + num\_pop(P! + LPN))$
(μ+λ)-MODE	$O(num\_pop^2 + num\_pop(P! + LPN))$
GL-EA	$O(num\_pop^2 + num\_pop(P! + LPN))$
TSEA	$O(num\_pop^3 + num\_pop(P! + LPN))$

F. Computational Complexity

Assume that there are N pixels, L bands, and P kinds of endmembers in the image, and the population size is *num_pop*. In one generation (i.e., generating *num_pop* offsprings), the computational complexity of generating offsprings is $O(num\_pop^3)$, and the computational complexity of computing the objective functions is $O(num\_pop(P! + LPN))$. Therefore, the total computational complexity of TSEA is $O(num\_pop^3 + num\_pop(P! + LPN))$. The computational complexity of DPSO, IMODPSO, (μ+λ)-MODE, and GL-EA are shown in Table I.

IV. EXPERIMENTS AND ANALYSIS

A. Parameter Settings

Four state-of-the-art EA-based EE algorithms, DPSO [32], IMODPSO [54], (μ+λ)-MODE [56], and GL-EA [53] were chosen as comparison algorithms in the experiments. Except for DPSO which only optimizes the RMSE, other algorithms are MOEA-based EE algorithms in which the volume inverse and the RMSE are minimized simultaneously. For all algorithms, the population size is set to 20, and the number of function evaluations is set to 6000 for all datasets. For (μ+λ)-MODE, the scaling factor pool F_{pool} is set to [0.6, 0.8, 1.0] and the crossover control parameter pool CR_{pool} is set to [0.1, 0.2, 1.0]. For GL-EA, the range of perturbation is set to 6 and the parameter

Fig. 3. Berlin dataset.

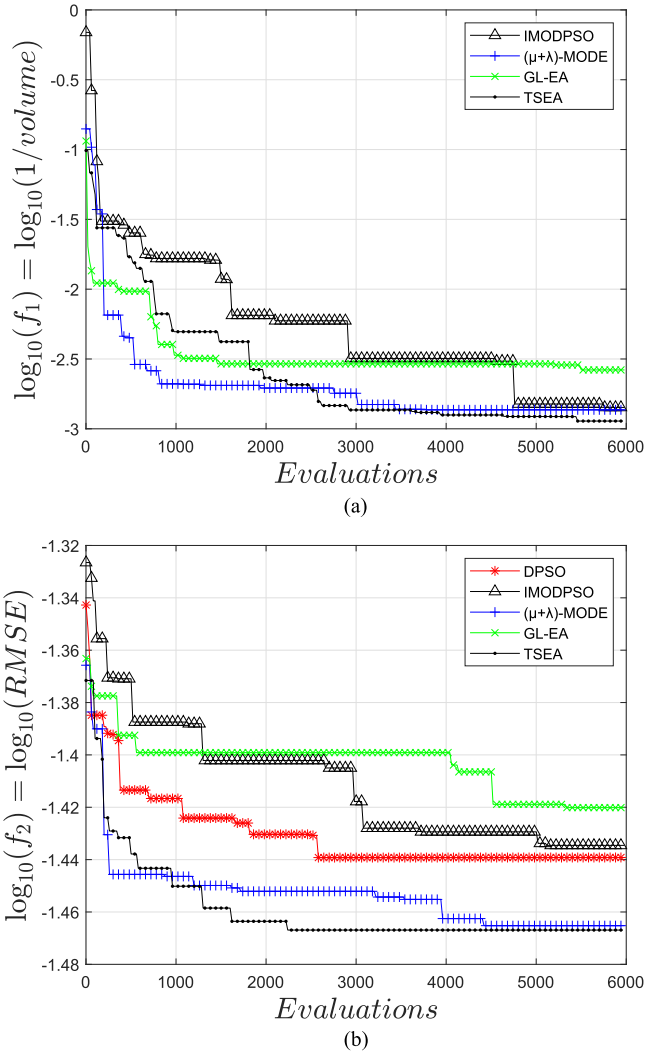


Fig. 4. Minimum objective function values during generations for the Berlin image. (a) Logarithm of volume inverse. (b) Logarithm of RMSE.

for dividing the global search and the local search is set to 0.8. For TSEA, the crossover rate and the distribution index of SBX are set to 1 and 20. The mutation rate and the distribution index of the polynomial mutation are set to 0.05 and 20. The scaling factor ε in the calculation of *Div* is set to 10^{-6}. Each algorithm is executed for ten independent runs and the best result is selected for comparison.

B. Experiments on the Berlin Dataset

The Berlin dataset is a subset of HyMap01 image product from the Berlin-Urban-Gradient dataset 2009 [70]. The image has 526×336 pixels covering 111 bands ranging from 0.4544 to 2.4465 μm after removing noisy bands. The spatial resolution of the image is 3.6 m. According to the reference spectral library provided by the hyperspectral product, there exist six types of endmembers including roof, pavement, low vegetation, tree, soil and other in the image. The class "other" consists of red sand, rail track, artificial turf and some other rare ground objects. The true color image composed of R-band 14, G-band 7, and B-band 1 is shown in Fig. 3.

Fig. 4 displays the convergence curves of the five EE algorithms, where the x-axis represents the evaluation times and the y-axis stands for the minimum logarithm of objective function values in the population. DPSO is not displayed in Fig. 4(a) because it is a single objective algorithm which only minimizes the RMSE. As shown in Fig. 4, TSEA reached the minimum objective function values for both the volume inverse and RMSE. During the optimization process, TSEA can escape from the local optimal solution and achieve a better solution using less evaluations than other methods, indicating the effectiveness and efficiency of the proposed subspace searching and two-stage optimization strategies.

The two objective function values of the output solutions obtained by the five algorithms are presented in Fig. 5. The number of displayed solutions is less than the population size because only the nondominated solutions in the population are displayed for the MOEA-based algorithms. We can see that the minimum values of f_1 and f_2 are both produced by TSEA. The solutions obtained by TSEA dominate the solutions of DPSO, IMODPSO, $(\mu+\lambda)$-MODE, and GL-EA, which means TSEA is able to produce high-quality solutions with smaller volume inverse and RMSE values. Table II shows the computational time. The 64-bit 2022b version of MATLAB was implemented on Intel Core i9-11900K. TSEA has a relatively high computation efficiency than other MOEA-based methods.

The extracted endmember spectra by the five algorithms are shown in Fig. 6. The endmember spectra obtained by some algorithms are absent in some classes because these algorithms did not extract all kinds of endmembers in a certain run. Results

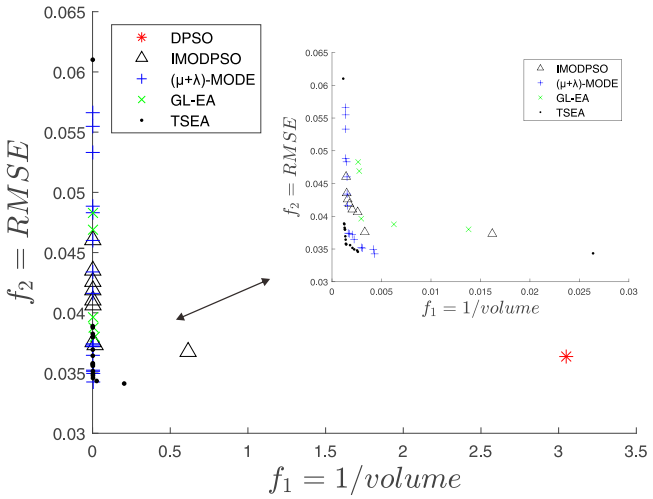


Fig. 5. PF results of the Berlin image.

TABLE II
COMPUTATIONAL TIME OF TSEA AND THE COMPARISON ALGORITHMS ON THE BERLIN DATASET

Algorithm	Time (s)
DSPO	4.34E+02
IMODPSO	1.94E+03
(μ+λ)-MODE	2.91E+03
GL-EA	3.71E+03
TSEA	1.36E+03

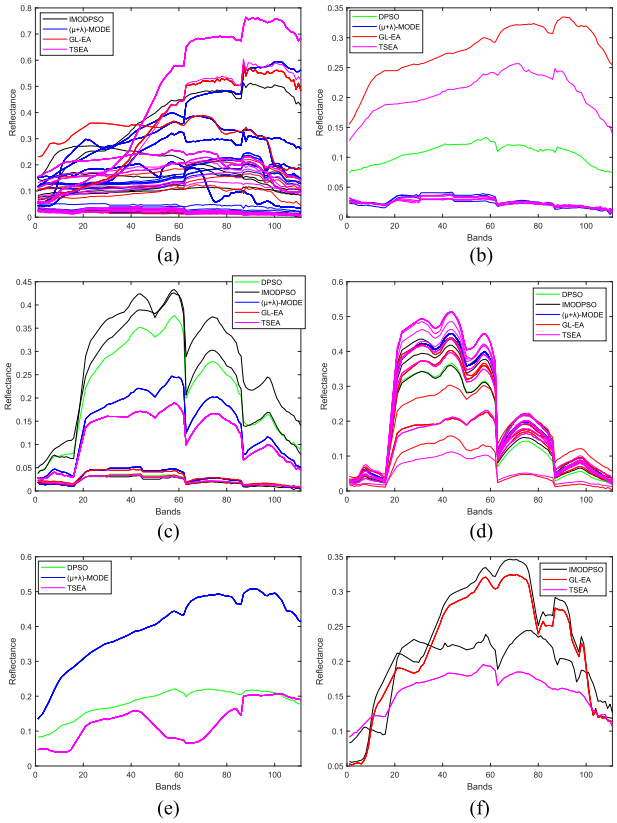


Fig. 6. Endmember spectra obtained by all algorithms on the Berlin image. (a) Roof. (b) Pavement. (c) Low vegetation. (d) Tree. (e) Soil. (f) Other.

TABLE III
SAD COMPARISON BETWEEN THE EXTRACTED ENDMEMBERS AND THE REFERENCE ENDMEMBERS ON THE BERLIN DATASET

Material	DPSO	IMODPSO	(μ+λ)-MODE	GL-EA	TSEA
Roof	**0.0760**	0.0942	0.0984	0.1099	0.0798
Pavement	**0.0509**	0.0661	0.0983	0.0561	0.0719
Low vegetation	0.0758	**0.0698**	0.0924	0.0842	0.0711
Tree	**0.0205**	0.0315	0.0274	0.0296	0.0279
Soil	0.0424	0.0654	**0.0344**	0.0518	0.0608
Other	0.1373	0.0905	0.0870	0.1068	**0.0764**
Mean	0.0672	0.0696	0.0730	0.0731	**0.0647**

The bold entities represent the minimum values.

Fig. 7. Urban dataset.

showed that only TSEA extracted six kinds of endmembers completely while other algorithms only obtained partial kinds of endmembers. It is worth noting that the four MOEA-based algorithms, i.e., IMODPSO, $(\mu+\lambda)$-MODE, GL-EA, and TSEA, were able to extract multiple spectra for one kind of endmember, whereas the single optimization algorithm DPSO only extracted one spectrum for each endmember. This is because the MOEA-based algorithms can obtain a series of nondominated solutions. These nondominated solutions have either small volume inverse or small RMSE which can provide users with more complete endmembers.

Table III displays the SAD values between the extracted endmembers and the reference endmembers obtained by the five algorithms. DPSO had the minimum SAD on roof, pavement, and tree. The smallest SAD values on low vegetation, soil, and other were produced by IMODPSO, $(\mu+\lambda)$-MODE, and TSEA, respectively. The mean SAD of TSEA was the smallest.

C. Experiments on the Urban Dataset

The Urban dataset, which is widely used in HU, contains 303 $\times$ 303 pixels with 162 bands ranging from 0.4–2.5 nm. Six kinds of endmembers exist in the image. They are asphalt road, grass, tree, roof, metal, and dirt. The RGB image of Urban is displayed

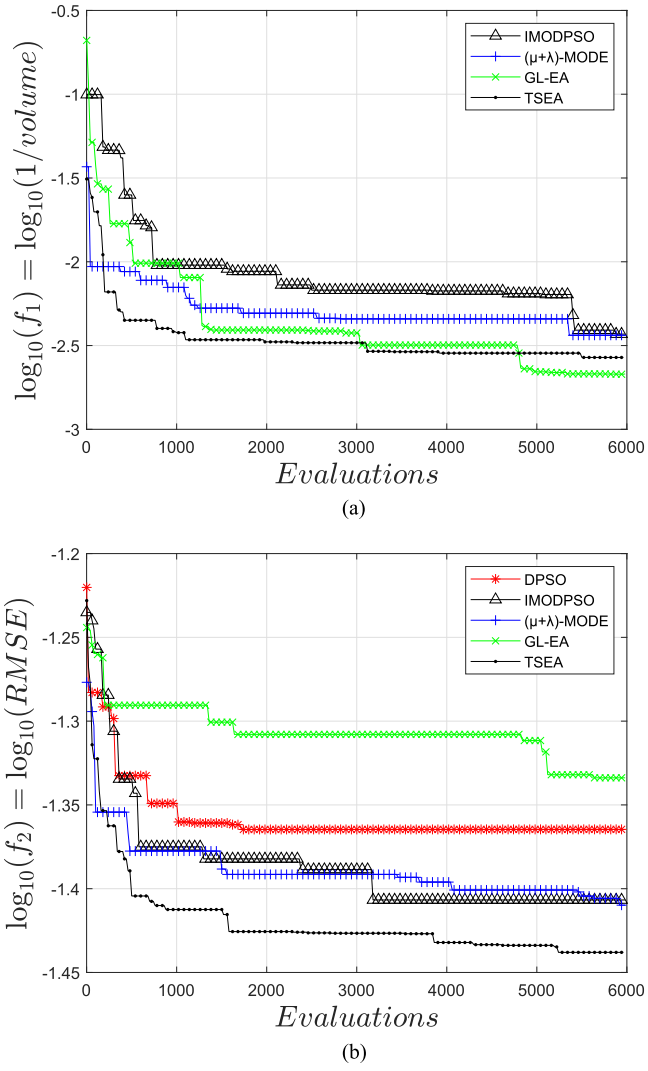


Fig. 8. Minimum objective function value at evaluations for the Urban image. (a) Logarithm of volume inverse. (b) Logarithm of RMSE.

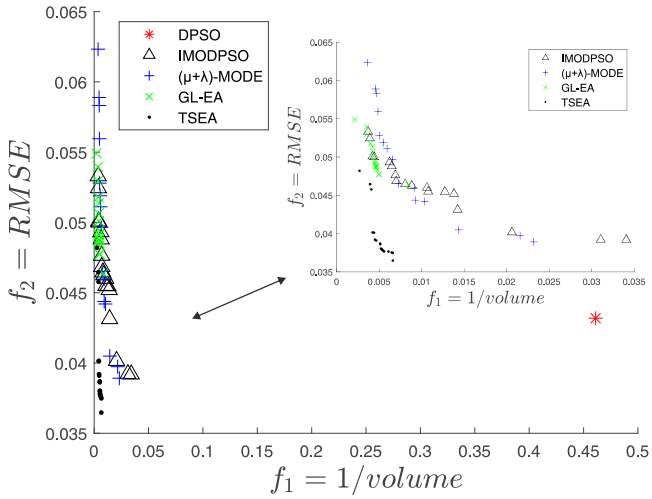


Fig. 9. PF results of the urban image.

TABLE IV
COMPUTATIONAL TIME OF TSEA AND THE COMPARISON ALGORITHMS ON THE URBAN DATASET

Algorithm	Time (s)
DSPO	1.55E+02
IMODPSO	1.38E+03
(μ+λ)-MODE	2.10E+03
GL-EA	2.52E+03
TSEA	1.56E+03

TABLE V
SAD COMPARISON BETWEEN THE EXTRACTED ENDMEMBERS AND THE REFERENCE ENDMEMBERS ON THE URBAN DATASET

Material	DPSO	IMODPSO	(μ+λ)-MODE	GL-EA	TSEA
Asphalt Road	0.0885	0.1021	0.0984	0.0959	**0.0780**
Grass	0.1201	**0.1088**	0.1180	0.1095	0.1264
Tree	0.1369	0.1433	0.1357	0.1116	**0.1028**
Roof	**0.1286**	0.1969	0.1311	0.1785	0.1630
Metal	0.2529	0.1507	0.1851	0.2574	**0.1241**
Dirt	**0.0648**	0.0770	0.0934	0.0920	0.1227
Mean	0.1320	0.1298	0.1269	0.1408	**0.1195**

The bold entities represent the minimum values.

in Fig. 7 and the red, green, and blue bands are set to bands 53, 35, and 10, respectively.

The convergence curves of the algorithms are displayed in Fig. 8. In terms of volume inverse, GL-EA achieved the fastest convergence speed among all the algorithms. However, the performance of GL-EA was the worst in terms of RMSE. The minimum RMSE value obtained by TSEA when evaluation times reached 1000 was smaller than those obtained by other algorithms when evaluation times reached 6000.

Fig. 9 displays the output results of objective function values for all the algorithms. It can be seen that GL-EA produced the solution with the minimum volume inverse and TSEA produced the solution with the minimum RMSE. The minimum value of volume inverse obtained by TSEA was very close to the minimum value obtained by other algorithms, whereas the minimum value of RMSE obtained by TSEA was much smaller than the minimum value obtained by other algorithms, which indicated that TSEA has better optimization ability considering the optimization of both objective functions. Solutions obtained by TSEA dominated most solutions of the other algorithms. GL-EA obtained one nondominated solution with the minimum volume inverse. From the computational time shown in Table IV, TSEA was more efficient than (μ+λ)-MODE and GL-EA.

The extracted endmember spectra of the five algorithms are presented in Fig. 10. IMODPSO, (μ+λ)-MODE and TSEA extracted the six types of endmember spectra completely. DPSO and GL-EA only extracted five kinds of endmember spectra and the spectrum of metal was not successfully extracted.

The SAD values between the extracted endmembers and the reference endmembers obtained by the five algorithms from the Urban image are presented in Table V. It can be seen that DPSO had the minimum SAD on roof and dirt. IMODPSO had the smallest SAD on grass. TSEA performed best on asphalt road,

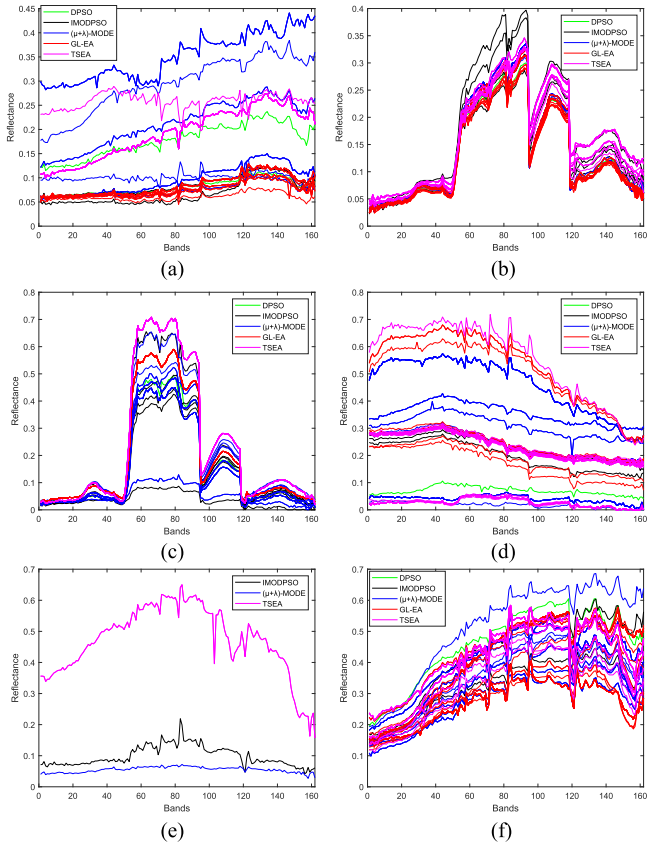


Fig. 10. Endmember spectra obtained by all algorithms on the Urban image. (a) Asphalt road. (b) Grass. (c) Tree. (d) Roof (e) Metal. (f) Dirt.

Fig. 11. Augsburg dataset.

tree, and metal. The SAD value of TSEA on metal was much smaller than that of other algorithms. The mean SAD of TSEA was the lowest.

D. Experiments on the Augsburg Dataset

The Augsburg dataset was acquired by using the airborne imaging spectrometer system HySpex, which was operated by the Remote Sensing Technology Institute (IMF) of the German

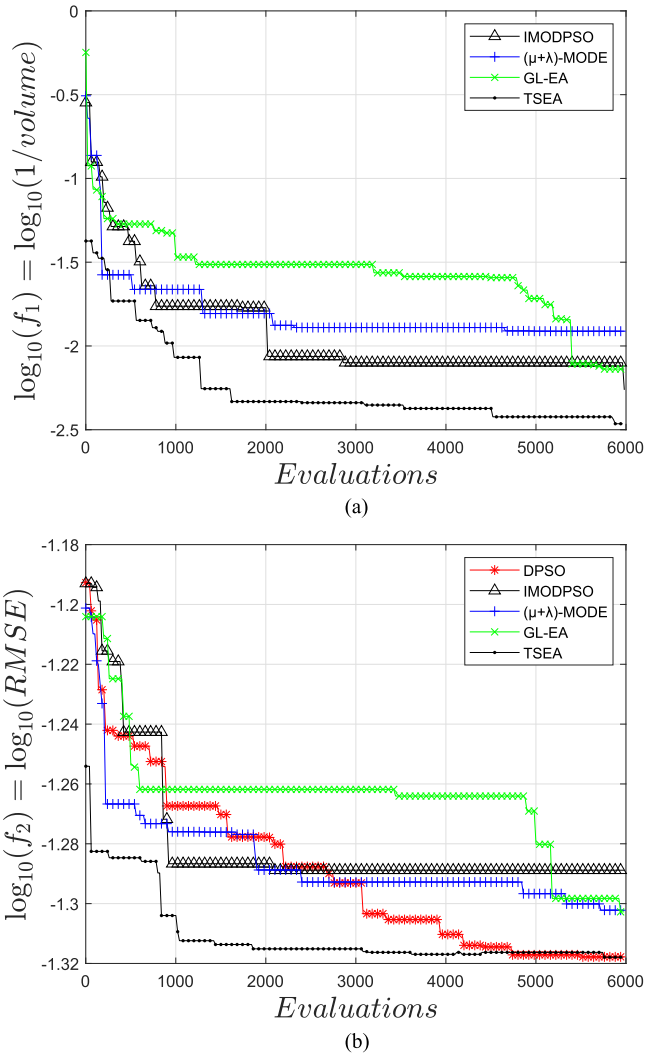


Fig. 12. Minimum objective function value at evaluations for the Augsburg image. (a) Logarithm of volume inverse. (b) Logarithm of RMSE.

Aerospace Center (DLR) [71]. The image used in this research is a 200 × 200 subset of the original data. There are 368 bands distributing from 0.416 to 2.498 μm. Six kinds of endmembers exist in the image. They are pavement, low vegetation, soil, tree, roof, and water. Fig. 11 shows the true color image of Augsburg consisting of R-band 77, G-band 43, and B-band 9.

The convergence curves displayed in Fig. 12 indicate that TSEA converges faster than other algorithms. In terms of volume inverse, the objective function value obtained by TSEA at evaluations 1500 was smaller than the objective function values obtained by other algorithms at evaluations of 6000. In terms of RMSE, DPSO, and TSEA achieved the same RMSE value in the end, and the RMSE value obtained by TSEA at evaluations of 1000 was smaller than the RMSE values obtained by the other three MOEA-based algorithms at evaluations of 6000.

The PF comparison results in Fig. 13 show that solutions produced by the four MOEA-based methods have a smaller volume inverse than DPSO. Solutions of TSEA dominated the other four algorithms. The solution with the smallest volume inverse was obtained by TSEA. Considering the RMSE, solutions

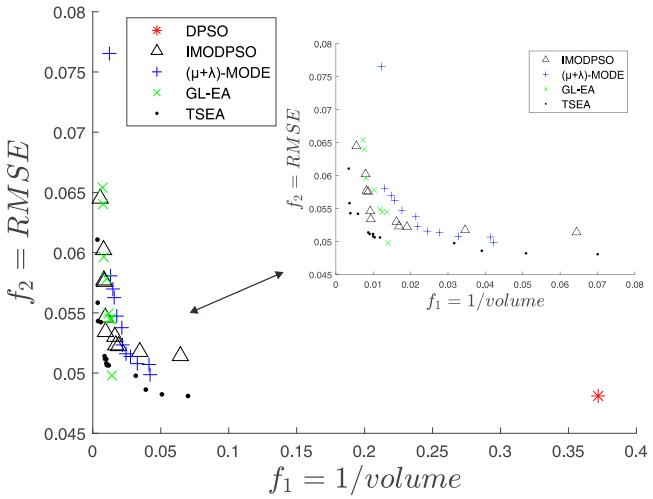


Fig. 13. PF results of the Augsburg image.

TABLE VI
COMPUTATIONAL TIME OF TSEA AND THE COMPARISON ALGORITHMS ON THE AUGSBURG DATASET

Algorithm	Time (s)
DSPO	3.97E+02
IMODPSO	4.02E+02
(μ+λ)-MODE	6.98E+02
GL-EA	2.51E+03
TSEA	3.53E+02

TABLE VII
SAD COMPARISON BETWEEN THE EXTRACTED ENDMEMBERS AND THE REFERENCE ENDMEMBERS ON THE AUGSBURG DATASET

Material	DPSO	IMODPSO	(μ+λ)-MODE	GL-EA	TSEA
Pavement	0.1930	0.2381	0.2064	0.2000	**0.1362**
Low vegetation	**0.0523**	0.0675	0.0731	0.0724	0.0687
Soil	0.3132	0.2125	0.2625	0.1813	**0.0887**
Tree	0.0996	0.0881	0.1441	0.0877	**0.0852**
Roof	0.0947	**0.0824**	0.1123	0.0856	0.1075
Water	**0.2080**	0.2803	0.3903	0.3218	0.4031
Mean	0.1601	0.1615	0.1981	0.1581	**0.1482**

The bold entities represent the minimum values.

produced by DPSO and TSEA had the smallest value. As shown in Table VI, in terms of efficiency, TSEA spent the least time compared with other algorithms.

Fig. 14 shows the endmember spectra obtained by all algorithms for the Augsburg image. GL-EA and TSEA extracted six kinds of endmembers completely. Other algorithms only extracted five kinds of endmembers.

As displayed in Table VII, the SAD of TSEA on pavement and soil were smaller than the other algorithms significantly. The minimum SAD on low vegetation and roof were obtained by DPSO and IMODPSO. In terms of mean SAD, TSEA had the best performance.

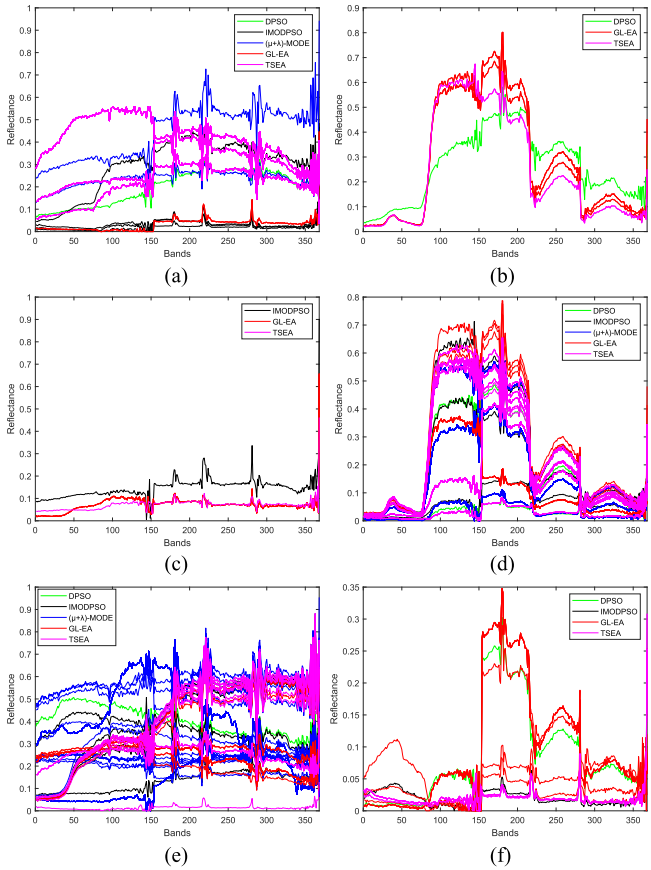


Fig. 14. Endmember spectra obtained by all algorithms on the Augsburg image. (a) Pavement. (b) Low vegetation. (c) Soil. (d) Tree (e) Roof. (f) Water.

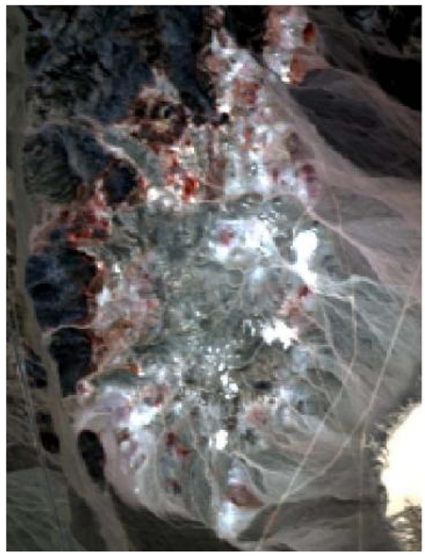
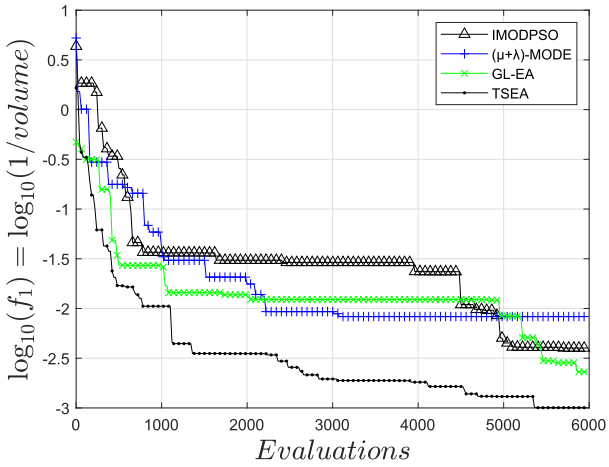


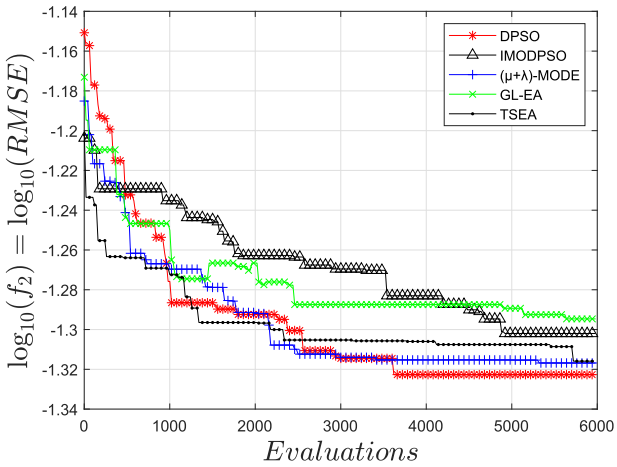
Fig. 15. Cuprite dataset.

E. Experiments on the Cuprite Dataset

The Cuprite dataset contains 224 channels ranging from 370 to 2480 μm with a spatial resolution of 20 m. After removing the noisy and water absorption bands, there are 188 bands remained. The image is displayed in Fig. 15. The spectra of



(a)



(b)

Fig. 16. Minimum objective function value at evaluations for the cuprite image. (a) Logarithm of volume inverse. (b) Logarithm of RMSE.

six representative minerals are selected from the USGS spectral library as reference spectra to validate the results of EE including alunite, andradite, dumortierite, kaolinite, montmorillonite, and nontronite.

The convergence curves of the five algorithms are presented in Fig. 16. Considering the volume inverse, the objective function value of TSEA was smaller than other algorithms almost throughout the optimization process. In terms of RMSE, DPSO had the fastest convergence speed and obtained the minimum value after 6000 evaluations. The final RMSE value of TSEA was close to DPSO and was much smaller than that of IMODPSO and GL-EA.

Fig. 17 displays the solutions obtained by all algorithms. We can see that TSEA obtained the minimum volume inverse, whereas DPSO obtained the minimum RMSE. The minimum volume inverse of TSEA is much smaller than other algorithms. The solutions generated by TSEA dominated most of the remaining solutions from other algorithms. As shown in Table VIII, in terms of efficiency, TSEA spent the least time among the five algorithms.

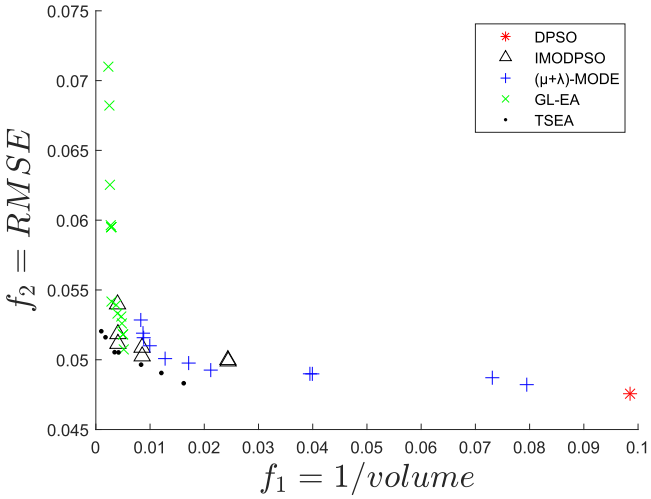


Fig. 17. PF results of the cuprite image.

TABLE VIII
COMPUTATIONAL TIME OF TSEA AND THE COMPARISON ALGORITHMS ON THE CUPRITE DATASET

Algorithm	Time (s)
DSPO	3.11E+02
IMODPSO	4.41E+02
(μ+λ)-MODE	5.78E+02
GL-EA	5.97E+02
TSEA	2.51E+02

TABLE IX
SAD COMPARISON BETWEEN THE EXTRACTED ENDMEMBERS AND THE REFERENCE ENDMEMBERS ON THE CUPRITE DATASET

Material	DPSO	IMODPSO	(μ+λ)-MODE	GL-EA	TSEA
Alunite	**0.1017**	0.1049	0.1043	0.1069	0.1053
Andradite	0.0738	**0.0729**	0.0737	0.0754	**0.0729**
Dumortierite	**0.0750**	0.0895	0.1161	0.0898	0.0912
Kaolinite	0.0715	0.0698	0.0722	0.0729	**0.0646**
Montmorillonite	**0.0637**	0.0656	0.0644	0.0659	0.0647
Nontronite	0.0777	0.0766	0.0807	0.0794	**0.0763**
Mean	**0.0772**	0.0799	0.0852	0.0817	0.0792

The bold entities represent the minimum values.

The endmember spectra extracted by the five algorithms are presented in Fig. 18. All of the algorithms only extracted five kinds of endmember spectra. The spectrum of dumortierite was missed by DPSO, and the remaining four MOEA-based algorithms failed to extract the spectrum of alunite.

The SAD values between the extracted endmembers and the reference endmembers obtained by the five algorithms from the Cuprite dataset are displayed in Table IX. DPSO had the minimum SAD on alunite, dumortierite, and montmorillonite. TSEA obtained the smallest SAD on andradite, kaolinite, and nontronite. In terms of mean SAD, the performance of TSEA was better than IMODPSO, $(\mu+\lambda)$-MODE, and GL-EA.

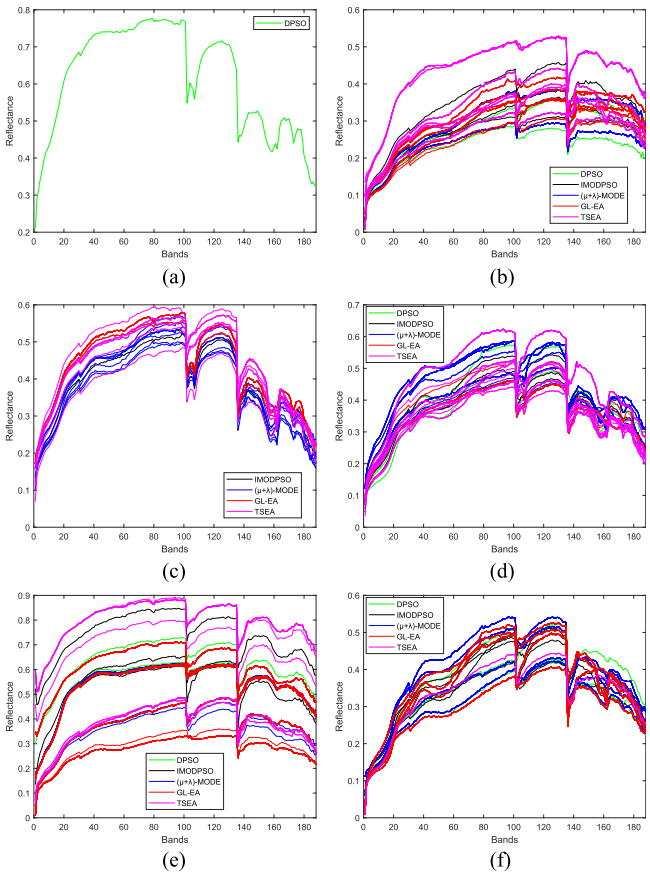


Fig. 18. Endmember spectra obtained by all algorithms on the cuprite image. (a) Alunite. (b) Andradite. (c) Dumortierite. (d) Kaolinite. (e) Montmorillonite. (f) Nontronite.

F. Ablation Experiments

Several ablation experiments were conducted to verify the effectiveness of each component of TSEA. The four different combinations of components applied in the experiments are described as follows.

1) *TSEA/Cluster:* The image restructuring step is removed from the proposed TSEA algorithm.
2) *TSEA/Stage1:* The image restructuring step is used. The first stage is removed from the two-stage evolutionary algorithm, and the mating and environmental selection strategies in the second stage are adopted throughout the whole optimization process.
3) *TSEA/Stage2:* The image restructuring step is used. The second stage is removed from the two-stage evolutionary algorithm, and the mating and environmental selection strategies in the first stage are adopted throughout the whole optimization process.
4) *TSEA:* The proposed TSEA algorithm, where the image restructuring strategy and the two-stage optimization strategy are utilized.

The PF results obtained by the above four algorithms from the four HSIs are displayed in Fig. 19. It can be seen from Fig. 19 that the performances of TSEA deteriorated when removing one of the three components. Solutions obtained by TSEA

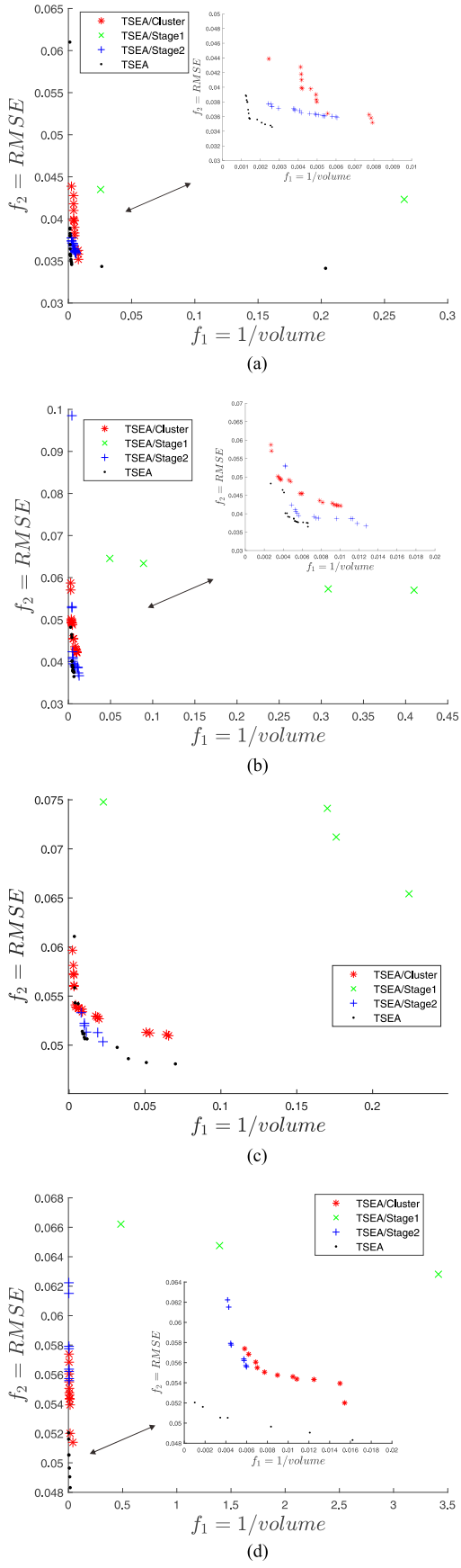


Fig. 19. PF results obtained by the four algorithms from the four HSIs. (a) Berlin. (b) Urban. (c) Augsburg. (d) Cuprite.

dominated most solutions of TSEA/Cluster, TSEA/Stage1, and TSEA/Stage2, indicating the indispensability of each component in TSEA. TSEA/Stage1 performed worst on the four datasets. Solutions generated by TSEA/Stage1 are distributed in the top right region of the figures with poor convergence degree, because the first stage aiming at improving the convergence of the population was absent. However, the distances among the solutions of TSEA/Stage1 were significantly farther than those of other algorithms, suggesting that the second stage can maintain good diversity for the population. From Fig. 19(a), (c), and (d), we can see that TSEA was able to produce solutions with better diversity than TSEA/Stage2. Most solutions of TSEA/Cluster were dominated by solutions produced by TSEA, which proved the effectiveness of the image restructuring strategy.

V. Discussion

Experiments on four real HSIs demonstrated the effectiveness of the proposed TSEA algorithm, and the indispensability of the strategies designed in TSEA was proven in the ablation study. Nevertheless, some issues about TSEA or MOEA-based EE should receive attention. One issue is the influence of clustering on the proposed TSEA algorithm. Clustering errors are inevitably produced when clustering is applied to real HSIs. The proposed algorithm assumes that pure pixels exist in the image. Images that satisfy this assumption are usually with relatively high spatial resolution. Considering that these images are not highly mixed, the majority of the pixels belonging to one certain class have similar spectra with their corresponding endmembers. Therefore, endmembers can be properly clustered into the corresponding subspaces. Although a small number of mixed pixels may be mistakenly clustered, they will not have a significant impact on the final EE result. From the ablation experiments, we can see that the optimization performance of TSEA deteriorates significantly when the clustering-based image restructuring strategy is removed, suggesting the effectiveness of the clustering process. However, if TSEA is applied to images with coarse spatial resolution, such as the Cuprite image with a resolution of 20 m, accurate results cannot be ensured. For example, TSEA did not extract all classes of endmembers from the results of Fig. 18. Another issue is how to adequately make use of the powerful optimization ability of MOEA to address the EE problem. In fact, none of the tested algorithms successfully achieved a complete PF, which means that perhaps some important results were missing. New models could be established to address this problem.

VI. Conclusion

In this article, we proposed a TSEA-based on subspace specified searching for hyperspectral EE. In TSEA, subspaces are built by dividing the original image into several groups to allow each component of the decision variable search in reduced space, so as to accelerate the optimization process. Besides, the solutions are updated by the two-stage optimization strategy to well maintain the convergence speed and the population diversity. Experiments on four HSIs demonstrated the efficiency and the effectiveness of TSEA.

Our future work will mainly focus on two directions: 1) *Normalization of the objective functions in EE*: The scale of the two commonly used objective functions for EE, namely, the volume inverse and the reconstruction error, differ significantly. In the optimization process, the volume inverse of one solution is often much smaller than the reconstruction error, which makes it hard to quantitatively evaluate the population's convergence and diversity degree. Conventional normalization methods cannot be applied to the two objective functions since the upper bounds of them are difficult to calculate. A novel normalization method is necessary to adjust the two objective functions to the same scale. 2) *Automatic estimation of the number of endmembers*: TSEA requires a user-defined number of endmembers. An improper number often had bad effect on the optimization results. In the future, multitask evolutionary methods can be considered to enable solutions contain different number of endmembers, and corresponding evaluation metrics are needed to adaptively choose the best results.

References

[1] R. J. Murphy and S. T. Monteiro, "Mapping the distribution of ferric iron minerals on a vertical mine face using derivative analysis of hyperspectral imagery (430–970 nm)," *ISPRS J. Photogramm. Remote Sens.*, vol. 75, pp. 29–39, 2013.

[2] C. M. Gevaert, J. Suomalainen, J. Tang, and L. Kooistra, "Generation of spectral–temporal response surfaces by combining multispectral satellite and hyperspectral UAV imagery for precision agriculture applications," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 3140–3146, Jun. 2015.

[3] L. Chen, J. Liu, S. Sun, W. Chen, B. Du, and R. Liu, "An iterative GLRT for hyperspectral target detection based on spectral similarity and spatial connectivity characteristics," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5505811.

[4] L. Ren, D. Hong, L. Gao, X. Sun, M. Huang, and J. Chanussot, "Orthogonal subspace unmixing to address spectral variability for hyperspectral image," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5501713.

[5] Z. Han, D. Hong, L. Gao, J. Yao, B. Zhang, and J. Chanussot, "Multimodal hyperspectral unmixing: Insights from attention networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5524913.

[6] N. Keshava and J. F. Mustard, "Spectral unmixing," *IEEE Signal Process. Mag.*, vol. 19, no. 1, pp. 44–57, Jan. 2002.

[7] J. M. Bioucas-Dias et al., "Hyperspectral unmixing overview: Geometrical, statistical, and sparse regression-based approaches," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 354–379, Apr. 2012.

[8] L. Gao, Z. Wang, L. Zhuang, H. Yu, B. Zhang, and J. Chanussot, "Using low-rank representation of abundance maps and nonnegative tensor factorization for hyperspectral nonlinear unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5504017.

[9] X. R. Feng, H. C. Li, R. Wang, Q. Du, X. Jia, and A. Plaza, "Hyperspectral unmixing based on nonnegative matrix factorization: A comprehensive review," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 4414–4436, 2022.

[10] L. Gao, Z. Han, D. Hong, B. Zhang, and J. Chanussot, "CyCU-Net: Cycle-consistency unmixing network by learning cascaded autoencoders," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5503914.

[11] Z. Han, D. Hong, L. Gao, B. Zhang, M. Huang, and J. Chanussot, "AutoNAS: Automatic neural architecture search for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5532214.

[12] R. A. Borsoi, T. Imbiriba, and P. Closas, "Dynamical hyperspectral unmixing with variational recurrent neural networks," *IEEE Trans. Image Process.*, vol. 32, pp. 2279–2294, 2023.

[13] R. A. Borsoi, T. Imbiriba, and J. C. M. Bermudez, "Deep generative endmember modeling: An application to unsupervised spectral unmixing," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 374–384, 2020.

[14] D. Hong et al., "Endmember-guided unmixing network (EGU-Net): A general deep learning framework for self-supervised hyperspectral unmixing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 11, pp. 6518–6531, Nov. 2022.

[15] T. Imbiriba, R. A. Borsoi, and J. C. M. Bermudez, "Low-rank tensor modeling for hyperspectral unmixing accounting for spectral variability," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 1833–1842, Mar. 2020.

[16] M. D. Iordache, J. M. Bioucas-Dias, and A. Plaza, "Sparse unmixing of hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 2014–2039, Jun. 2011.

[17] G. Zhang et al., "Spectral variability augmented sparse unmixing of hyperspectral images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5527413.

[18] H. Li, R. Feng, L. Wang, Y. Zhong, and L. Zhang, "Superpixel-based reweighted low-rank and total variation sparse unmixing for hyperspectral remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 1, pp. 629–647, Jan. 2021.

[19] D. A. Roberts, M. Gardner, R. Church, S. Ustin, G. Scheer, and R. O. Green, "Mapping chaparral in the Santa Monica Mountains using multiple endmember spectral mixture models," *Remote Sens. Environ.*, vol. 65, no. 3, pp. 267–279, 1998.

[20] R. A. Borsoi, T. Imbiriba, J. C. M. Bermudez, and C. Richard, "Deep generative models for library augmentation in multiple endmember spectral mixture analysis," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 10, pp. 1831–1835, Oct. 2021.

[21] A. Ertürk, M. D. Iordache, and A. Plaza, "Sparse unmixing with dictionary pruning for hyperspectral change detection," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 1, pp. 321–330, Jan. 2017.

[22] X. Xu, B. Pan, Z. Chen, Z. Shi, and T. Li, "Simultaneously multiobjective sparse unmixing and library pruning for hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 4, pp. 3383–3395, Apr. 2021.

[23] C. I. Chang, C. C. Wu, and C. T. Tsai, "Random N-finder (N-FINDR) endmember extraction algorithms for hyperspectral imagery," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 641–656, Mar. 2011.

[24] J. M. P. Nascimento and J. M. B. Dias, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 898–910, Apr. 2005.

[25] A. Plaza, P. Martínez, R. Pérez, and J. Plaza, "A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 3, pp. 650–663, Mar. 2004.

[26] T. H. Chan, C. Y. Chi, Y. M. Huang, and W. K. Ma, "A convex analysis-based minimum-volume enclosing simplex algorithm for hyperspectral unmixing," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4418–4432, Nov. 2009.

[27] J. Li, A. Agathos, D. Zaharie, J. M. Bioucas-Dias, A. Plaza, and X. Li, "Minimum volume simplex analysis: A fast algorithm for linear hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 9, pp. 5067–5082, Sep. 2015.

[28] S. Zhang, A. Agathos, and J. Li, "Robust minimum volume simplex analysis for hyperspectral unmixing," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6431–6439, Nov. 2017.

[29] X. Xu, J. Li, C. Wu, and A. Plaza, "Regional clustering-based spatial preprocessing for hyperspectral unmixing," *Remote Sens. Environ.*, vol. 204, pp. 333–346, 2018.

[30] G. Martin and A. Plaza, "Spatial-spectral preprocessing prior to endmember identification and unmixing of remotely sensed hyperspectral data," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 380–395, Apr. 2012.

[31] X. Shen, W. Bao, and K. Qu, "Spatial-spectral hyperspectral endmember extraction using a spatial energy prior constrained maximum simplex volume approach," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1347–1361, 2020.

[32] B. Zhang, X. Sun, L. Gao, and L. Yang, "Endmember extraction of hyperspectral remote sensing images based on the discrete particle swarm optimization algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4173–4176, Nov. 2011.

[33] B. Zhang, X. Sun, L. Gao, and L. Yang, "Endmember extraction of hyperspectral remote sensing images based on the ant colony optimization (ACO) algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 7, pp. 2635–2646, Jul. 2011.

[34] R. Liu, L. Zhang, and B. Du, "A novel endmember extraction method for hyperspectral imagery based on quantum-behaved particle swarm optimization," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 4, pp. 1610–1631, Apr. 2017.

[35] J. Li, H. Li, Y. Liu, and M. Gong, "Multi-fidelity evolutionary multitasking optimization for hyperspectral endmember extraction," *Appl. Soft Comput.*, vol. 111, 2021, Art. no. 107713.

[36] F. Cheng, N. Chen, C. Wang, Q. Wang, and B. Du, "A global-to-local evolutionary algorithm for hyperspectral endmember extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, 2023, Art. no. 5504117.

[37] Z. Wang, J. Wei, J. Li, P. Li, and F. Xie, "Evolutionary multiobjective optimization with endmember priori strategy for large-scale hyperspectral sparse unmixing," *Electronics*, vol. 10, no. 17, 2021, Art. no. 2079.

[38] X. Jiang, M. Gong, H. Li, M. Zhang, and J. Li, "A two-phase multiobjective sparse unmixing approach for hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 1, pp. 508–523, Jan. 2018.

[39] Y. Wei, X. Xu, B. Pan, T. Li, and Z. Shi, "A multiobjective group sparse hyperspectral unmixing method with high correlation library," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 15, pp. 7114–7127, 2022.

[40] B. Du, Q. Wei, and R. Liu, "An improved quantum-behaved particle swarm optimization for endmember extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 6003–6017, Aug. 2019.

[41] M. Xu, L. Zhang, B. Du, L. Zhang, Y. Fan, and D. Song, "A mutation operator accelerated quantum-behaved particle swarm optimization algorithm for hyperspectral endmember extraction," *Remote Sens.*, vol. 9, no. 3, 2017, Art. no. 197.

[42] O. W. Tembhurne and D. Shrimankar, "EEA-PSO: Endmember extraction using advance particle swarm optimization," in *Proc. 4th Int. Conf. Recent Adv. Inf. Technol.*, 2018, pp. 1–8.

[43] X. Sun, L. Yang, B. Zhang, L. Gao, and J. Gao, "An endmember extraction method based on artificial bee colony algorithms for hyperspectral remote sensing images," *Remote Sens.*, vol. 7, no. 12, pp. 16363–16383, 2015.

[44] M. Xu, Y. Zhang, Y. Fan, Y. Chen, and D. Song, "Linear spectral mixing model-guided artificial bee colony method for endmember generation," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 12, pp. 2145–2149, Dec. 2020.

[45] Z. Fu, C. M. Pun, H. Gao, and H. Lu, "Endmember extraction of hyperspectral remote sensing images based on an improved discrete artificial bee colony algorithm and genetic algorithm," *Mobile Netw. Appl.*, vol. 25, pp. 1033–1041, 2020.

[46] D. W. R. Soares, G. T. Laureano, and C. G. Camilo Junior, "GAEEII: An optimised genetic algorithm endmember extractor for hyperspectral unmixing," in *Proc. IEEE Congr. Evol. Comput.*, 2019, pp. 2386–2393.

[47] Y. Rezaei, M. R. Mobasheri, M. V. Zoej, and M. E. Schaepman, "Endmember extraction using a combination of orthogonal projection and genetic algorithm," *IEEE Geosci. Remote Sens. Lett.*, vol. 9, no. 2, pp. 161–165, Mar. 2012.

[48] Y. Zhong, L. Zhao, and L. Zhang, "An adaptive differential evolution endmember extraction algorithm for hyperspectral remote sensing imagery," *IEEE Geosci. Remote Sens. Lett.*, vol. 11, no. 6, pp. 1061–1065, Jun. 2014.

[49] L. Gao et al., "Multiple algorithm integration based on ant colony optimization for endmember extraction from hyperspectral imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 6, pp. 2569–2582, Jun. 2015.

[50] W. R. Douglas, G. T. Laureano, and C. G. Camilo, "Comparison of VCA and GAEE algorithms for endmember extraction," in *Proc. IEEE Congr. Evol. Comput.*, 2018, pp. 1–8.

[51] Z. Wang, J. Li, Y. Liu, F. Xie, and P. Li, "An adaptive surrogate-assisted endmember extraction framework based on intelligent optimization algorithms for hyperspectral remote sensing images," *Remote Sens.*, vol. 14, no. 4, 2022, Art. no. 892.

[52] R. Liu, B. Du, and L. Zhang, "Multiobjective optimized endmember extraction for hyperspectral image," *Remote Sens.*, vol. 9, no. 6, 2017, Art. no. 558.

[53] Q. Cheng, B. Du, L. Zhang, and R. Liu, "ANSGA-III: A multiobjective endmember extraction algorithm for hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 2, pp. 700–721, Feb. 2019.

[54] L. Tong, B. Du, R. Liu, and L. Zhang, "An improved multiobjective discrete particle swarm optimization for hyperspectral endmember extraction," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7872–7882, Oct. 2019.

[55] X. Jiang, Y. Zhao, M. Gong, T. Zhan, and M. Zhang, "A vertex-directed evolutionary algorithm for multiobjective endmember estimation," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5535913.

[56] L. Tong, B. Du, R. Liu, L. Zhang, and K. C. Tan, "Hyperspectral endmember extraction by $(\mu+\lambda)$ multiobjective differential evolution algorithm based on ranking multiple mutations," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2352–2364, Mar. 2021.

[57] L. Drumetz, T. R. Meyer, J. Chanussot, A. L. Bertozzi, and C. Jutten, "Hyperspectral image unmixing with endmember bundles and group sparsity inducing mixed norms," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3435–3450, Jul. 2019.
[58] B. Somers, M. Zortea, A. Plaza, and G. P. Asner, "Automated extraction of image-based endmember bundles for improved spectral unmixing," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 5, no. 2, pp. 396–408, Apr. 2012.
[59] R. Liu and X. Zhu, "Endmember bundle extraction based on multiobjective optimization," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 10, pp. 8630–8645, Oct. 2021.
[60] R. Liu, P. Wang, B. Du, and B. Qu, "Endmember bundle extraction based on improved multi-objective particle swarm optimization," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, 2023, Art. no. 5506405.
[61] F. van den Bergh and A. P. Engelbrecht, "A cooperative approach to particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 225–239, Jun. 2004.
[62] C. A. C. Coello, *Evolutionary Algorithms for Solving Multi-Objective Problems*. Berlin, Germany: Springer-Verlag, 2007.
[63] A. A. Green, M. Berman, P. Switzer, and M. D. Craig, "A transformation for ordering multispectral data in terms of image quality with implications for noise removal," *IEEE Trans. Geosci. Remote Sens.*, vol. 26, no. 1, pp. 65–74, Jan. 1988.
[64] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.
[65] Y. Tian, C. He, R. Cheng, and X. Zhang, "A multistage evolutionary algorithm for better diversity preservation in multiobjective optimization," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 51, no. 9, pp. 5880–5894, Sep. 2021.
[66] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," ETH Zürich, Zürich, Switzerland, TIK Report 103, 2001.
[67] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Syst.*, vol. 9, no. 2, pp. 115–148, 1995.
[68] K. Deb and M. Goyal, "A combined genetic adaptive search (GeneAS) for engineering design," *Comput. Sci. Inform.*, vol. 26, pp. 30–45, 1996.
[69] A. K. Jain, "Data clustering: 50 years beyond K-means," *Pattern Recognit. Lett.*, vol. 31, no. 8, pp. 651–666, 2010.
[70] A. Okujeni, S. van der Linden, and P. Hostert, "Berlin-urban-gradient dataset 2009 - An EnMAP preparatory flight campaign," EnMAP Flight Campaigns, GFZ Data Services, Potsdam, Germany, Tech. Rep., 2016, doi: 10.2312/enmap.2016.002.
[71] J. Hu et al., "MDAS: A new multimodal benchmark dataset for remote sensing," *Earth Syst. Sci. Data*, vol. 15, no. 1, pp. 113–131, 2023.

Cong Lei received the B.Sc. degree in geographical information science from Sun Yat-sen University, Guangzhou, China, in 2023. He is currently working toward the Ph.D. degree in cartography and geographical information system with the School of Geography and Planning, Sun Yat-sen University, Guangzhou, China.

His research interests include endmember extraction, hyperspectral unmixing, and remote sensing retrieval.

Rong Liu received the B.S. degree in surveying and mapping engineering from Wuhan University, Wuhan, China, in 2013, and the Ph.D. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing (LIESMARS), Wuhan University, Wuhan, China, in 2018.

She is currently an Associate Professor with the School of Geography and Planning, Sun Yat-sen University, Guangzhou, China. She was a postdoctoral researcher and senior researcher with the Remote Sensing Technology Institute, German Aerospace Center (DLR), Cologne, Germany, and also with Signal Processing in Earth Observation, Technical University of Munich, Munich, Germany, from 2018 to 2021. Her research interests include remote sensing image processing and evolutionary computation.

Ye Tian received the B.Sc. degree in software engineering and the M.Sc. and Ph.D. degrees in computer application technology from Anhui University, Hefei, China, in 2012, 2015, and 2018, respectively.

He is currently an Associate Professor with the School of Computer Science and Technology, Anhui University, Hefei, China. His research interests include evolutionary computation and its applications.

Dr. Tian was the recipient of the 2018, 2021, and 2024 IEEE Transactions on Evolutionary Computation Outstanding Paper Award, the 2020 IEEE Computational Intelligence Magazine Outstanding Paper Award, and the 2022 IEEE Computational Intelligence Society Outstanding PhD Dissertation Award.