Impact of Initialization on Design of Endmember Extraction Algorithms

Antonio Plaza, *Member, IEEE*, and Chein-I Chang, *Senior Member, IEEE*

***Abstract*—Many endmember extraction algorithms (EEAs) have been developed to find endmembers that are assumed to be pure signatures in hyperspectral data. However, two issues arising in EEAs have not been addressed: one is the knowledge of the number of endmembers that must be provided *a priori*, and the other is the initialization of EEAs, where most EEAs initialize their endmember-searching processes by using randomly generated endmembers, which generally result in inconsistent final selected endmembers. Unfortunately, there has been no previous work reported on how to address these two issues, i.e., how to select a set of appropriate initial endmembers and how to determine the number of endmembers p. This paper takes up these two issues and describes two-stage processes to improve EEAs. First, a recently developed concept of virtual dimensionality (VD) is used to determine how many endmembers are needed to be generated for an EEA. Experiments show that the VD is an adequate measure for estimating p. Second, since EEAs are sensitive to initial endmembers, a properly selected set of initial endmembers can make significant improvements on the searching process. In doing so, a new concept of endmember initialization algorithm (EIA) is thus proposed, and four different algorithms are suggested for this purpose. It is surprisingly found that many EIA-generated initial endmembers turn out to be the final desired endmembers. A further objective is to demonstrate that EEAs implemented in conjunction with EIA-generated initial endmembers can significantly reduce the number of endmember replacements as well as the computing time during endmember search.**

***Index Terms*—Automatic target generation process (ATGP), endmember extraction algorithm (EEA), endmember initialization algorithm (EIA), iterative error analysis (IEA), maximin-distance algorithm, unsupervised fully constrained least squares (UFCLS) algorithm.**

I. Introduction

ACCORDING to the definition given in [1], an endmember is an idealized pure signature for a class. Endmember extraction is one of the fundamental and crucial tasks in hyperspectral data exploitation. It has received considerable interest in recent years, with many researchers devoting their effort to developing algorithms for endmember extraction from hyperspectral data [2]. An ultimate goal of an endmember extraction algorithm (EEA) is to find the purest form of each spectrally distinct material on a scene. Unfortunately, very little effort has been reported on the issue of algorithm initialization, which has a significant impact on the final selection of endmembers produced by an EEA [3]. A similar problem is also encountered in vector quantization, where an algorithm may be trapped locally by an inappropriate selection of initial code words [4]. The issue of the initialization that occurs in vector quantization also plays an important role in the EEA design [3] but has been largely overlooked over the past years. This paper aims to investigate, explore, and propose solutions in relation to this neglected topic.

Overall, two primary factors have a significant impact on an EEA. One is the adoption of an appropriate criterion to terminate an EEA, such as application of a stopping rule, which is implemented by either using an error threshold or setting a predetermined number of endmembers to be searched. Since error thresholds are usually dependent on the data properties, it is highly complex to preset an appropriate threshold *a priori* in many practical applications. Additionally, an inappropriate error threshold may result in two consequences: one is early termination of an EEA due to an insufficient number of endmembers, and the other is a slowdown of the algorithm due to expensive computations related to a high number of endmembers, which is more than what it is needed for the search.

Therefore, a more reasonable and feasible approach is to preset an appropriate number of endmembers p for each algorithm to be terminated. In this case, the determination of p becomes critical. If p is set too low, the extracted endmembers may not represent the data well. On the other hand, if p is set too high, some endmembers may not be pure signatures; instead, they may be either mixed or interfering signatures. In order to cope with this issue, a recently developed concept of virtual dimensionality (VD) in [5] and [6] is used to estimate the value of p. Experiments demonstrate that the VD indeed provides a very reliable estimate of p for each EEA.

Once the value of p is determined, a second relevant issue is the selection of a set of appropriate p initial endmembers to speed up the searching process. According to the algorithm design, an optimal algorithm should not depend on the selection of initial conditions, which can only affect the algorithm convergence rate, albeit without altering the final results. This is not true for many algorithms that are implemented in a finite number of runs. For example, Newton's method will never converge if an initial condition is selected incorrectly [7]. As will be demonstrated, an EEA also suffers from similar problems. Since all available EEAs are computationally expensive, the endmember-searching processes implemented by

Manuscript received August 18, 2005; revised May 11, 2006. The work of A. Plaza was supported by the Spanish Ministry of Education and Science under the PR2003-0360 Fellowship.

A. Plaza is with the Department of Computer Science, University of Extremadura, 10071 Caceres, Spain (e-mail: aplaza@unex.es).

C.-I Chang is with the Remote Sensing Signal and Image Processing Laboratory, Department of Computer Science and Electrical Engineering, University of Maryland Baltimore County, Baltimore, MD 21250 USA, and also with the Department of Electrical Engineering, National Chung Hsing University, Taichung 402, Taiwan, R.O.C. (e-mail: cchang@umbc.edu).

Digital Object Identifier 10.1109/TGRS.2006.879538

these algorithms are generally nonexhaustive but rather focused on selective feasible regions. Therefore, an initial set of selected endmembers may ultimately determine the final set of endmembers. In this case, how to select a desired initial set of endmembers becomes a key issue in the EEA design.

In this paper, a new concept of endmember initialization algorithm (EIA) is introduced. More specifically, four EIAs are proposed to produce a set of target pixels that can be used as initial endmembers, namely: 1) automatic target generation process (ATGP) [5], [8]; 2) unsupervised fully constrained least squares (UFCLS) algorithm [9]; 3) iterative error analysis (IEA) algorithm [10]; and 4) maximin-distance algorithm [11]. Two EEAs, namely: 1) pixel purity index (PPI) [12] and 2) N-finder algorithm (N-FINDR) [13], are selected to study their sensitivity to initial conditions. Various scenarios are simulated for performance evaluation and analysis. As shown in conducted experiments, the VD actually provides a good estimate of the number of endmembers to be used for initialization. Additionally, our experiments also demonstrate that many EIA-generated initial endmembers become the final endmembers and that an EEA implemented in conjunction with EIA-generated initial endmembers can significantly reduce endmember replacements as well as computing time during the course of the endmember-searching process.

The remainder of this paper is organized as follows. Section II discusses two major issues that arise during the initialization of EEAs. Section III develops four EIAs that produce a set of target pixels directly from the data, which can be used as initial endmembers. Section IV briefly describes two EEAs (PPI and N-FINDR) to be used for comparative study and analysis. Section V includes the development of computer simulations to demonstrate the significant impact of initialization issues (such as the number of endmembers and the selection of initial endmembers) on the performance of EEAs. Section VI includes similar experiments using real hyperspectral data. Finally, Section VII summarizes the main contributions of this paper and concludes with some remarks.

II. Initialization Issues

From a viewpoint of algorithm design, three major issues determine the performance of an algorithm, namely: 1) initial conditions; 2) stopping criteria; and 3) learning rules. On some occasions, the stopping criteria are closely related to the initial conditions, which are thresholds set to terminate an algorithm. This section aims to probe the first two criteria, initial condition and stopping rule, which have a significant influence on the performance of EEAs. Over the past years, algorithms designed for finding endmembers have mainly focused on the third issue, i.e., learning rules. However, little work has been devoted to algorithm initialization, which can be considered as important as learning rules, as will be described.

A. Initial Conditions Preset to Terminate an EEA

There are generally two initial conditions that can be used to terminate an EEA: one is to preset an error threshold ε to terminate the algorithm. Since the selection of an appropriate ε is usually data dependent, it is generally difficult to do so without prior knowledge of the data [4]. On one end, if the value of ε is set too low, the algorithm may run into a stability problem. Additionally, it may take a long time to converge while producing more endmembers than what is actually needed. On the other end, if the value of ε is too high, the algorithm may terminate earlier than it should. In this case, the set of generated endmembers may be insufficient, and some desired endmembers will not be included. As an alternative, we can preset the number of endmembers required to be generated, i.e., parameter p. In this case, a problem similar to the selection of an error threshold ε occurs, i.e., determination of an appropriate value for p. If the value of p is selected to be too low, then not all desired endmembers will be extracted (specifically, those being "weak" endmembers). On the other hand, if the value of p is selected to be too high, some extracted endmembers may turn out to be unwanted nonpure signatures.

Therefore, a dilemma similar to that encountered in the selection of an appropriate value for ε also arises in the selection of an appropriate value for p. Interestingly, this problem may be taken care of by a new concept called VD, which was recently developed in [5] and [6] and can be used to estimate an appropriate value for p. Despite the fact that the VD may not necessarily correspond to the intrinsic dimensionality (ID) of the data, it has been shown in [5] and [6] that this criterion is practically useful and provides a good estimate of the number of spectrally distinct signatures in a given data set. This is because the VD does not require prior knowledge of the data. Instead, the VD-estimated value is determined by the false alarm probability P_F, which can be directly derived from the data to be processed. It first calculates eigenvalues for both the sample correlation matrix and the sample covariance matrix (referred to as correlation-eigenvalues and covariance-eigenvalues) for each of the spectral bands. If a distinct spectral signature makes a contribution to the eigenvalue-represented signal energy in one spectral band, then its associated correlation-eigenvalue will be greater than its corresponding covariance-eigenvalue in this particular band. Otherwise, the correlation-eigenvalue would be very close to the covariance-eigenvalue, in which case only noise would be present in this particular band.

By applying this concept, the Neyman–Pearson detection theory is introduced to formulate the issue of whether a distinct signature is present or not in each of the spectral bands as a binary hypothesis testing problem, where a so-called Neyman–Pearson detector is generated to serve as a decision maker based on a prescribed P_F (i.e., false alarm probability). In light of this interpretation, the issue of determining an appropriate value for p is further simplified and reduced to a specific value of P_F that is preset by the Neyman–Pearson detector. Compared to the selection of an appropriate error threshold ε without prior knowledge, specifying a fixed value for P_F is more reasonable and realistic for practical applications because the results are determined by the P_F that an EEA is designed to tolerate. As demonstrated in experiments, a suitable empirical choice is $P_F = 10^{-3}$ [5], [6], where the method used to estimate the VD was that by Harsanyi *et al.* [14], which is referred to as the HFC method. Since this method does not require a noise whitening process, weak signal sources

may be obscured by noise while remaining undetected. As an alternative, the HFC method can be modified by including a noise whitening process as preprocessing to remove the second-order statistical correlation. The purpose is that signal sources can be decorrelated from the noise to achieve better signal detection. The resulting method will be referred to as the noise-whitened HFC (NWHFC) [5], [6].

B. Use of an Initial Set of Endmembers for an EEA

After the number of endmembers p has been determined, a follow-up step is to select a set of initial endmembers $\{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \ldots, \mathbf{e}_p^{(0)}\}$ in order to initialize an EEA. It is interesting to note that as far as we know, none of the existing EEAs have discussed the issue of how to select such an initial set. An appropriate selection of initial endmembers can be very beneficial. On some occasions, it is critical to produce correct results as well as to speed up the endmember-searching process. As a result, the design of algorithms for finding the appropriate initial conditions becomes highly desirable [15]. As noted, an EEA often starts with any set of initial endmembers, which are most likely to be randomly generated. If an EEA performs an exhaustive search for p endmembers, then the final results should not be dependent on what set of initial endmembers is selected for initialization. Unfortunately, such an exhaustive search suffers from several drawbacks. First, it is computationally very expensive, in particular, for hyperspectral imagery with large volumes of data. Second, it may take quite long to find a desired set of p endmembers. Finally, it is not feasible in many practical applications. Therefore, an efficient EEA does not conduct a fully exhaustive search but rather focuses on the selections of endmembers from some feasible regions. However, these regions are generally determined by the initial conditions and must be selected very carefully to avoid being trapped in local optimality. Therefore, in order for an EEA to be effective, the selected initial endmembers must be representative and not arbitrary. Surprisingly, this issue has not received attention in the past.

Sections V and VI demonstrate that EEAs are actually sensitive to the selection of endmembers to be used for initialization. This evidence implies that a judicious selection of initial endmembers is crucial for success in the final selection of endmembers. To address this issue, four unsupervised target detection algorithms are developed in the next section. These algorithms can be used to produce a better set of initial endmembers, which can speed up algorithm performance and help generate the best possible final endmembers in terms of signature purity.

III. EIAs

The four EIA algorithms presented in this section generate target pixels in accordance with certain criteria, such as orthogonal subspace projection (OSP)-based ATGP, LSE-based UFCLS and IEA algorithms, and a simple maximin-distance algorithm.

There are some important differences between EIAs and EEAs. When an EEA is implemented, it assumes that the number of endmembers is known in advance and produces p endmembers "simultaneously." So, for a different value of p, an EEA generally produces a different set of endmembers. In other words, for any given number of endmembers p, an EEA must recalculate all the endmembers and cannot take advantage of a set of $p-1$ endmembers previously generated by the same algorithm. In addition, these $p-1$ endmembers do not necessarily constitute a subset of the set of p endmembers generated subsequently. In contrast, an EIA produces a set of target pixels in a "sequential" order. A set of p EIA-generated pixels always includes the set of previously generated $p-1$ target pixels. This feature is highly desirable for an EIA because it saves a great deal of computational time. An EEA generally extracts endmembers in the sense of purest signatures (provided that it is not possible to find completely pure signatures). As a result, the spectral signatures generated by an EEA are not necessarily real-image pixels, whereas an EIA is generally designed to search for the target pixels of interest, even though their spectral signatures may not necessarily be pure signatures.

A. ATGP

The ATGP algorithm was previously developed in [5] and [8] to find potential target pixels that can be used to generate a signature matrix used for an OSP approach in [16]. Let the initial target signature $\mathbf{t}_0$ be the pixel vector with a maximum length. The ATGP begins by applying an orthogonal subspace projector $P_{\mathbf{U}}^{\perp} = \mathbf{I} - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T$ with $\mathbf{U} = [\mathbf{t}_0]$ to every pixel vector $\mathbf{r}$ in the data. It then finds a target signature, which is denoted by $\mathbf{t}_1$, with the maximum absolute projection in $\langle \mathbf{t}_0 \rangle^{\perp}$ produced by $P_{\mathbf{t}_0}^{\perp}$, which is the space orthogonal to the space linearly spanned by $\mathbf{t}_0$. A second target signature $\mathbf{t}_2$ can then be found by applying another orthogonal subspace projector $P_{[\mathbf{t}_0\mathbf{t}_1]}^{\perp}$ with $\mathbf{U} = [\mathbf{t}_0\ \mathbf{t}_1]$ to the original image, where the target signature that has the maximum orthogonal projection in $\langle \mathbf{t}_0, \mathbf{t}_1 \rangle^{\perp}$ is selected as $\mathbf{t}_2$. The preceding procedure is repeated until a set of target pixels $\{\mathbf{t}_0, \mathbf{t}_1, \ldots, \mathbf{t}_{p-1}\} = \mathbf{t}_0 \cup \mathbf{U}_{p-1}$ is extracted.

B. UFCLS

The UFCLS algorithm takes advantage of the fully constrained (i.e., abundance sum-to-one and nonnegativity constraints) least squares (FCLS) method developed for unsupervised linear spectral unmixing [5], [9]. It first finds the pixel vector with the maximum length and selects it as a desired initial target denoted by $\mathbf{t}_0$. It then assumes that all other pixels in the image scene are pure pixels made up of $\mathbf{t}_0$ with 100% abundance. Of course, this is not generally true. Thus, it subsequently finds a pixel that has the largest least squares error (LSE) from $\mathbf{t}_0$ and selects it as the first target pixel denoted by $\mathbf{t}_1$. The FCLS method is then used to estimate the abundance fractions for $\mathbf{t}_0$ and $\mathbf{t}_1$, which is denoted by $\hat{\alpha}_0^{(1)}(\mathbf{r})$ and $\hat{\alpha}_1^{(1)}(\mathbf{r})$, respectively, for each pixel vector $\mathbf{r}$. The superscript indicates the number of iterations that have already been executed.

Then, the algorithm approximates $\mathbf{r}$ by an optimally constrained linear mixture of $\mathbf{t}_0$ and $\mathbf{t}_1$ given by $\hat{\alpha}_0^{(1)}(\mathbf{r})\mathbf{t}_0 + \hat{\alpha}_1^{(1)}(\mathbf{r})\mathbf{t}_1$, and the pixel that yields the largest LSE from its estimated mixture is selected as the second target pixel $\mathbf{t}_2$. The same procedure is repeated until a set of target pixels

$\{\mathbf{t}_0, \mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_{p-1}\} = \{\mathbf{t}_0\} \cup \{\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_{p-1}\}$ is extracted. When the UFCLS algorithm is terminated, the final generated target set $\{\mathbf{t}_0, \mathbf{t}_1, \ldots, \mathbf{t}_{p-1}\}$ is selected as a set of initial endmembers $\{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \ldots, \mathbf{e}_p^{(0)}\}$ for an EEA.

C. IEA

The IEA algorithm proposed in [10] is similar to the UFCLS algorithm in the sense that both make use of FCLS-based linear spectral unmixing to search for possible endmembers. In this paper, we use a special case of the IEA algorithm in which the final spectra obtained by the algorithm are not obtained as averaged values of a set of pixels but as real pixels in the data instead. Therefore, the only difference between the UFCLS algorithm and our version of the IEA algorithm is the starting value that each algorithm uses. While the UFCLS selects the pixel vector with the maximum length as the initial target pixel, the IEA algorithm selects the mean spectrum of the data to start its searching process.

D. Maximin-Distance Algorithm

In this subsection, we describe a very simple EIA called the maximin-distance algorithm, which has been commonly used in pattern recognition applications [11]. It can generate a reasonably good set of initial endmembers.

Let the first initial endmember obtained be the pixel vector with the maximum length, i.e., $\mathbf{e}_1 = \arg\{\max_{\mathbf{r}} \mathbf{r}^T\mathbf{r}\}$. Then, for each $2 \leq j \leq p$, the jth endmember $\mathbf{e}_j$ with the largest distance to the set $S_{j-1} = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_{j-1}\}$ is defined and found as the following expression: $\mathbf{e}_j = \arg\{\max_{\mathbf{r}} d(\mathbf{r}, S_{j-1})\}$, where $d(\mathbf{r}, S_{j-1}) = \min_{1 \leq k \leq j-1} d(\mathbf{r}, \mathbf{e}_k) = \min\{d(\mathbf{r}, \mathbf{e}_1), d(\mathbf{r}, \mathbf{e}_2), \ldots, d(\mathbf{r}, \mathbf{e}_{j-1})\}$. It is worth noting that when $j = 2$, then $S_1 = \{\mathbf{e}_1\}$, in which case $\mathbf{e}_2 = \arg\{\max_{\mathbf{r}} d(\mathbf{r}, \mathbf{e}_1)\}$. It should also be noted that the distance measure used in the maximin-distance algorithm can be any spectral similarity measure such as the Euclidean distance, Spectral Angle Mapper (SAM), or spectral information divergence (SID) [5]. We rely on SAM to produce a set of initial endmembers $\{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \ldots, \mathbf{e}_p^{(0)}\}$ using the maximin-distance algorithm.

IV. EEAs

Two well-known EEAs have been considered for our comparative study and analysis, i.e., the PPI in [12] and the N-FINDR algorithm in [13]. The reason for our selection is based on the fact that PPI has been widely available via the Research Systems ENVI software package [17], whereas N-FINDR has been openly used in the public domain for endmember extraction.

Due to the lack of detailed implementations of these two algorithms, we intend to offer our own understanding of these two algorithms by providing detailed implementations. Hopefully, these experiments may allow any user who is interested in the two algorithms to repeat our experimental results without having to apply particular software packages such as ENVI. It should be pointed out that our implementations have been verified by direct comparisons to the original algorithms and by using a variety of input data sets, where their produced results have been the same in all cases. The step-by-step descriptions of these algorithms are given next.

A. PPI

The PPI algorithm by Boardman *et al.* [12] works as a simple technique designed to search for a set of vertices of a convex hull in an L-dimensional hyperspectral image cube. The algorithm is initialized by a large set of randomly generated L-dimensional vectors called "skewers," which is denoted by $\{\mathbf{skewer}_j\}$, where a skewer is defined as a random unit vector that is used to "skew" the data, i.e., to find extreme pixels in the direction of each skewer. Unfortunately, users of the ENVI software cannot freely choose their own initial endmembers to initialize PPI [17]. In order to adapt PPI to any set of initial endmembers, we delineate the steps carried out by our own version of PPI. A step-by-step description thus follows.

Step 1) *Preprocessing*. Apply a maximum noise fraction (MNF) transformation [18] (or a so-called noise-adjusted principal component (NAPC) in [19]) to reduce the data dimensionality from L to $p - 1$.

Step 2) *Initialization*. Let $\{\mathbf{skewer}_j\}$ be a large set of k randomly generated L-dimensional "skewers," and let t be a preset threshold value.

Step 3) *Skewer projections*. For each $\mathbf{skewer}_j$, all the data sample vectors are projected onto $\mathbf{skewer}_j$ to find sample vectors at their extreme positions to form an extremal set for $\mathbf{skewer}_j$, which is denoted by $S_{\text{extrema}}(\mathbf{skewer}_j)$. Despite the fact that a different $\mathbf{skewer}_j$ generates a different extremal set $S_{\text{extrema}}(\mathbf{skewer}_j)$, it is very likely that some sample vectors may appear in more than one extremal set.

Step 4) *Candidate selection*. Define an indicator function $I_S(\mathbf{r})$ of a given set S as

$$I_S(\mathbf{r}) = \begin{cases} 1; & \text{if } \mathbf{r} \in S \\ 0; & \text{if } \mathbf{r} \notin S \end{cases}. \tag{1}$$

Using the preceding function, select the sample vectors that yield the t largest values of $N_{\text{PPI}}(r) = \sum_j I_{S_{\text{extrema}}(\mathbf{skewer}_j)}(\mathbf{r})$.

Step 5) *Endmember selection*. Load the set of selected $\{\mathbf{r}_i\}_{i=1}^t$ pixels in an L-dimensional visualization tool [17] and manually select a subset of $p \leq t$ pixels $\{\mathbf{e}_j\}_{j=1}^p \subseteq \{\mathbf{r}_i\}_{i=1}^t$, which fall at the corners of the input data cube.

All the steps executed in the preceding PPI algorithm [(except step 5), which is subject to human manipulations] have been verified via extensive experiments using Research Systems ENVI 3.6, where both versions produced the same results. The total number of runs is determined by the number of skewers used throughout the process, and the set of final pixels $\{\mathbf{r}_i\}_{i=1}^t$ will strongly depend on the selection of skewers, which are generated by a random process.

Our aim in this paper is to use the set of spectrally distinct pixels produced by an EIA as skewers for the PPI, instead of using randomly generated skewers. It should be noted that

the PPI only generates a list of candidates from which final endmembers must be manually selected. A (subjective) manual selection procedure is consequently performed by successively projecting the data toward lower dimensional spaces using the L-dimensional visualization tool available from Research Systems ENVI 3.6. To compare other methods, final endmembers are not generated as mean spectra of pixel clusters near the extremes of skewer projections. Instead, individual image pixels falling at the corners of the data cloud are selected, and the resulting pixel signatures are then used to produce the final endmember set. In this case, an endmember is actually a pixel representing a pure pixel. It can also be called an endmember pixel.

B. N-FINDR

Another common EEA is the N-FINDR, which was developed by Winter [13]. This algorithm assumes that the L-dimensional volume formed by a simplex with $(L+1)$ vertices specified by the purest pixels is always larger than those formed by any other combination of $(L+1)$-dimensional pixel vectors. This general idea was described in [13], but this source lacked a detailed step-by-step algorithmic implementation. The steps to implement the N-FINDR are given as follows.

Step 1) *Preprocessing.* Apply MNF or NAPC to reduce data dimensionality from L to $p-1$.

Step 2) *Initialization.* Let $\{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \ldots, \mathbf{e}_p^{(0)}\}$ be a set of initial endmembers randomly generated from the data. It should be noted that the original N-FINDR algorithm description in [13] does not provide any information about how to determine the number of endmembers p to be extracted.

Step 3) *Volume calculation.* At iteration $k \geq 0$, find $V(\mathbf{e}_1^{(\mathrm{k})}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{e}_p^{(\mathrm{k})})$ defined by

$$V\left(\mathbf{e}_1^{(\mathrm{k})}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{e}_p^{(\mathrm{k})}\right) = \frac{\left|\det\begin{bmatrix} 1 & 1 & \cdots & 1 \\ \mathbf{e}_1^{(\mathrm{k})} & \mathbf{e}_2^{(k)} & \cdots & \mathbf{e}_p^{(\mathrm{k})} \end{bmatrix}\right|}{(p-1)!} \quad (2)$$

which is proportional to the volume of the simplex with vertices $\mathbf{e}_1^{(\mathrm{k})}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{e}_p^{(\mathrm{k})}$ denoted by $S(\mathbf{e}_1^{(\mathrm{k})}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{e}_p^{(\mathrm{k})})$.

Step 4) *Simplex replacement.* For each sample vector $\mathbf{r}$, recalculate the volumes of p simplexes $V(\mathbf{r}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{e}_p^{(\mathrm{k})}), \ldots, V(\mathbf{e}_1^{(\mathrm{k})}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{r})$ denoted by $S(\mathbf{r}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{e}_p^{(\mathrm{k})}), \ldots, S(\mathbf{e}_1^{(\mathrm{k})}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{r})$. Each of the preceding simplexes is formed by replacing one endmember $\mathbf{e}_j^{(\mathrm{k})}$ with the sample vector $\mathbf{r}$. If none of these p recalculated volumes is greater than $V(\mathbf{e}_1^{(\mathrm{k})}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{e}_p^{(\mathrm{k})})$, then no endmember in $\mathbf{e}_1^{(\mathrm{k})}, \mathbf{e}_2^{(\mathrm{k})}, \ldots, \mathbf{e}_p^{(\mathrm{k})}$ is replaced. Otherwise, the endmember absent in the combination resulting in the largest volume among the p simplexes aforementioned is replaced by the sample vector $\mathbf{r}$. In this case, let us assume that such an endmember is denoted by $\mathbf{e}_j^{(\mathrm{k}+1)}$. A new set of endmembers is then produced by letting $\mathbf{e}_j^{(\mathrm{k}+1)} = \mathbf{r}$ and $\mathbf{e}_i^{(\mathrm{k}+1)} = \mathbf{e}_i^{(\mathrm{k})}$ for $i \neq j$. Let $\mathrm{k} \leftarrow \mathrm{k}+1$, and then go to step 3), (this process is repeated until all the sample vectors in the data are exhausted).

As seen in step 4), repeated replacements are required before the algorithm arrives at the final set of desired endmembers. In order to avoid many unnecessary replacements in step 4); the algorithm can greatly benefit from an adequately selected set of initial endmembers.

Before concluding this section, a comment is worth being reiterated. While an EEA generates p endmembers simultaneously, an EIA produces one target pixel at a time "sequentially." Therefore, when the value of p is changed, the EEA must be reimplemented as opposed to an EIA, which always takes advantage of the target pixels previously generated. It is realized by applying a smaller value of p, hence, with no need to generate all the target pixels again. More specifically, let us assume that p target pixels have already been generated. In order for an EIA to produce q target pixels, with $q > p$, we would only have to generate $p+1, p+2, \ldots, q-1, q$ target pixels, without having to refind a whole set of q target pixels. This benefit allows us to save a vast amount of computing time.

V. Synthetic-Image Experiments

Synthetic images are used via computer simulations to demonstrate the significant impact of initialization on endmember extraction. A major advantage of using synthetic imagery is that all the details of the simulated images are known under a completely controllable environment because they can be manipulated individually and precisely. As a result, algorithm performance can be examined objectively and impartially. To do so, the EEAs in Section IV will be used to substantiate our experimental results, and EIA-generated pixels will be extracted directly from the simulated synthetic images by using the algorithms described in Section III, which do not necessarily produce pure signatures as an EEA tends to do. In the use of those pixels as initial endmembers, they are referred to as endmember pixels instead of endmembers.

The reflectance spectra of ten U.S. Geological Survey (USGS) ground-truth mineral spectra (alunite, buddingtonite, calcite, kaolinite, muscovite [see Fig. 1(a)], chlorite, jarosite, montmorillonite, nontronite, and pyrophilite [see Fig. 1(b)] have been managed for computer simulations. All signatures are available online (http://speclab.cr.usgs.gov/spectral-lib.html). The ten signatures in Fig. 1 simulate a square synthetic image scene (designated by CS1) with a size of 100×100 pixels. The four corner pixels of the image labeled 1, 2, 3, and 4 were simulated by the pure spectral signatures of alunite, buddingtonite, calcite, and kaolinite in Fig. 1(a), with the center pixel simulated by the pure signature of muscovite and labeled 5 (see Fig. 2).

The signature abundance decreased linearly at the four corner pixels 1, 2, 3, and 4 [see Fig. 2(a)–(d)] and at the center pixel 5 [see Fig. 2(i)]. The images in Fig. 2(a)–(f) were simulated by assuming that the pixels located at the center of an imaginary circle consist of 100% abundance, with their immediate neighboring pixels made up of progressively reduced (and regularly decreasing) abundance fractions, until

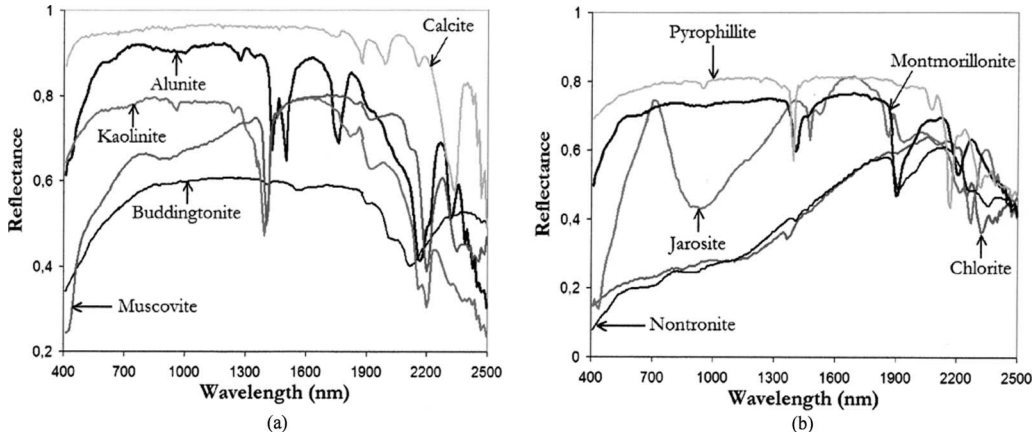

Fig. 1. USGS spectra. (a) Alunite, buddingtonite, calcite, kaolinite, and muscovite. (b) Chlorite, jarosite, montmorillonite, nontronite, and pyrophilite.

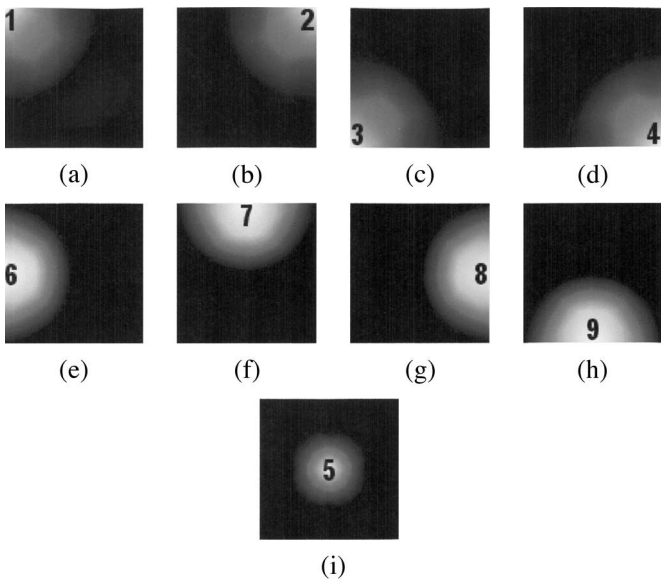

Fig. 2. Abundance maps for minerals in the simulated CS1 scene, where white indicates 100% abundance and black indicates 0% abundance. (a) Alunite. (b) Buddingtonite. (c) Calcite. (d) Kaolinite. (e) Chlorite. (f) Jarosite. (g) Montmorillonite. (h) Pyrophilite. (i) Muscovite.

a fraction of 0% is simulated outside the imaginary circle. The radii of the imaginary circles centered at the pixels labeled 1, 2, 3, and 4 in Fig. 2(a)–(d) were 60 pixels in all cases, whereas the radius of the imaginary circle centered at pixel 5 in Fig. 2(i) was 25 pixels. Four other nonpure signatures (90%-chlorite/10%-nontronite, 80%-jarosite/20%-nontronite, 70%-montmorillonite/30%-nontronite, and 60%-pyrophilite/40%-nontronite) were simulated and placed halfway between a pair of any two corner pixels. They were labeled 6, 7, 8, and 9, as shown in Fig. 2(e)–(h). Purity in signatures also decreased linearly at these four middle pixels. In this case, the radii of the imaginary circles centered at those pixels in Fig. 2(e)–(h) were 49 pixels.

There are only five pure pixels, which are labeled 1, 2, 3, 4, and 5, in the scene. There are also four semipure pixels labeled 6, 7, 8, and 9, which contain samples of signatures made up of 90%-chlorite, 80%-jarosite, 70%-montmorillonite, and 60%-pyrophilite, respectively. Except for the pixels located at the four corners and the center, which were pure, all the pixels in the synthetic image are mixed either in a binary mixture or a ternary mixture. Finally, it should be noted that the abundance fractions of all the simulated image pixels in the maps in Fig. 2 are fully constrained, i.e., they satisfy both abundance sum-to-one and abundance nonnegativity constraints.

TABLE I
VD ESTIMATES FOR CS1 AND CS2 WITH VARIOUS FALSE ALARM PROBABILITIES

Scene	P_F	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
CS1	HFC	9	9	9	9	9
	NWHFC	9	9	9	9	9
CS2	HFC	15	13	11	9	9
	NWHFC	15	12	10	9	9

A different synthetic image (based on the previous one and designated as CS2) was also generated. Its management was carried out as follows: First, brightness variations were simulated by replacing the muscovite mineral in Fig. 2(i) by a shade endmember. The map in Fig. 2(i) was consequently used as a mechanism to scale other spectra in the scene by a number between 0 and 1. Second, slight variations of every considered material (which are observed in real-world applications) were simulated by using various types of randomly used minerals. In particular, different spectra were used, such as 6 (for alunite), 2 (buddingtonite), 3 (calcite), 8 (kaolinite), 6 (chlorite), 9 (jarosite), 10 (montmorillonite), 3 (pyrophillite), and 5 (nontronite), which are all available at the USGS library. Gaussian noise with a $30:1$ signal-to-noise ratio (SNR), as defined in [16], was added to the two synthetic scenes aforementioned to simulate contributions from ambient (clutter) and instrumental sources.

To estimate the number of endmembers in CS1 and CS2, both the HFC and NWHFC methods were applied. Table I lists various values of p that are estimated for CS1 and CS2 by the two methods with different false alarm probabilities. As shown in Table I, both HFC and NWHFC consistently estimated the VD as $p = 9$ for CS1, which is the same number for spectrally distinct signatures present in the synthetic image. The table also reveals that $p = 9$ was a good value for the VD as estimated by both HFC and NWHFC in CS2 with $P_F \in [10^{-4}, 10^{-5}]$.

Table II shows the signature purity (by means of percentages) in the nine pixels extracted from CS1 by the four EIAs

TABLE II
SIGNATURE PURITY OF THE NINE PIXELS EXTRACTED BY THE FOUR EIAS AND BY SEVERAL COMBINATIONS OF EIAS/EEAS FROM SCENE CS1 EXPRESSED AS THE FRACTIONAL ABUNDANCE (IN PERCENTAGE) OF THE MOST PREDOMINANT MINERAL SIGNATURE FOR EACH EXTRACTED PIXEL

Method	Alunite (100)	Buddingtonite (100)	Calcite (100)	Kaolinite (100)	Muscovite (100)	Chlorite (90)	Jarosite (80)	Montmorillonite (70)	Pyrophillite (60)
ATGP	99.22	98.04	98.04	99.22	98.43	86.77	75.29	67.58	58.63
ATGP/PPI	100	100	100	100	100	89.95	79.62	70	60
ATGP/N-FINDR	100	100	100	100	100	89.95	78.31	70	60
UFCLS	98.04	96.34	99.22	95.23	98.43	86.77	74.02	65.93	57.18
UFCLS/PPI	100	100	100	100	100	89.95	79.31	65.93	60
UFCLS/N-FINDR	100	100	100	100	100	88.54	72.21	67.72	57.18
IEA	99.22	99.22	98.04	96.34	98.43	86.77	78.68	69.78	59.12
IEA/PPI	100	100	100	100	100	89.95	78.68	69.78	58.63
IEA/N-FINDR	100	100	100	100	100	89.95	78.68	69.78	60
Maximin	90.44	89.95	93.33	91.79	89.95	86.77	72.21	65.93	57.67
Maximin/PPI	100	100	98.04	99.22	95.23	86.77	75.29	65.93	55.29
Maximin/N-FINDR	99.22	100	98.04	95.23	89.95	88.54	75.29	67.72	56.06
Random/PPI	100	100	98.04	97.25	99.22	86.45	77.74	64.36	54.26
Random/N-FINDR	100	98.04	98.04	100	100	89.95	78.68	64.36	55.63
Winter's N-FINDR	100	99.22	98.04	100	100	90	79.31	65.93	57.18

TABLE III
SIGNATURE PURITY OF THE NINE PIXELS EXTRACTED BY THE FOUR EIAS AND BY SEVERAL COMBINATIONS OF EIAS/EEAS FROM SCENE CS2 EXPRESSED AS THE FRACTIONAL ABUNDANCE (IN PERCENTAGE) OF THE MOST PREDOMINANT MINERAL SIGNATURE FOR EACH EXTRACTED PIXEL

Method	Alunite (100)	Buddingtonite (100)	Calcite (100)	Kaolinite (100)	Shade (100)	Chlorite (90)	Jarosite (80)	Montmorillonite (70)	Pyrophillite (60)
ATGP	94.77	95.23	89.95	95.23	100	86.77	77.74	62.85	55.34
ATGP/PPI	100	100	99.22	98.04	100	86.77	79.62	69.78	59.12
ATGP/N-FINDR	100	100	89.95	95.23	100	89.95	79.62	68.20	57.18
UFCLS	87.48	98.04	87.48	98.04	98.04	81.48	68.41	65.83	48.59
UFCLS/PPI	93.33	98.04	99.22	98.04	100	88.54	79.62	65.83	57.18
UFCLS/N-FINDR	98.82	98.04	98.04	98.04	100	89.95	78.68	65.83	58.26
IEA	98.82	99.22	97.65	98.04	99.22	88.54	79.62	57.88	48.59
IEA/PPI	99.22	99.22	98.04	98.04	100	89.95	79.62	64.21	55.63
IEA/N-FINDR	99.22	99.22	98.04	98.04	100	89.95	79.62	65.83	57.18
Maximin	79.37	74.92	77.74	81.48	87.06	69.49	68.63	53.62	45.33
Maximin/PPI	88.07	90.44	93.33	91.79	87.06	78.68	68.63	62.85	52.02
Maximin/N-FINDR	88.07	90.44	85.46	91.79	90.53	80.28	75.29	62.85	55.34
Random/PPI	94.77	96.34	98.04	95.23	99.22	84.36	75.29	63.19	48.59
Random/N-FINDR	95.23	97.25	90.44	90.44	98.04	86.77	77.74	64.21	48.59
Winter's N-FINDR	93.33	98.04	87.83	87.06	100	87.74	78.31	64.21	54.26

and several combinations of EIAs/EEAs. Only the fractional abundance of the most predominant signature in each extracted pixel is reported. Most EIAs extracted pixels that are close to the five pure pixels (100% abundance), which are located at the vertices and center of the scene. The IEA algorithm performed slightly better than ATGP for initial endmembers, whereas the maximin algorithm provided the lowest performance rate.

As revealed in Table II, the four EIAs were unable to locate the purest versions of "mixed" (semipure) endmembers: chlorite (90% abundance), jarosite (80%), montmorillonite (70%), and pyrophillite (60%). In contrast, as the four EIAs were used to initialize the PPI and N-FINDR, the pixels extracted by both algorithms were closer to the purest form of each spectrally distinct material in the scene. Interestingly enough, Table II discloses lower signature purity in the case of endmembers extracted by EEAs, in particular, for the most highly "mixed" endmembers, when random initial endmembers were used to initialize the PPI and N-FINDR. Since the way of generating random initial endmembers in the original N-FINDR algorithm by Winter is not available in [13], their original software has also been run for comparison purposes, and similar results have been reached.

Table III shows the results obtained for the CS2 scene, where the IEA algorithm provides the best initial endmembers in terms of signature purity for the four tested EIAs. In turn, ATGP yields a highly similar performance, being able to extract the shade endmember at the center of the scene in pure form.

As the results in Tables II and III imply, ATGP seems to perform best in combination with PPI and slightly better than the UFCLS and IEA algorithms in combination with N-FINDR. This conclusion indicates that a judicious selection of initial endmembers can be beneficial to the endmember-searching process. Using random initial endmembers tends to cause significant drops in signature purity for both pure and "mixed" endmembers. Additionally, we have observed that PPI produces the same final set of endmembers when the number of randomly generated skewers has been set to $k = 1000$ or above (values of $k = 10^4, 10^5,$ and 10^6 were also tested). Based on the preceding experiment, we have set the cutoff threshold parameter t to the mean of the PPI scores being obtained after $k = 1000$ iterations. These parameter values agree with the previous scores in [2].

For illustration purposes, Table IV cross-tabulates the number of replacements and computing time in seconds (measured

TABLE IV
NUMBER OF REPLACEMENTS AND COMPUTING TIME MEASURED FOR EIAS AND EIA + EEAS FOR CS1 AND CS2

	CS1		CS2	
Method	Replacements	Time (seconds)	Replacements	Time (seconds)
ATGP	-	16.15	-	15.13
ATGP/PPI	4	21.12	5	20.36
ATGP/N-FINDR	4	21	5	20.36
UFCLS	-	37.89	-	38.34
UFCLS/PPI	6	44.1	5	44.16
UFCLS/N-FINDR	5	43.83	5	43.63
IEA	-	43.12	-	41.23
IEA/PPI	6	49.17	5	47.17
IEA/N-FINDR	5	48.86	6	47.48
Maximin	-	11.96	-	12.45
Maximin/PPI	6	18.4	7	19.38
Maximin/N-FINDR	7	18.94	8	19.67
Random/PPI	-	39.95	-	43.28
Random/N-FINDR	25	36.24	28	38.94
Winter's N-FINDR	-	11.25	-	12.23

in a 512-MB RAM personal computer with a 2.6-GHz AMD Athlon processor). This cross measurement is performed for several combinations of EIAs/EEAs, where the computing time for the tested EIAs is also reported in seconds. Two main observations need to be made. First, not providing the number of replacements for the random/PPI combination in Table IV is due to the fact that in this case, the initial skewers are artificially generated unit vectors (i.e., not real pixels in the input data). However, it is worth mentioning that the number of distinct pixels that are selected at least once as extremes throughout the process is 36 in the case of CS1 and 41 in the case of CS2. Second, the computing times calculated for PPI do not include the manual step of the algorithm (i.e., they were calculated by merely following steps 1)–4).

In the case of the N-FINDR, when used with EIA-generated pixels as initial endmember pixels, a much smaller number of replacements is observed in comparison with its initialization made by random endmembers. We should point out that the number of replacements for Winter's N-FINDR algorithm has not been included in Table IV because we lacked enough specific knowledge about the initial (randomly generated) endmembers in the original algorithm [13].

Using EIAs to initialize EEAs often results in a reduction of computing time. Since the two synthetic images used for computer simulations are relatively small, improvement over computational performance has not been significant. Such an enhanced working rate has especially been lower than the performance reported for a real hyperspectral data set, as shall be examined in the next section.

VI. REAL-IMAGE EXPERIMENTS

The image scene in Fig. 3 was selected for real-data experiments. It was collected by the AVIRIS spectrometer over the Cuprite Mining District, NV, in 1997. The scene is available online at http://aviris.jpl.nasa.gov/html/aviris.freedata.html and consists of 350×350 pixels and 224 bands.

The data set is well understood mineralogically, a fact that has made this scene a standard test site for validation and quantitative assessment of EEAs. The data set was also atmospherically corrected and available in reflectance units. Such a correction enables the results to be related to the USGS laboratory reflectance data (available at http://speclab.cr.usgs.gov). Prior to the analysis, 105–115 and 150–170 bands were removed due to water absorption and low SNR.

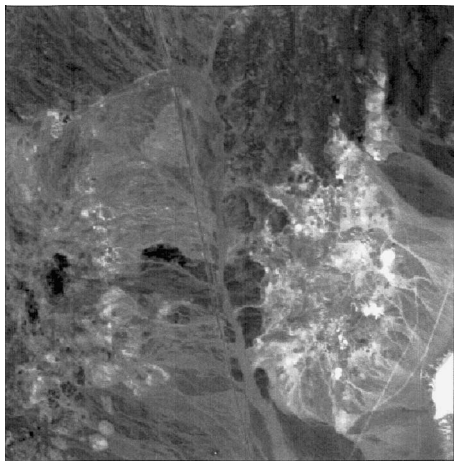

Fig. 3. Spectral band # 50 (827 nm) of the AVIRIS Cuprite scene.

TABLE V
VD ESTIMATES FOR THE AVIRIS CUPRITE SCENE WITH VARIOUS FALSE ALARM PROBABILITIES

P_F	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
HFC	37	27	23	22	21
NWHFC	32	24	22	22	18

The mineral signatures shown in Fig. 1 appear prominently exposed and in pure form in the Cuprite scene, as indicated by a USGS Tetracorder map that shows the spatial distribution of minerals in the area (at http://speclab.cr.usgs.gov/cuprite95.tgif.2.2um_map.gif). This map reveals several areas made up of pure mineral signatures, with the buddingtonite and calcite minerals often appearing as anomalies, along with spatially homogeneous areas made up of alunite, kaolinite, and montmorillonite at both sides of the road, crossing the area from north to south. This area includes the well-known "montmorillonite playa," which is located at the rightmost bottom corner of the scene. Additional minerals present in the area include chalcedony (which also appears in spatially homogeneous areas), dickite, halloysite, andradite, dumortierite, and sphene, where the latter three minerals have been recently reported to be present [20]. Most mixed pixels in the scene consist of alunite, kaolinite, and/or muscovite.

The number of endmembers p has been first estimated using the VD concept. Table V tabulates various values of p calculated via the HFC and NWHFC methods (with different false alarm probabilities P_F), where a reasonable estimate for the VD seemed to be 22 when P_F was set to 10^{-3} or 10^{-4}. This number was also in accordance with the number of distinct minerals labeled in the USGS Tetracorder map. The results in Table V are quite similar, meaning that the spectra representing pure mineral signatures have consistently been extracted, and most of the extracted pixels are overlapped for different values

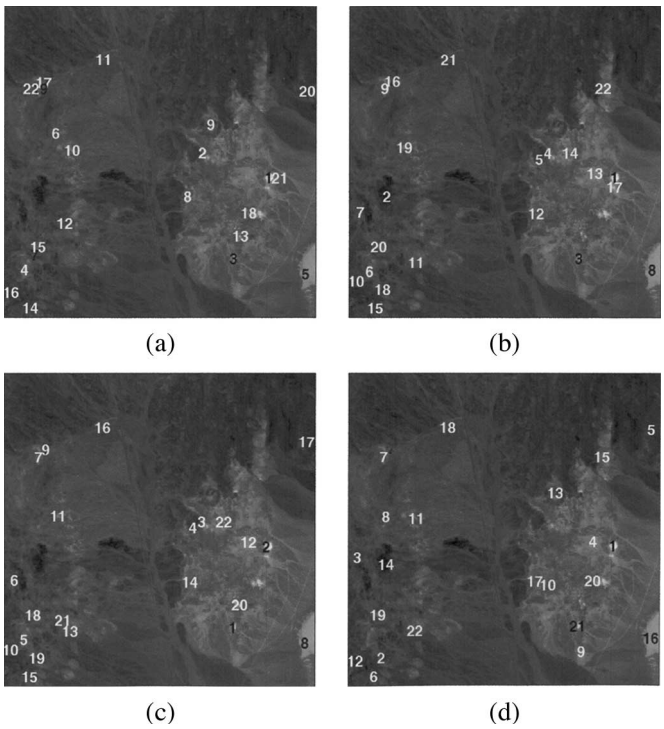

Fig. 4. Twenty-two initial endmember pixels, labeled in order, generated by four EIAs for the Cuprite image scene. (a) ATGP. (b) UFCLS. (c) IEA. (d) Maximin.

of p. Therefore, only the experiments for a case study with $p = 22$ are discussed.

Fig. 4 shows the 22 target pixels (labeled in order) generated by the four considered EIAs. The pixel locations can be assessed by using the reference USGS Tetracorder map. Using these EIA-generated target pixels as initial endmember pixels, the PPI and N-FINDR algorithms have been able to extract 22 endmember pixels. The number of skewers used for PPI has been empirically set to $k = 10^4$, and t has been set to the mean of PPI scores obtained after $k = 10^4$ iterations [2]. On the other hand, Table VI shows the SAM values [5] between the endmembers in the final set extracted by different EIA/EEA combinations and their corresponding spectral signatures in the USGS library. Small SAM values indicate a high spectral similarity. It is worth noting that in several cases, different pure variants of the same mineral were extracted as different endmembers, in particular, for the alunite, kaolinite, and muscovite minerals. In those cases, the table shows the SAM scores for each of the endmembers associated with the variants of the same mineral. The numbers in parentheses indicate the number of times each EIA/EEA combination achieved the minimum (or equal minimum) SAM score (out of 22).

As shown in Table VI, the ATGP/PPI and ATGP/N-FINDR combinations result in the largest number of minimal SAM values among all considered EIAs. These results are consistent with those found previously in the computer simulations. On the other hand, the maximin/PPI and maximin/N-FINDR combinations were the least effective in real-data experiments, as in the case of the computer simulations. Table VI also demonstrates that atmospheric transmission effects were particularly relevant in specific minerals such as the two variants of kaolinite, halloysite, or the andradite (the SAM values for these minerals were generally higher than those reported for other minerals). This situation reveals the importance of atmospheric corrections in hyperspectral data processing. Overall, Table VI indicates that an intelligent algorithm initialization can result in improved endmembers according to signature purity criteria.

Table VII tabulates the computing time performed by the four EIAs in seconds along with the number of replacements and computing time for PPI (using EIA-generated initial endmembers) and N-FINDR (using both random initial endmembers and EIA-generated endmembers). The overall number of pixels selected as extremes by the random/PPI combination is 82, resulting in a highly significant computation time for the algorithm (the times reported in Table VII only refer to steps 1)–4) of PPI). The rightmost column in Table VII lists the number of final endmember pixels overlapped with the initial endmembers for each EIA/EEA combination. Twelve out of 22 ATGP-generated initial pixels turn out to be the final endmembers for PPI, while ten out of 22 ATGP-generated target pixels end up being final endmembers for the N-FINDR. Both the UFCLS and IEA algorithms also produce many initial pixels that became final endmembers for PPI (ten for both the UFCLS and IEA algorithms) and N-FINDR (five for UFCLS and seven for IEA).

ATGP has proved to be the most suitable EIA in terms of final overlapped endmember, as opposed to maximin, which seemed to be the least effective. A great deal of saving is also evident in terms of the number of replacements and computing time, to which using EIA-generated pixels as initial endmembers leads. Although replacements and overlapped pixels have not been included in Winter's N-FINDR algorithm, a computationally effective software version of the N-FINDR is available. Unfortunately, it has been run as a software package, and knowing how the initial endmembers were selected has been unfeasible. Such an improvement over computing time may be due to the use of a programming language such as C++, which is in contrast with our own higher level implementation in Matlab being significantly slower. The experiments may thus entail significant improvement over our EIA/EEA algorithms for future work.

Finally, we would like to emphasize that our proposed EIA/EEAs have been evaluated by a qualitative cross validation among different spectral unmixing techniques. The objective has been to observe if the improvements derived from the EIA application to the final unmixing can be significant. The two used methods for evaluation were fully constrained (FCLSU) and unconstrained linear spectral unmixing (ULSU) [5]. Quantitative experiments demonstrated that the use of EIAs (instead of random endmembers) to provide initial conditions for the PPI and N-FINDR generally resulted in very similar abundance fractions estimated by both FCLSU and ULSU. A common indicator of poor model fitting and/or inappropriate selection of initial endmembers is estimation of negative abundance fractions by ULSU. This situation was very rarely observed when an EIA was used for initialization. In contrast, a much more significant fraction of negative fractional abundances was observed when random endmembers were applied to the initial condition.

TABLE VI
SAM-BASED SPECTRAL SIMILARITY SCORES BETWEEN THE USGS MINERAL SPECTRA AND THEIR CORRESPONDING ENDMEMBER PIXELS PRODUCED BY SEVERAL COMBINATIONS OF AN EIA FOLLOWED BY AN EEA

USGS Mineral signature	Random/ PPI (3)	Random/ N-FINDR (1)	ATGP/ PPI (20)	ATGP/ N-FINDR (14)	UFCLS/ PPI (7)	UFCLS/ N-FINDR (4)	IEA/ PPI (5)	IEA/ N-FINDR (2)	Maximin/ PPI (4)	Maximin/ N-FINDR (3)	Winter's N-FINDR (5)
Alunite #1	0.084	0.112	0.084	0.105	0.084	0.112	0.084	0.084	0.084	0.084	0.081
Alunite #2	0.095	0.103	0.094	0.098	0.116	0.112	0.116	0.116	0.116	0.112	0.095
Alunite #3	0.112	0.110	0.073	0.077	0.086	0.094	0.073	0.094	0.110	0.094	0.092
Alunite #4	0.099	0.094	0.042	0.045	0.047	0.063	0.047	0.055	0.094	0.103	0.102
Buddingtonite	0.106	0.095	0.071	0.073	0.071	0.073	0.071	0.076	0.071	0.073	0.084
Calcite	0.105	0.112	0.099	0.105	0.093	0.105	0.094	0.106	0.093	0.093	0.105
Kaolinite #1	0.136	0.151	0.136	0.151	0.136	0.156	0.136	0.136	0.136	0.136	0.136
Kaolinite #2	0.145	0.134	0.112	0.119	0.126	0.134	0.134	0.167	0.167	0.134	0.152
Muscovite #1	0.108	0.106	0.083	0.106	0.083	0.077	0.083	0.085	0.093	0.105	0.094
Muscovite #2	0.092	0.098	0.081	0.081	0.092	0.092	0.100	0.098	0.102	0.114	0.098
Muscovite #3	0.091	0.109	0.065	0.074	0.073	0.071	0.088	0.091	0.109	0.107	0.112
Chlorite	0.125	0.096	0.058	0.065	0.080	0.074	0.090	0.094	0.114	0.123	0.096
Jarosite	0.112	0.108	0.083	0.091	0.094	0.102	0.097	0.097	0.106	0.112	0.102
Montmorillonite	0.106	0.096	0.091	0.099	0.096	0.099	0.096	0.107	0.112	0.120	0.089
Nontronite	0.102	0.099	0.054	0.058	0.062	0.073	0.077	0.102	0.123	0.121	0.099
Pyrophillite	0.094	0.090	0.077	0.080	0.083	0.091	0.081	0.090	0.117	0.112	0.090
Chalcedony	0.099	0.102	0.081	0.094	0.102	0.102	0.102	0.105	0.104	0.099	0.102
Dickite	0.101	0.098	0.063	0.061	0.073	0.084	0.073	0.073	0.096	0.101	0.080
Halloysite	0.119	0.134	0.112	0.117	0.117	0.125	0.117	0.119	0.125	0.125	0.134
Andradite	0.142	0.139	0.142	0.142	0.142	0.167	0.151	0.142	0.142	0.167	0.155
Dumortierite	0.082	0.085	0.047	0.040	0.045	0.068	0.072	0.081	0.082	0.082	0.085
Sphene	0.078	0.080	0.051	0.051	0.062	0.070	0.068	0.073	0.080	0.097	0.073

TABLE VII
NUMBER OF REPLACEMENTS AND COMPUTING TIME MEASURED FOR EIAS AND EIA + EEAS FOR THE AVIRIS SCENE. THE NUMBER OF OVERLAPPED PIXELS BETWEEN THE INITIAL AND FINAL ENDMEMBER SETS ARE ALSO REPORTED FOR EACH EIA/EEA COMBINATION

Method	Replacements	Computing time (seconds)	Pixels overlapped
ATGP	-	161	-
ATGP/PPI	10	282	12
ATGP/N-FINDR	12	264	10
UFCLS	-	302	-
UFCLS/PPI	12	424	10
UFCLS/N-FINDR	17	428	5
IEA	-	321	-
IEA/PPI	12	441	10
IEA/N-FINDR	12	412	7
Maximin	-	145	-
Maximin/PPI	17	279	5
Maximin/N-FINDR	18	273	4
Random/PPI	82	3138	-
Random/N-FINDR	64	487	4
Winter's N-FINDR	-	45	-

VII. CONCLUSION

This paper investigates two issues arising in EEA initialization that are crucial for algorithm design and yet have been generally overlooked. The first major issue is determination of the number of endmembers used for an EEA. This problem can be resolved by using the concept of VD. According to our experiments, the VD provides a good estimate of the number of endmembers. The second major issue is how to produce an appropriate set of initial endmembers to be used for an EEA. This problem can be taken care of by introducing the concept of EIA, which produces a set of target pixels used to initialize an EEA. To meet this need, four algorithms are developed, including the ATGP, the UFCLS algorithm, IEA, and the maximin-distance algorithm. Experimental results reveal that the ATGP outperforms the other EIAs on both synthetic and real-image experiments. This algorithm was only slightly slower than maximin but much faster than all other initialization methods tested in this paper. Additionally, experiments also demonstrate that an EIA not only can speed up algorithm performance but also can produce a set of good initial endmembers, many of which eventually become the final endmembers. Finally, despite the PPI and N-FINDR having been known in public domain for years, their detailed descriptions have not been available for users who might be interested in such algorithms without the need of using the software packages. The step-by-step algorithmic implementations presented in this paper may offer an additional benefit to those users.

ACKNOWLEDGMENT

The authors would like to thank M. Winter for providing the N-FINDR algorithm software for evaluation, the anonymous reviewers for the suggestions and comments, and A. Curado (Department of English, UEX) for the linguistic revision.

REFERENCES

[1] R. A. Schowengerdt, *Remote Sensing: Models and Methods for Image Processing*, 2nd ed. New York: Academic, 1997.
[2] 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.
[3] C.-I Chang and A. Plaza, "Fast iterative algorithm for implementation of pixel purity index," *IEEE Trans. Geosci. Remote Sens. Letters*, vol. 3, no. 1, pp. 63–67, Jan. 2006.
[4] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Norwell, MA: Kluwer, 1992.
[5] C.-I Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*. Norwell, MA: Kluwer, 2003.
[6] C.-I Chang and Q. Du, "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 3, pp. 608–619, Mar. 2004.
[7] R. W. Hamming, *Introduction to Applied Numerical Analysis*. New York: Hemisphere, 1989.

[8] H. Ren and C.-I Chang, "Automatic spectral target recognition in hyperspectral imagery," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 39, no. 4, pp. 1232–1249, Oct. 2003.
[9] D. Heinz and C.-I Chang, "Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 39, no. 3, pp. 529–545, Mar. 2001.
[10] R. A. Neville, K. Staenz, T. Szeredi, J. Lefebvre, and P. Hauff, "Automatic endmember extraction from hyperspectral data for mineral exploration," in *Proc. 4th Int. Airborne Remote Sens. Conf. and Exhib. 21st Canadian Symp. Remote Sens.*, Ottawa, ON, Canada, Jun. 1999, pp. 21–24.
[11] J. T. Tou and R. C. Gonzalez, *Pattern Recognition Principles*. Reading, MA: Addison-Wesley, 1974, pp. 92–94.
[12] J. W. Boardman, F. A. Kruse, and R. O. Green, "Mapping target signatures via partial unmixing of AVIRIS data," in *Proc. Summaries JPL Airborne Earth Sci. Workshop*, Pasadena, CA, 1995, pp. 23–26.
[13] M. E. Winter, "N-FINDR: An algorithm for fast autonomous spectral endmember determination in hyperspectral data," in *Proc. SPIE—Image Spectrometry V*, 1999, vol. 3753, pp. 266–277.
[14] J. C. Harsanyi, W. Farrand, and C.-I Chang, "Detection of subpixel spectral signatures in hyperspectral image sequences," in *Proc. Amer. Soc. Photogrammetry and Remote Sens.*, Reno, NV, 1994, pp. 236–247.
[15] I. Katsavounides, C. C. J. Kuo, and Z. Zhang, "A new initialization technique for generalization Lloyd iteration," *IEEE Trans. Signal Process.*, vol. 1, no. 10, pp. 144–146, Oct. 1994.
[16] J. C. Harsanyi and C.-I Chang, "Hyperspectral image classification and dimensionality reduction: An orthogonal subspace projection approach," *IEEE Trans. Geosci. Remote Sens.*, vol. 32, no. 4, pp. 779–785, Jul. 1994.
[17] Research Systems, *ENVI User's Guide*, Research Systems, Inc., Boulder, CO, 2001.
[18] A. A. Green, A. A. 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.
[19] J. B. Lee, A. S. Woodyatt, and M. Berman, "Enhancement of high spectral resolution remote sensing data by a noise-adjusted principal components transform," *IEEE Trans. Geosci. Remote Sens.*, vol. 28, no. 3, pp. 295–304, May 1990.
[20] J. M. P. Nascimento and J. M. B. Bioucas, "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 898–910, Apr. 2005.

Antonio Plaza (M'05) received the M.S. and Ph.D. degrees in computer science from the University of Extremadura, Cáceres, Spain, in 1997 and 2002, respectively.

Since 2000, he has been an Associate Professor with the Department of Computer Science, University of Extremadura, where he was an Assistant Professor from 1997 to 1999. He was also a Visiting Researcher with the Remote Sensing Signal and Image Processing Laboratory, University of Maryland Baltimore County, Baltimore, the Applied Information Sciences Branch, Goddard Space Flight Center, Greenbelt, MD; and the AVIRIS Data Facility, Jet Propulsion Laboratory, Pasadena, CA. He has authored and coauthored more than 100 publications including journal papers and conference proceedings. He is coediting (with Prof. C.-I Chang) a book on *High-Performance Computing in Remote Sensing* (Chapman & Hall/CRC Press). His research interests include computer vision, image processing, pattern recognition, and efficient implementations of large-scale scientific problems on parallel and distributed systems, including commodity clusters and heterogeneous networks of workstations, grid computing facilities, and hardware-based computer architectures such as graphic processing units or field-programmable gate arrays.

Chein-I Chang (S'81–M'82–SM'92) received the B.S. degree in mathematics from Soochow University, Taipei, Taiwan, R.O.C., the M.S. degree in mathematics from the Institute of Mathematics, National Tsing Hua University, Hsinchu, Taiwan, the M.A. degree in mathematics from the State University of New York, Stony Brook, the M.S. and M.S.E.E. degrees from the University of Illinois, Urbana-Champaign, and the Ph.D. degree in electrical engineering from the University of Maryland, College Park.

Since 1987, he has been with the University of Maryland Baltimore County (UMBC), Baltimore, where he is currently a Professor with the Department of Computer Science and Electrical Engineering. From 2004 to 2005, he was a Visiting Research Specialist with the Institute of Information Engineering, National Cheng Kung University, Tainan, Taiwan. From 2005 to 2006, he was a Distinguished Lecturer Chair with the National Chung Hsing University, Taichung, Taiwan, sponsored by the Ministry of Education in Taiwan. He is currently holding a Chair Professorship with the National Chung Hsing University. He is the author of the book *Hyperspectral Imaging: Techniques for Spectral Detection and Classification* (Kluwer, 2003) and the two edited books, *Recent Advances in Hyperspectral Signal and Image Processing* (Trivandrum, Kerala: Research Signpost, Trasworld Research Network, India, 2006) and *Hyperspectral Data Exploitation: Theory and Applications* (Wiley, 2006). He will also author the book *Hyperspectral Imaging: Signal Processing Algorithm Design and Analysis* (Wiley, 2007) and coedit with A. Plaza the book *High Performance Computing in Remote Sensing* (CRC, 2007). He is the holder of three patents, with several patents on hyperspectral image processing. His research interests include multispectral/hyperspectral image processing, automatic target recognition, medical imaging, information theory and coding, signal detection and estimation, and neural networks. He serves on the Editorial Board of the *Journal of High-Speed Networks* and was the Guest Editor of a special issue of the same journal on telemedicine and applications. He will co-Guest Edit three special issues on: 1) broadband multimedia sensor networks in healthcare applications for the *Journal of High-Speed Networks*; 2) systems and developments in health care applications for the *EURASIP Journal on Applied Signal Processing* (2007); and 3) high-performance computing of hyperspectral imaging for the *International Journal of High Performance Computing Applications* (December 2007).

Dr. Chang is a Fellow of the International Society for Optical Engineers (SPIE). He is an Associate Editor in the area of hyperspectral signal processing for the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING. He was a recipient of the National Research Council Senior Research Associateship Award from 2002 to 2003, which was sponsored by the U.S. Army Soldier and Biological Chemical Command, Edgewood Chemical and Biological Center, Aberdeen Proving Ground, Aberdeen, MD. He is also a Member of Phi Kappa Phi and Eta Kappa Nu.