APPROVAL SHEET

Title of Dissertation: Finding Endmember Classes and Endmembers in Hyperspectral Images

Name of Candidate: Cheng Gao
Doctor of Philosophy, 2016

Dissertation and Abstract Approved: Chein-I Chang

Chein-I Chang
Professor
Department of Computer Science and Electrical Engineering

Date Approved: 4/27/16

Abstract

Title of Dissertation: **FINDING ENDMEMBER CLASSES AND ENDMEMBERS IN HYPERSPECTRAL IMAGES**

Name of Candidate: **Cheng Gao, Ph. D. 2016**

Dissertation Directed by: **Dr. Chein-I Chang**

Professor

Department of Computer Science and Electrical Engineering

Remotely sensed images have been used in a broad range of applications ranging from chemical/biological defense, geology, agriculture to environmental protection, law enforcement and intelligence applications. With the recent advanced technology, remote sensing instruments have significantly improved spatial resolution and also spectral resolution. The images from these instruments have hundreds of different contiguous spectral bands with typical spectral resolution of approximately 10 nanometers which can be used to uncover subtle material substances that cannot be resolved by multispectral sensors.

With significantly improved spectral and spatial resolutions provided by hyperspectral imaging sensors, one challenging issue is to find how many spectrally distinct signatures are present in the resulting image data. Recently, Virtual Dimensionality (VD) was developed to address this issue which is defined as the number of spectrally distinct signatures in hyperspectral imagery. Once VD is

determined, a follow-up issue is to find these signatures from the image data. As defined, an endmember is an ideal, pure signature for a class, more specifically, spectral class. However, in reality, the spectral signature for a material may vary due to a number of reasons including environmental, atmospheric and temporal factors. As a result, an endmember may appear in various forms in hyperspectral images. In order to resolve endmember variability issue, this dissertation develops three new algorithms.

The first one is called Endmember Variability Algorithm (EVA) which is an unsupervised endmember class finding algorithm using a half-way A second one develops a new criterion similar to Fisher's ratio used in Fisher's Linear Discriminant Analysis (FLDA) and various versions of algorithms using the new criterion are further designed. Unlike current methods that either require training data set or need a predefined parameter, two proposed methods can find endmembers and their corresponding classes in a complete unsupervised manner. A third algorithm is Fully Constrained Least Square Endmember Finding Algorithm (FCLS EFA) which uses average unmixing error to find endmembers that assumed to be most representative pixels as endmembers for linear spectral representation via a linear mixing model.

Finding Endmember Classes and Endmembers in Hyperspectral Images

By

Cheng Gao

Dissertation submitted to the Faculty of the Graduate School

of the University of Maryland, Baltimore County, in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

2016

Acknowledgements

In my 3-year Ph. D. study, there are many people I own my gratitude to for their patience, support and encouragement. Without them, there is no way I could make to my Ph. D. First and foremost, I would like to thank my research advisor Dr. Chein-I Chang for providing his great insights into the hyperspectral data processing. He has guided me to the right direction of many references that I was in great need of. He has the expertise and knowledge in identifying potential journal papers for publications and has helped me publish all my papers and researches. This dissertation would never have been completed without his enthusiasm and dedications.

Additionally, I would like to thank my supervisors at University of Maryland, Baltimore, Dr. Peter Hu and Dr. Colin Mackenzie. While I was working as a graduate research assistant at Shock Trauma Research, they have not only provided me with financial support to finish my Ph. D. program, but also taught me their medical expertise, which prepares me better for my future research.

There are many people in my lab that I would also like to say thank you to. In addition to helping me go through all the dilemmas that I have experienced during my research, they have also joined me in discussing the details of my research. They are my honest audiences for my presentations who contributed great comments on my researches. Without their help and support, I would have never been able to achieve where I am. I would also like to thank Dr. Damon Bradley for his great suggestions on my dissertation.

Finally, I would like to thank my many other friends and my parents who live in China for keeping encouraging me through the hardship of my study in U. S.

Table of Contents

List of Figures

Chapter 1 Introduction

Hyperspectral imaging [1] is an emerging technique in remote sensing processing that expands and improves capability of multispectral image analysis in many aspects. In particular, it has found success in many applications ranging from agriculture crop classification, mineral exploration, environmental monitoring to defense, law enforcement and intelligence gathering.

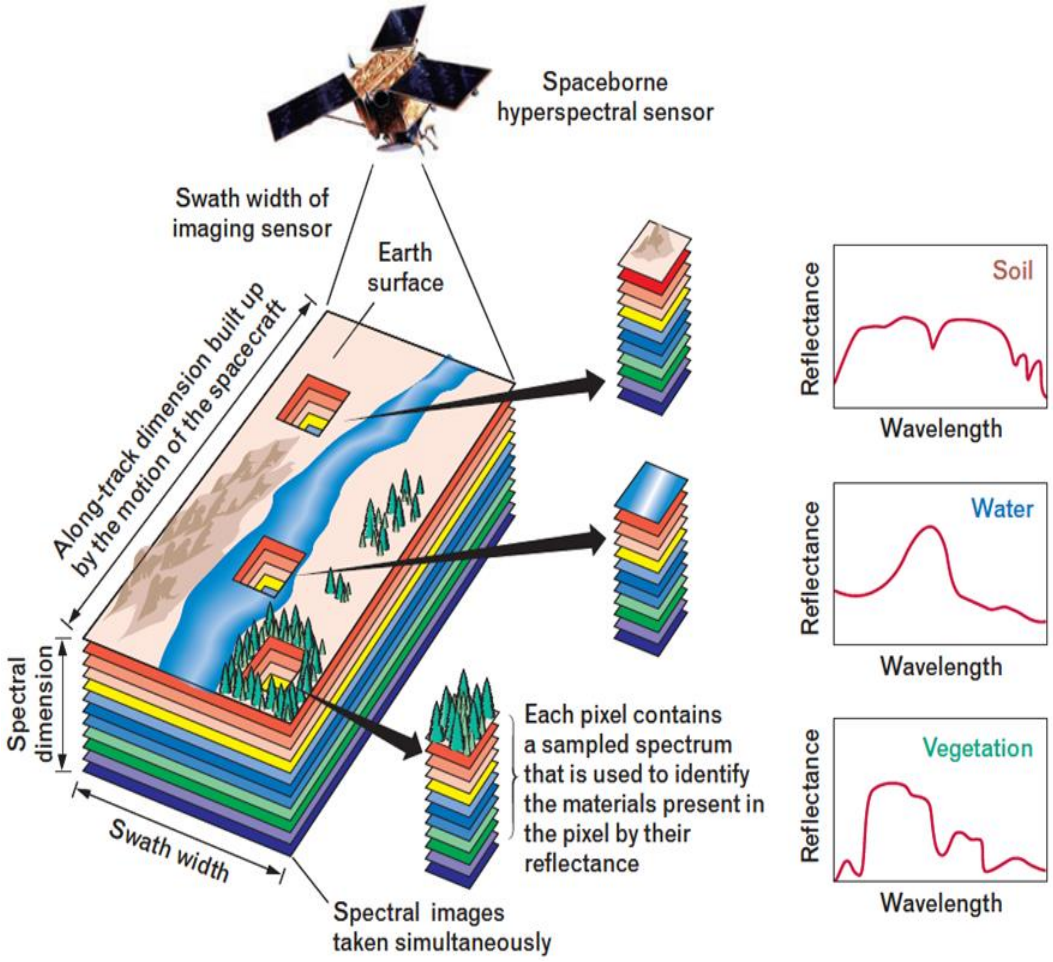


Figure 1.1 Illustration of hyperspectral imagery

Unlike multispectral sensors which utilize tens of discrete spectral bands, hyperspectral remote sensors make use of hundreds of contiguous spectral bands for

data collection. As a result, the acquired data provide wealth of spectral information as shown in Fig 1.1 and can uncover many materials that usually cannot be resolved by multispectral sensors because of low spectral resolution.

With significantly improved spectral and spatial resolutions of hyperspectral imaging sensors, one major and challenging issue is to find how many spectrally distinct signatures are present in the data, each of which represent a particular spectral class. Recently, Virtual Dimensionality (VD) [2] was developed to address this issue which is defined as the number of spectrally distinct signatures in a hyperspectral image. However, determination of VD in the hyperspectral data is very challenging and extremely difficult. In this dissertation, we assume that VD has been determined, by a reliable technique [2], denoted by p.

Once the value of p is determined, a follow-up issue is to find these p spectrally distinct signatures [3]. In a given hyperspectral image, many of such spectrally distinct signatures may appear as so-called endmembers. According to the definition given by Schowengerdt [4], an endmember is an idealistic, pure signature for a class, more specifically, spectral class. For multispectral imagery, an endmember may not be found since most data sample vectors may be heavily mixed due to low spatial and spectral resolution. By contrast, with recent advances of hyperspectral imaging sensors, many subtle material spectral features that cannot be resolved by multispectral imagery can be now uncovered by hyperspectral imagery. These substances are generally not known *a priori* or identified by visual inspection. Endmembers are considered to be one of such substances. In general, their existence in image data is always not guaranteed. Most importantly, once endmembers are

present, their spatial extent is relatively limited. Besides, their sample pools are also very small. Accordingly, they may appear as anomalies. In this case, spatial characteristics offer little advantage in finding endmembers [1].

As noted, justification of existence of endmembers in reality is extremely difficult due to several practical reasons. One is lack of careful data calibration such as atmospheric correction. Another is lack of ground truth such as data base or spectral library. A third one is lack of prior knowledge that can be used for validation. Even if it can, it is extremely expensive to obtain ground truth [5]. Most importantly, due to the fact that endmembers are generally corrupted by many unknown effects, finding such "*true*" endmembers is very challenging because these contaminated endmembers are no longer pure signatures. Under such circumstances, extracting something which is not present in the data does not make sense. Instead, the term of "*finding*" is a more appropriate than "*extracting*" for this dilemma. Unfortunately, in real world application this is indeed the case we encounter all the time. Nevertheless, these signatures still represent most significant information in data interpretation. In order to resolve this dilemma the concept of endmember variability was recently introduced to take into account variability [6-10] of an endmember present in the data. More specifically, instead of working on a single endmember, a group of signatures, referred to as an endmember class, is considered to represent one type of endmember so as to take care of signature corruption caused by physical effects, such as noise, interference encountered in real environments. This dissertation is developed to take up this issue and mainly focuses on endmember class finding while considering its variability. Unlike many algorithms developed in literatures, the

proposed algorithms in this dissertation are unsupervised. In other words, these algorithms do not require training samples. Another advantage of these algorithms is that they do not need to determine a pre-defined parameter such as the mean of endmember class center.

The dissertation is organized as follows. In Chapter 2, different methods that deal with endmember variability in the previous literatures are summarized and compared. In Chapter 3, a new endmember variability algorithm is developed to find endmember classes in an unsupervised fashion. In Chapter 4, a new criterion similar to Fisher's ratio used in Fisher's linear discriminant analysis is developed and various algorithms using the new criterion are further designed. In Chapter 5, a new algorithm is developed to find endmembers while minimizing the average unmixing error. Chapter 6 summarizes the work presented in this dissertation and future studies.

Chapter 2 A Review of Endmember Variability

As described in Chapter 1, hyperspectral data provides us with both rich spectral and spatial information. In a certain hyperspectral image, a measured spectral signature of a material substance may vary due to variable illumination and environmental, atmospheric and temporal conditions. Ignoring these variations will introduce errors in hyperspectral data analysis. A number of approaches that deal with endmember variability have been investigated in recent years to improve better endmember finding and spectral unmixing. In this chapter, reasons for endmember variability are explained and a review and literature survey of previous works is conducted. At the end, we briefly discuss two new methods to be developed in this dissertation and their advantages over existing endmember variability techniques.

2.1 Reasons for Endmember Variability

In recently years, hyperspectral image analysis has been used to solve a wide range of applications in remote sensing including planetary exploration, environmental monitoring and target detection [1]. Two major areas are of particular interest. One is endmember finding and the other is spectral unmixing. Given a hyperspectral image, endmember finding is the task of finding the spectral signatures of purity present in the scene. Spectral unmixing is a task that estimates the abundances of each endmember in a data sample vector. A general approach is to model pixels in a hyperspectral image as linear combinations of endmembers

$$\mathbf{x}_n = \sum_{j=1}^{p} \alpha_j \mathbf{e}_j + \boldsymbol{\varepsilon}, \quad n = 1, \cdots, N \tag{2.1}$$

where $\{\mathbf{e}_j\}_{j=1}^{p}$ are a set of p $L\times 1$-dimensional endmembers, $\mathbf{x}_n$ is a $L\times 1$ vector representing the spectral signature of the n^{th} pixel in a hyperspectral image, N is the total number of pixels in the image and $\boldsymbol{\varepsilon}$ is a measurement error or noise. More specifically two abundance constraints are generally imposed on the model (2.1), i.e., Abundance Sum-to-one constraint (ASC) $\sum_{j=1}^{p}\alpha_j = 1$ and Abundance Non-negative Constraint (ANC) $\alpha_j \geq 0$.

In hyperspectral image analysis, the first task is to determine p or how many endmembers are present in the scene. It is defined as virtual dimensionality (VD) and determination of VD is very challenging and extremely difficult. Once VD is determined, finding these spectral signatures in the hyperspectral image is the next task. A number of algorithms including Automatic Target Generating Process (ATGP) [11], Simplex Growing Algorithm (SGA) [12] and N-FINDR [13] can be used to accomplish this task. After finding these signatures in the scene, Fully Constrained Least Square (FCLS) algorithm [14] is used to unmix the pixel in the hyperspectral image.

Although (2.1) is extensively used in the literature, the linear mixing model lacks the ability of representing the spectral variability of endmembers in a scene. Endmember spectral signatures are represented as single points in a high-dimensional space. The above-mentioned endmember finding algorithms, ATGP, SGA and N-FINDR are all single endmember-based techniques.

The spectral signature of a material varies within hyperspectral data due to a number of reasons including environmental, atmospheric and temporal factors. One major

source of spectral variability results from variation due to illumination conditions [15, 16]. Variation in topography and surface roughness, which leads to varying levels of shadowed and brightly lit regions, causes changes in illumination. Differences in the architecture of plant canopies, changes in the distribution of leaf orientation in vegetated regions, varying building structure and layout in urban areas cause different illumination levels and areas of shade. Although accurate digital terrain-elevation models and photometric information for an area can help remove some of the effects of illumination, this information, in general, is unavailable for a scene and if it were known, would require significant computationally intensive preprocessing. Another significant source that causes spectral variations come from changing atmospheric conditions. Many atmospheric gases, including oxygen, ozone, carbon monoxide and carbon dioxide, have strong absorption features or scattering characteristics in a number of wavelengths throughout the electromagnetic spectrum [5]. Although a number of approaches have been developed for this attempt to remove the atmospheric effects from hyperspectral data, some of the spectral variability due to atmospheric conditions may not be still eliminated. As discussed by Gao et al. [17], many of these approaches may not accurately account for nitrogen dioxide levels in the atmosphere, which can be extremely high in urban areas, or measured radiance from a pixel may be modified by the radiance of neighboring pixels due to scattering of solar radiation by atmospheric molecules.

Despite the fact that spectral variability is acknowledges from these sources, unmixing and endmember finding generally do not account for spectral variability anyway. As a result, errors resulting from inaccurate endmember representation will

be propagated thorough analysis. Proportion values can be estimated incorrectly with inaccurate endmember representations. To avoid these errors and to represent spectral variability during analysis, a number of spectral unmixing and endmember finding algorithms that incorporate spectral variability have been developed in [10]. A significant improvement in abundance estimation is shown in [5] with algorithms accounting for endmember variability.

Methods that account for endmember spectral variability can be organized into two general categories based on variability in endmember representation. One is to consider endmembers as sets, and the other is to consider endmembers as statistical distributions, each of which will be reviewed as follows.

2.2 Endmembers as Sets

2.2.1Multiple Endmember Mixture Analysis and Variants

In Multiple Endmember Mixture analysis, there are several spectral unmixing methods that can be used to estimat abundance values by exhaustively searching a given spectral library for endmembers whose corresponding estimated proportion values satisfy some criteria. The very first representative of these algorithms is multiple endmember spectral mixture analysis (MESMA) algorithm [18]. MESMA estimates the abundance for an input pixel by searching the endmembers for which abundance values are found that satisfy three conditions as follows.

1. The root mean square (rms) error between the input pixel and its reconstruction using the endmembers and their abundances is below a prescribed threshold.

2. The rms error for contiguous spectral bands is below a prescribed threshold.
3. The estimated abundance values are within a prescribed range.

As noted above, all of the three conditions require prescribed threshold, which could raise the following questions.

1. Should a prescribed threshold be the same for the same known spectral library?
2. Should different endmember classes have the same prescribed threshold?
3. Is there any common threshold that can be used in general with different images?

Besides the drawbacks of a prescribed threshold, MESMA also suffers from computational inefficiency. The algorithm allows each input pixel from a hyperspectral scene to be represented using a unique collection of endmember spectra from a spectral library. Therefore, even if the number of endmembers used for unmixing is limited for one input pixel, the full image can be mapped using many more endmembers. With a large spectral library, there will be a considerable number of endmembers to choose from, which leads extreme computational inefficiency.

In order to mitigate these difficulties, a number of extensions to MESMA have been developed in recent years.

There are a number of variants derived from MESMA. One is multiple-endmember linear spectral unmixing (MELSUM) method which uses a criterion relaxed to only identify nonnegative abundances while not requiring sum-to-one constrains [19]. Another is Bayesian spectral mixture analysis (BSMA) method which obtains the final proportion value of a material in a pixel through a weighted sum of the

abundance values found in all combinations, where the weights are proportional to the probabilities of each endmember deduced from a spectral library [17]. Rather than conducting an exhaustive search, randomly selecting endmembers from the spectral library to unmix the hyperspectral scene over several times is also a strategy used in automated Monte Carlo unmixing (AutoMCU) method [20] where the abundance value of each pixel are obtained by using their mean and standard deviation associated with each material from the several runs. In addition to improving computational efficiency, AutoMCU is able to quantify explicitly the abundance indeterminacy.

Although MESMA-based variants improve computational efficiency, none of these methods address an inherent issue arising in spectral variability using an appropriate spectral library. In practice, we cannot guarantee that such a spectral library is always available and accurate.

2.2.2 Endmember Bundles

Researchers can obtain spectral libraries using laboratory measurements of materials of interest or manual endmember identification from the imagery under study or data previously acquired. However, spectral libraries or the expertise are generally required for identifying spectral signatures of various materials. Therefore, approaches to autonomously estimate endmember classes and perform spectral unmixing of input data are needed.

In [6], a semiautomatic endmember set estimation technique is proposed. Endmember seeds are manually selected and an endmember class grows by identifying the data

pixels that have high correlation coefficients with the seeds. The physically meaningful range for correlation coefficients is from zero to one. The semiautomatic method is also based on prescribed thresholds. Therefore, it also suffers from the same issue such as how to determine the threshold and how the prescribed threshold is applied for all of endmember classes.

In order to avoid using the prescribed threshold, a subsequent work [10] by Somer et al. developed a fully automatic approach for building endmembers classes by repeatedly applying a standard endmember finding algorithm, such as automatic target generation process (ATGP), to a randomly selected portion of the input data. The endmember classes are obtained by grouping all of the found endmembers into p clusters using the K-means clustering algorithm. However, this automatic approach has its own drawbacks. Different numbers of portions, randomness of portions of the input data and randomness of K-mean method would lead to very different endmember classeses results. Additionally, pixel spectra that are found at one spatial subset may be partially mixed at the global scale which will adversely affect clustering.

2.2.3 Band Selection Weighting and Transformation

Band selection or weighting is another approach to address spectral variability. In this method, wavelengths with minimum spectral variability are found to be primarily used for spectral unmixing. Extending this concept, one could find spectral transformations that transform the input hyperspectral data into a space that minimizes the effect of spectral variability. In particular, the Fisher discriminant

approach (FDA) for spectral unmixing learns a transformation for the spectral signature elements to minimize the scatter within endmember sets and maximize the scatter among them prior to the estimation of abundance values [21] to avoid unmixing confusion [22]. The goal of FDA is to estimate the transformation projection matrix, $\hat{\mathbf{W}}$

$$\hat{\mathbf{W}} = \arg\max_{\mathrm{w}} \frac{|\mathbf{W}^T \mathbf{S}_b \mathrm{W}|}{|\mathbf{W}^T \mathbf{S}_w \mathrm{W}|} \tag{2.2}$$

where $\mathbf{S}_b$ and $\mathbf{S}_w$ are between- and within-class dispersion matrices [23]. Each pixel is transformed by $\hat{\mathbf{W}}$ before estimating abundance values. An effective transformation matrix will make the transformed elements within the same set nearly identical to each other and any one of the transformed elements from each of the endmember sets can be used to estimate the corresponding abundance values for that material. However, the noise correlation will affect the transformation and in the transformed space, some subtle spectral variability will be removed.

2.2.4 Support Vector Machine Unmixing

In [24, 25], researchers have developed a number of approaches using support vector machines (SVMs) for spectral unmixing while addressing spectral variability. In SVMs, a two-class classifier is commonly trained in a supervised fashion. Given a training set, $\{(\mathrm{x}_1, \mathrm{y}_1), (\mathrm{x}_2, \mathrm{y}_2), \cdots, (\mathrm{x}_N, \mathrm{y}_N)\}$ where x_i is the i^{th} data point and $y_i \in \{0,1\}$ is the desired class label, an SVM learns a hyper plane that separates two classes. In hyperspectral unmixing, two SVM-classifier classes are considered as the favorable class containing pixels from mixing endmembers at specific proportion

values versus the unfavorable class of those coming from other abundance choice. To discretize the solution space of unmixed abundances to a finite number of candidates is the first step for SVM-based unmixing. For example, if two materials are under consideration, with an abundance resolution of 0.1 for each material, nine proportion solution candidates $[(0.9,0.1), (0.8,0.2), \cdots, (0.1,0.9)]$ are generated. For each candidate, synthesized pixel data are created by drawing elements from endmember sets and mixing them using the proportion candidate. In the following, a number of SVMs are trained, one for each solution candidate, by labeling the synthesized data of the selected candidate as one and the other as zero. For an unknown pixel to be unmixed, all the SVM classifiers are used and the SVM that gives the largest classification margin will identify the corresponding candidate as the unmixed abundance solutions.

One advantage of SVM unmixing is that spectral variability is automatically taken care of when creating synthesized data for SVM training. However, only a finite and possibly limited number of abundance proportion choices can be considered due to the discretization of proportion values.

2.2.5 Sparse and Local Unmixing

In the sparse unmixing approach for endmember variability by Castrodad et al. [27], endmember bundle estimation is conducted first and then unmixing on a training data set. For example, suppose that N_m pixels are composed purely of material m denoted by the $L \times N_m$ matrix $\mathbf{X}_m$. The dictionary elements representing the endmember set of

material m , represented by the columns of the matrix $\tilde{\mathbf{E}}_m$, are obtained by minimizing the objective function

$$R(\tilde{\mathbf{E}}_m, \mathbf{p}_m) = \left\| \mathbf{X}_m - \tilde{\mathbf{E}}_m \mathbf{p}_m \right\|_F^2 + \lambda_S \left\| \mathbf{p}_m \mathbf{I} \right\|_1 \tag{2.3}$$

where $\mathbf{p}_m$ represents the abundance values for $\tilde{\mathbf{E}}_m$, $\mathbf{I}$ is the identity matrix and λ_S is a fixed regularization parameter. The N_m pixels in $\mathbf{X}_m$ are given from some training data such as hand selected from a ground truth data set. The second term in the objective function introduces sparseness. The minimization process could be achieved by Gauss-Seidel iteration [27].

Local unmixing (LU) extracts endmember sets with elements identified in spatial neighborhoods across the hyperspectral image [28, 29]. Local endmembers are found manually or using some endmember finding method. The found local endmembers are clustered using spectral angle criteria to form endmember bundles.

In sparse and local unmixing, regional specific endmember bundles are allowed. However, the assumption about sparsity and spatial constrain may be not valid in all the situations.

2.2.6 Multiscaled-band Partitioning method

Unlike partitioning a given hyperspectral image spatially, researchers in [30] partitions the image spectrally using multiscaled-band partitioning method. Endmember finding algorithms are applied to spectral subsets of the original bands and therefore it extracts the highest amount of information existing in the spectral domain using multiscaled-band partitioning at various band scales from broad and narrower intervals. The k-means method is then used to cluster found endmembers.

In this method, both inter- and intra-class variability are taken into consideration. However, it may generate different results using different number of scales and additionally computational complexity is inevitable.

2.3 Endmembers as distributions

An alternative to the set-based approach to addressing spectral variability in a material is the use of a multivariate statistical distribution. When endmembers are represented as statistical distributions, a sample from these distributions can be viewed as a possible variation.

2.3.1 Bayesian approach

"A number of methods make use of a Bayesian approach for endmember finding and spectral unmixing. When the endmember distributions are completely specified, joint estimation of endmembers and proportions, such as Bayesian source separation and nonnegative matrix factorization, can also provide endmember variations in spectral unmixing [30, 31]. Such an approach requires the exact knowledge of the endmember distribution."

2.3.2 Normal compositional model

"As the most prominent statistical distribution for endmembers, an expectation-maximization algorithm using normal compositional model (NCM) is proposed to iteratively update abundance values for every input pixel as well as the mean and variance parameter values for each endmember distribution until convergence is reached [32]. Eches et al. [33] developed a Markov chain Monte Carlo (MCMC)

sampling approach for estimating abundance values and the endmember distribution covariance using NCM under the assumption that the mean values of the endmembers are known. ”

2.3.3 Beta compositional model

“As an alternative compositional model, the beta compositional model (BCM) is developed. Under the BCM, the input data points are random variables distributed according to a convex combination of beta random variables. The motivation for the use of the beta distribution is that the values are constrained to the range from zero to one, which was a physically meaningful range for endmember reflectance values. The spectral unmixing method based on BCM developed in [34], assumes known endmember parameter values and estimates abundance values for input pixels using an approximation to the BCM. ”

2.3.4 Methods of higher moments

“Rather than defining a fixed parametric form for each endmember distribution, [35] proposed a method to estimate abundance value of an input data by minimizing the squared difference between the first and second moments of the estimated convex combination of the endmember values and those of the input data, provided that the known values for the first and second moments of each endmember distribution are given. An advantage of this approach is that the full parametric form of each endmember distribution does not need to be specified and instead only the first and second moments of each endmember distribution are needed.”

2.4 Proposed Methods

This dissertation investigates two completely different approaches from existing methods briefly reviewed above, to deal with endmember class problems. One is clustering-based algorithms for finding endmember classes and endmembers. In this method, endmember finding algorithms such as automatic target generation process (ATGP), unsupervised non-negativity constrained least squares (UNCLS) and unsupervised fully constrained least squares (UFCLS) [36] are used to find seed endmembers and an endmember class finding algorithm using half way distance is then used to find endmember class for each seed endmember. There is no prescribed threshold need to be determined and each endmember class has its own variability. The other method is Fisher's ratio-based approach to finding endmember classes and endmembers. This method defines two types of endmember variability, Between endmember variability (BEV) and Within endmember variability (WEV). A new criteria endmember variability ratio (EVR) is defined as the ratio of BEV and WEV and different versions of algorithms including SuCcessive, SeQuential and Growing are developed to maximize EVR.

Table 2.1 A review of Endmember Variability

No.	Category	Ideas	Papers	Year	Authors	Need Spectral Library or need training data?	Need threshold?	Advantages	Drawbacks
1	Endmembers as sets	Multiple Endmember Spectral Mixture Analysis (MESMA)	Mapping chaparral in the Santa Monica mountains using multiple endmember spectral mixture model	1998	D. Roberts	Yes	Yes	Direct and straightforward	Computationally inefficient; Need to determine threshold in practice;
2			Analysis of OMEGA/Mars Express data hyperspectral data using a multiple-endmember linear spectral unmixing model (MELSUM): Methodology and first results	2008	J.-Ph. Combe	Yes	Yes	Only consider nonnegative abundances	
3			Spectral mixture analysis for subpixel vegetation fractions in the urban environment: How to incorporate endmember variability?	2005	C. Song	Yes	Yes	Weigthed abundaces	
4			Scale dependence of biophysical structure in deforested areas bordering the Tapajós national forest, Central Amazon	2003	G. Asner	Yes	Yes	Unmix with randomly selected endmembers from library	
5		Band Selection, Weighting and transformation	A novel approach based on Fisher discriminant null space for decomposition of mixed pixels in hyperspectral imagery	2010	J. Jin	Yes	No	Computionally efficient	Information condensed space; noise may affect
6		Support Vector Mahcine Unmixing	A novel technique for subpixel image classification based on support vector machine	2010	F. Bovolo	Yes	No	Spectral variability automatically taken care of	A finite number of abundance proportion choices
7			SVM-based unmixing-to-classification conversion for hyperspectral abundance quantification	2011	F. Mianji	Yes	No		
8		Endmember Bundles	Endmember bundles: A new approach to incorporating endmember variability into spectral mixture analysis	2000	C. A. Bateson	No	No	Easy to understand; easy to apply	Results depend on endmember seed and need to determine threshold
9			Automated extraction of imagebased endmember bundles for improved spectral unmixing	2012	B. Somer	No	No	Fully automatic	Results depend random partitions of input image; may include mixed pixels in sub image
10		Sparse and local unmixing	Learning discriminative sparse representations for modeling, source separation, and mapping of hyperspectral imagery	2011	A. Castrodad	Yes	No	Data point or region specific endmember bundles allowed	Enforce sparsity and spatial assumptions that may be invalid
11			Spatially adaptive hyperspectral unmixing	2011	K. Canham	Yes	No		
12		Multiplescaled-band Partitioning	A New Approach for Endmember Extraction and Clustering Addressing Inter- and Intra-Class Variability via Multiscaled-Band Partitioning	2015	C. Andreou	No	No	Spectral variability taken care of by multiscaled-band partitioning	Complex and computationally inefficient
13		Spatial and spectral methods	An Image-Base Endmember Bundle Extraction Algorithm Using Both Spatial and Spectral Information	2015	M. Xu	No	Yes	Spatial and spectral information	Results depend random partitions of input image; may include mixed pixels in sub image; thresholds need for homogeneity index
14	Endmembers as distributions	Bayesian source separation	Separation of non-negative mixture of non-negative sources using a Bayesian approach and MCMC sampling	2006	S. Moussaoui	Yes	No	Joint estimation of endmembers and abundances	Fully specifications of endmember distributions needed; physically unrealistic endmembers may be allowed
15			Joint Bayesian endmember extraction and linear unmixing for hyperspectral imagery	2009	N. Dobigeon	Yes	No		
16			A new Bayesian unmixing algorithm for hyperspectral images mitigating endmember variability	2015	A. Halimi	Yes	No		
17			Unmixing multitemporal hyperspectral images accounting for endmember variability	2015	A. Halimi	No	No	Spatial information	
18		Normal compositional model	Application of the normal compositional model to the analysis of hyperspectral imagery	2003	D. Stein	No	No	Efficient use of Gaussian distribution	Physically unrealistic endmembers included; covariance between bands not addressed
19		Beta compositional model	Spectral unmixing using the beta compositional model	2013	A. Zare	No	No	Relectance values constrained to physically meaningful range; able to represent skew in endmember distributions	Detailed specifications of endmembers distributions required
20		Methods of higher moments	"Mixture models with higher order moments	1997	P. Bosdogianni	No	No	Simple form of address any endmember distribution	Accuracy limited by number of moments used; endmember moments needed in advance

Chapter 3 Clustering-Based Algorithms for Finding Endmember Classes and Endmembers

3.1 Introduction

As mentioned in Chapter 1, endmember variability needs to be taken into account when it comes to endmember finding. In this chapter, an endmember variability algorithm (EVA) is developed to solve the endmember class problem in a totally unsupervised fashion. In EVA, endmember finding algorithms such as automatic target generation process (ATGP), unsupervised non-negativity constrained least squares (UNCLS) and unsupervised fully constrained least squares (UFCLS) are used to find seed endmembers and algorithm for finding endmember classes (AFEC) is used to find an endmember class for each seed endmember.

3.2 A Preprocessing Step-Pixel Purity Index

The Pixel Purity Index (PPI) [37] is one of the most popular and widely used algorithms for finding endmembers in hyperspectral imaging. Technically speaking, PPI is a convex geometry-based technique which looks for data samples with maximal or minimal orthogonal projections along certain directions of interestingness. In this dissertation, PPI is used as a preprocessing step. Specifically, data samples of interest are found by PPI so that we can work on fewer data samples instead of the entire image.

The PPI has been widely used for endmember finding. Let $\{\mathbf{r}_i\}_{i=1}^{N}$ be a given set of data sample vectors. For a given value of K (usually K is less than N) we now use a random generator to produce a set of K random unit vectors (in the sense they all have the amplitude of 1), referred to as skewers, $\{\mathbf{skewer}_k\}_{k=1}^{K}$ which cover K different random directions. All the data sample vectors $\{\mathbf{r}_i\}_{i=1}^{N}$ are then orthogonally projected on this randomly generated skewer set, $\{\mathbf{skewer}_k\}_{k=1}^{K}$. According to geometry of convexity, an endmember which is considered as a pure signature should occur at end points of some of these skewers with either maximum projection or minimum projection. For each sample vector $\mathbf{r}_i$ we further calculate the number of skewers, denoted by $N_{PPI}(\mathbf{r}_i)$ at which this particular sample vector occurs as an end point to tally the PPI count for $\mathbf{r}_i$. Fig 3.1 illustrates how the concept of the PPI works where three skewers. $\mathbf{skewer}_1$, $\mathbf{skewer}_2$, $\mathbf{skewer}_3$ are indicated by three random unit vectors, the sample vectors are shown by open circles and three endmembers $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$ by solid circles located at three vertices of the triangles and a cross "+" used to indicate a maximum or minimum projection of an endmember on a skewer.

Due to convexity, all the sample vectors inside the triangle should have their PPI counts = 0 in the sense that they are mixtures of the three endmembers at the vertices of the triangle indicated by dashed lines. It should be noted that a maximum or minimum that occurred at a skewer is one that yields the maximum or minimum value among all the sample vectors. Also, a projection can be positive or negative

depending upon whether the projection occur the same or opposite direction of a skewer.

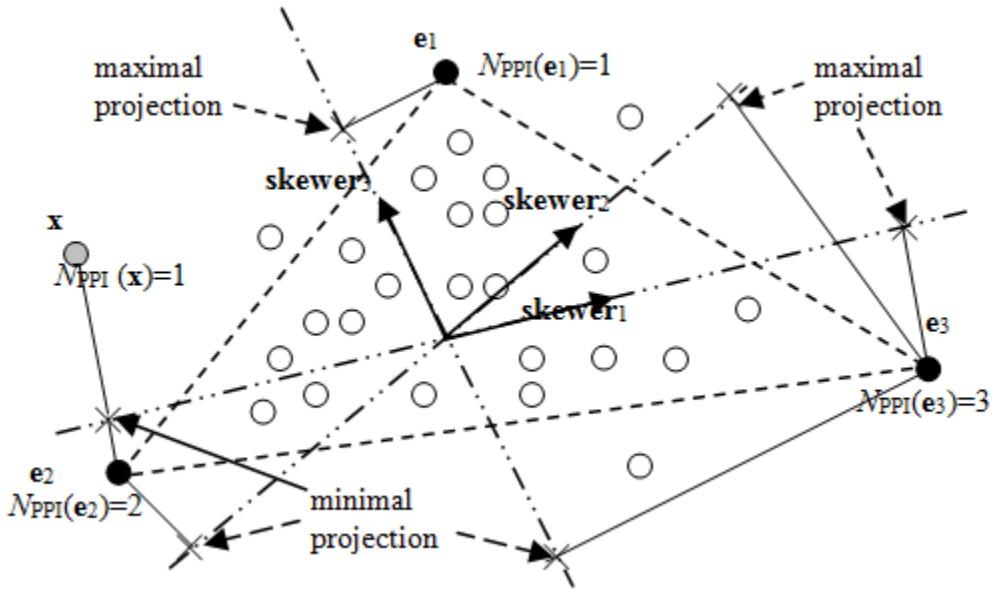


Figure 3.1 An illustration of PPI with three endmembers $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$

3.3 Half-Way Distance Clustering Algorithm for Finding Endmember Classes

In this section, three existing unsupervised endmember finding algorithms have been introduced and an endmember class finding algorithm is developed. The endmember variability algorithm, taking advantage of the endmember class finding algorithm, is used to solve endmember class problem in an iterative fashion.

3.3.1 Automatic Target Generation Process (ATGP)

The automatic target generation process (ATGP) has been found very useful and effective for unsupervised target detection. It performs a sequence of orthogonal

subspace projections to extract potential targets of interest. The detailed procedure for ATGP is described as follows.

Automatic Target Generation Process (*ATGP*)

1. Initial condition:

   Select an initial target pixel vector $\mathbf{t}_0 = arg\left\{max_{\mathbf{r}} \mathbf{r}^T\mathbf{r}\right\}$ and an error threshold ε.

   Set $k=1$ and $\mathbf{U}_0 = \left[\mathbf{t}_0\right]$.

2. At the k^{th} iteration, apply $\mathbf{P}_{\mathbf{U}_0}^{\perp}$ via $\mathbf{P}_{\mathbf{U}}^{\perp} = \mathbf{I} - \mathbf{U}\left(\mathbf{U}^T\mathbf{U}\right)^{-1}\mathbf{U}^T$ to all image pixels $\mathbf{r}$ in the image and find the k^{th} target $\mathbf{t}_k$ satisfying

$$\mathbf{t}_k = arg\left\{max_{\mathbf{r}}\left[\left(\mathbf{P}_{\mathbf{U}_{k-1}}^{\perp}\mathbf{r}\right)^T\left(\mathbf{P}_{\mathbf{U}_{k-1}}^{\perp}\mathbf{r}\right)\right]\right\} \tag{3.1}$$

   where $\mathbf{U}_{k-1} = \left[\mathbf{t}_1\mathbf{t}_2\cdots\mathbf{t}_{k-1}\right]$ is the target matrix generated at the $(k-1)^{th}$ stage.

3. Stopping rule:

   If $\mathbf{t}_k^T\mathbf{P}_{\mathbf{U}_{k-1}}^{\perp}\mathbf{t}_k < \varepsilon$, let $\mathbf{U}_k = \left[\mathbf{U}_{k-1}\mathbf{t}_k\right] = \left[\mathbf{t}_1\mathbf{t}_2\cdots\mathbf{t}_k\right]$ be the k^{th} target matrix, go to step 2 where $\mathbf{t}_k^T\mathbf{P}_{\mathbf{U}_{k-1}}^{\perp}\mathbf{t}_k$ is the orthogonal projection correlation index (OPCI). Otherwise, continue.

4. At this stage, ATGP is terminated and the final set of produced target pixel vectors comprises k target pixel vectors, $\left\{\mathbf{t}_0, \mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_k\right\} = \left\{\mathbf{t}_0\right\} \cup \left\{\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_{k-1}\right\}$ that were found by repeatedly using (3.1).

3.3.2 Unsupervised Non-negative Constrained Least Squares (UNCLS)

Linear spectral mixture analysis is a widely used approach to determine and quantify materials in remotely sensed imagery [Heinz, 2000]. Since every pixel is acquired by spectral bands at different wavelength, they can be represented by column vectors, and a hyperspectral image is actually an image cube. Suppose that l is the number of bands. Let $\mathbf{r}$ be a $l \times 1$ column pixel vector in a hyperspectral image where the bold face is used for vectors. Let $\mathbf{M}$ be a $l \times p$ material signature matrix denoted by $[\mathbf{m}_1, \mathbf{m}_2, \cdots, \mathbf{m}_p]$, where $\mathbf{m}_j$ is a $l \times 1$ column vector represented by the signature of the j^{th} signature resident in the image scene, and p is the number of signatures in the image scene. Let $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_p)^T$ be a $p \times 1$ abundance column vector associated with $\mathbf{r}$, where α_j denotes the abundance fraction of the j^{th} signature present in the pixel vector $\mathbf{r}$. A classical approach to solving mixed pixel classification problem is linear unmixing, which assumes that there are p materials in an image scene and the spectral signature of an image pixel vector $\mathbf{r}$ is linearly mixed by these p material signatures. In this case, the spectral signature of a pixel vector $\mathbf{r}$ can be represented by a linear regression model as follows:

$$\mathbf{r} = \mathbf{M}\alpha + \mathbf{n} \tag{3.2}$$

where $\mathbf{n}$ is noise or can be interpreted as a measurement error. A linear unmixing method attempts to unmix the unknown abundance fractions via an inverse of the linear mixture model specified by (3.2) so as to achieve the tasks of material discrimination, detection, classification, quantification, etc.

In general, an NCLS approach solves the following optimization problem

$$\text{Minimize } \text{LSE} = (\mathbf{M}\boldsymbol{\alpha} - \mathbf{r})^T (\mathbf{M}\boldsymbol{\alpha} - \mathbf{r}) \text{ subject to } \boldsymbol{\alpha} > 0 \tag{3.3}$$

The detailed procedure for NCLS is referred to [36].

The procedure for unsupervised NCLS (UNCLS) is summarized as follows.

Unsupervised NCLS (UNCLS) Algorithm

1. Initial condition:

   Select ε to be a prescribed error threshold and let $\mathbf{t}_0 = arg\{max_{\mathbf{r}}[\mathbf{r}^T\mathbf{r}]\}$ where $\mathbf{r}$ is all image pixel vectors. Let $k = 0$.

2. Let $k \leftarrow k+1$ and apply the NCLS algorithm with the signature matrix $\mathbf{M} = [\mathbf{t}_0, \mathbf{t}_1, \cdots, \mathbf{t}_{k-1}]$ to estimate the abundance fraction of $\mathbf{t}_0, \mathbf{t}_1, \cdots, \mathbf{t}_{k-1}$, $\hat{\alpha}_0^{(k)}(\mathbf{r}), \hat{\alpha}_1^{(k)}(\mathbf{r}), \cdots, \hat{\alpha}_{k-1}^{(k)}(\mathbf{r})$.

3. Find the maximum least squares error defined by

   $$max_{\mathbf{r}} \text{LSE}^{(k-1)}(\mathbf{r}) = max_{\mathbf{r}}\{(\mathbf{r} - [\textstyle\sum_{j=1}^{k-1} \hat{\alpha}_j^{(k)}(\mathbf{r})\mathbf{t}_j])^T (\mathbf{r} - [\textstyle\sum_{j=1}^{k-1} \hat{\alpha}_j^{(k)}(\mathbf{r})\mathbf{t}_j])\} \tag{3.4}$$

   If $\text{LSE}^{(k-1)}(\mathbf{r}) < \varepsilon$ for all $\mathbf{r}$, the algorithm stops; otherwise continue.

4. Find $\mathbf{t}_k = arg\{max_{\mathbf{r}} \text{LSE}^{(k-1)}(\mathbf{r})\}$. Go to step 2.

It is worth noting that the superscript (k) in $\hat{\alpha}_j^{(k)}(\mathbf{r})$ is a counter to indicate the number of iterations. It starts with $k = 1$. The subscript j in $\hat{\alpha}_0^{(k)}(\mathbf{r})$ starts with $j = 1$ and is the index of the j^{th} target signature $\mathbf{t}_j$ generated by UNCLS algorithm. Another comment is worthwhile. The UNCLS algorithm is primarily designed for unsupervised subpixel detection. It can be modified by replacing the NCLS algorithm in step 2 with the fully constrained least square (FCLS) algorithm, which is introduced in detail next.

3.3.3 Unsupervised Fully Constrained Least Squares (UFCLS)

In general, a FCLS approach solves the following optimization problem

$$\text{Minimize } \text{LSE} = (\mathbf{M}\boldsymbol{\alpha} - \mathbf{r})^T (\mathbf{M}\boldsymbol{\alpha} - \mathbf{r}) \text{ subject to } \boldsymbol{\alpha} > 0, \ \sum_{j=1}^{p} \alpha_j = 1 \tag{3.5}$$

Comparing to NCLS, FCLS has one more constraint abundance sum-to-one constraint (ASC). The detailed procedure of UFCLS is as follows.

1. Initial condition:

   Select ε to be a prescribed error threshold and let $\mathbf{t}_0 = arg\{max_{\mathbf{r}}[\mathbf{r}^T\mathbf{r}]\}$ where $\mathbf{r}$ is all image pixel vectors. Let $k = 0$.

2. Find $\mathbf{t}_1$ that yields the largest $\text{LSE}^{(0)}(\mathbf{r}) = (\mathbf{r} - \mathbf{t}_0)^T(\mathbf{r} - \mathbf{t}_0)$ $\mathbf{t}_1 = arg\{max_{\mathbf{r}} \text{LSE}^{(0)}(\mathbf{r})\}$.

3. Let $k \leftarrow k+1$ and apply the FCLS algorithm with the signature matrix $\mathbf{M} = [\mathbf{t}_0, \mathbf{t}_1, \cdots, \mathbf{t}_{k-1}]$ to estimate the abundance fraction of $\mathbf{t}_0, \mathbf{t}_1, \cdots, \mathbf{t}_{k-1}$, $\hat{\alpha}_0^{(k)}(\mathbf{r})$, $\hat{\alpha}_1^{(k)}(\mathbf{r})$, $\cdots$, $\hat{\alpha}_{k-1}^{(k)}(\mathbf{r})$.

4. Find the maximum least squares error defined by

   $$max_{\mathbf{r}} \text{LSE}^{(k-1)}(\mathbf{r}) = max_{\mathbf{r}}\{(\mathbf{r} - [\textstyle\sum_{j=1}^{k-1} \hat{\alpha}_j^{(k)}(\mathbf{r})\mathbf{t}_j])^T(\mathbf{r} - [\textstyle\sum_{j=1}^{k-1} \hat{\alpha}_j^{(k)}(\mathbf{r})\mathbf{t}_j])\} \tag{3.6}$$

   If $\text{LSE}^{(k-1)}(\mathbf{r}) < \varepsilon$ for all $\mathbf{r}$, the algorithm stops; otherwise continue.

5. Find $\mathbf{t}_k = arg\{max_{\mathbf{r}} \text{LSE}^{(k-1)}(\mathbf{r})\}$. Go to step 2.

3.3.4 Algorithm for Finding Endmember Classes

Finding endmember classes is very challenging as shown in [38]. Many approaches have been reported in the literature, most of which are either empirical or use prior knowledge. None of them can be implemented automatically in an unsupervised

fashion. This section presents a completely automatic algorithm to find endmember classes.

Algorithm for Finding Endmember Classes (AFEC)

1. Initial conditions:
   Input a set of endmembers, denoted by $\{\mathbf{e}_j\}_{j=1}^{p}$.

2. For each $\mathbf{e}_j$ for $1 \le j \le p$, find $D_j = d(\mathbf{e}_j, \bar{S}_j) = min_{\mathbf{e}_i \in \bar{S}_j} \{d(\mathbf{e}_j, \mathbf{e}_i)\}$ where $\bar{S}_j = \{\mathbf{e}_i\}_{i=1, i \ne j}^{p}$ and $d(\cdot,\cdot)$ is a distance measure which can be either Euclidean distance or Mahalanobis distance.

3. For the j^{th} endmember class EC_j for $1 \le j \le p$ we can use find EC_j as follows.

$$EC_j = \{\mathbf{r} \mid d(\mathbf{r}, \mathbf{e}_j) < D_j / 2\} \tag{3.7}$$

It should be noted that the threshold $D_j / 2$ in (3.7) chosen to be half way of minimum distance D_j is to avoid overlapping with other endmember classes. As a result, all such generated endmember classes are disjoint with no intersection.

Proof of all generated endmember classes by algorithm for finding endmember classes are disjoint without any intersection.

Suppose one pixel $\mathbf{r}$ belongs to both EC_1 and EC_2. So we have

$$d(\mathbf{r}, \mathbf{e}_1) < D_1 / 2 \le d(\mathbf{e}_1, \mathbf{e}_2) / 2 \tag{3.8}$$

$$d(\mathbf{r}, \mathbf{e}_2) < D_2 / 2 \le d(\mathbf{e}_1, \mathbf{e}_2) / 2 \tag{3.9}$$

Add both sides,

$$d(\mathbf{r}, \mathbf{e}_1) + d(\mathbf{r}, \mathbf{e}_2) < d(\mathbf{e}_1, \mathbf{e}_2) \tag{3.10}$$

which is contradicted with $d(\mathbf{r},\mathbf{e}_1)+d(\mathbf{r},\mathbf{e}_2)\geq d(\mathbf{e}_1,\mathbf{e}_2)$ (by triangle inequality property in metric space).

So $\mathbf{r}$ cannot be in both EC_1 and EC_2.

3.3.5 Endmember Variability Algorithm (EVA)

Assume that $\{\mathbf{e}_j\}_{j=1}^{p}$ is a set of desired endmembers and $\{EC_j\}_{j=1}^{p}$ are their corresponding endmember classes where EC_j is the spectral class specified by the j^{th} endmember $\mathbf{e}_j$ which is a set made up of $\{\mathbf{s}_i^j\}_{i=1}^{n_j}$ where $\mathbf{s}_i^j$ is an element in EC_j and considered to be a spectral variant of $\mathbf{e}_j$, and n_j is the total number of elements in EC_j.

Endmember Variability Algorithm (EVA)

1. Initial condition:

   a. Determine the value of *p*, number of endmembers required to be found. Use endmember initialization algorithms (EIA) such as Automatic Target Generation Process (ATGP) and Unsupervised Fully Constrained Least Squares (UFCLS) to produce a set of initial endmembers, $E^{(0)}=\{\mathbf{e}_j^{(0)}\}_{j=1}^{p}$

   b. Set $k = 1$.

2. Finding k^{th} endmember classes, $\{EC_j^{(k)}\}_{j=1}^{p}$:

   At the k^{th} iteration implement the algorithm for finding endmember classes, $\{EC_j^{(k)}\}_{j=1}^{p}$ using the $(k-1)^{th}$ set of endmembers $EC^{(k-1)}$.

3. Finding the k^{th} set of endmembers, $EC^{(k)}$:

   Calculate the mean of each endmember class, denoted by $\left\{\boldsymbol{\mu}_j^{(k)}\right\}_{j=1}^{p}$ where $\boldsymbol{\mu}_j^{(k)}$ is the mean of $EC_j^{(k)}$.

   a. Either let $\mathbf{e}_j^{(k)} = \boldsymbol{\mu}_j^{(k)}$ form $E^{(k)} = \left\{\mathbf{e}_j^{(k)}\right\}_{j=1}^{p}$ or

   b. Find the closest data samples, $\left\{\mathbf{t}_j^{(k)}\right\}$ to the class means $\left\{\boldsymbol{\mu}_j^{(k)}\right\}_{j=1}^{p}$ according to

$$\mathbf{t}_j^{(k)} = arg\left\{min_{\mathbf{r}} m(\mathbf{r}, \boldsymbol{\mu}_j^{(k)})\right\} \tag{3.11}$$

   with $m(\cdot,\cdot)$ is a spectral measure such as Euclidean distance, spectral angle mapper (SAM) or spectral information divergence (SID) [1]. Let $\mathbf{e}_j^{(k)} = \mathbf{t}_j^{(k)}$ and $EC^{(k)} = \left\{\mathbf{e}_j^{(k)}\right\}_{j=1}^{p}$

4. Sopping rule:

   If $EC^{(k)} = EC^{(k-1)}$, algorithm is terminated. Otherwise, continue.

5. Let $k \leftarrow k+1$ and go to step 2.

   Stopping Rule

   Due to endmember variability finding identical endmembers in both sets, $EC^{(k)} = EC^{(k-1)}$ may cause fluctuations in step 4 of EVA. In order to execute this stopping rule with flexibility we take advantage of an algorithm developed in [Li, 2014], called endmember identification algorithm and modify it for our purpose.

Stopping Rule for EVA

1. Assume that $EC^{(k)} = \left\{\mathbf{e}_j^{(k)}\right\}_{j=1}^{p}$ are the k^{th} generated set of p endmembers and $EC^{(k-1)} = \left\{\mathbf{e}_i^{(k-1)}\right\}_{i=1}^{p}$ is the $(k-1)^{th}$ generated set of p endmembers.

2. For each endmember $\mathbf{e}_j^{(k)}$, find an endmember class in $\left\{EC_i^{(k-1)}\right\}_{i=1}^{p}$ assigned to $\mathbf{e}_j^{(k)}$ by

$$\mathbf{e}_j^{(k)} \in EC_{j^*}^{(k-1)} \quad \Leftrightarrow \quad j^* = arg\left\{min_{1\le i\le p} SAM(\mathbf{e}_j^{(k)}, \mathbf{e}_i^{(k-1)})\right\} \tag{3.12}$$

where SAM is short for spectral angle mapper, which is defined as $SAM(\mathbf{s}_1, \mathbf{s}_2) = \cos^{-1}\left(\frac{\langle \mathbf{s}_1, \mathbf{s}_2 \rangle}{\|\mathbf{s}_1\|\|\mathbf{s}_2\|}\right)$.

3. For each endmember $\mathbf{e}_i^{(k-1)}$ in $E^{(k-1)} = \left\{\mathbf{e}_i^{(k-1)}\right\}_{i=1}^{p}$, find an endmember class in $E^{(k-1)} = \left\{\mathbf{e}_i^{(k-1)}\right\}_{i=1}^{p}$ to which it belongs by

$$\mathbf{e}_i^{(k-1)} \in EC_{i^*}^{(k)} \quad \Leftrightarrow i^* = arg\left\{min_{1\le j\le J} SAM(\mathbf{e}_j^{(k)}, \mathbf{e}_i^{(k-1)})\right\} \tag{3.13}$$

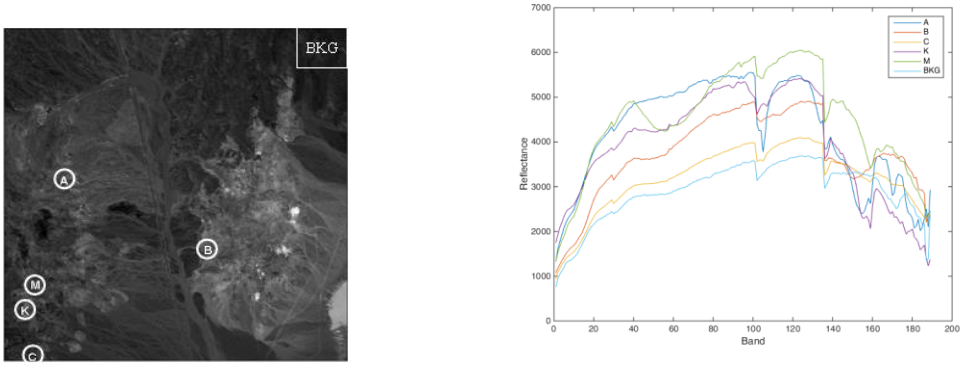
4. If for each endmember $\mathbf{e}_j^{(k)}$ in $EC^{(k)} = \left\{\mathbf{e}_j^{(k)}\right\}_{j=1}^{p}$, $\mathbf{e}_j^{(k)}$ can find its corresponding $\mathbf{e}_{j^*}^{(k-1)}$ in $EC^{(k-1)} = \left\{\mathbf{e}_i^{(k-1)}\right\}_{i=1}^{p}$ via (3.12) for some j^* and in the meantime, for each endmember in $EC^{(k-1)} = \left\{\mathbf{e}_i^{(k-1)}\right\}_{i=1}^{p}$, $\mathbf{e}_i^{(k-1)}$ can find its corresponding $\mathbf{e}_{i^*}^{(k)}$ in $EC^{(k)} = \left\{\mathbf{e}_j^{(k)}\right\}_{j=1}^{p}$ via (3.13) for some i^*, then $EC^{(k)} = EC^{(k-1)}$. Otherwise $EC^{(k)} \neq EC^{(k-1)}$.

3.4 Experiments

In the following experiments, the closest data sample to the class mean is used as the new class center. Endmember Variability Algorithm (EVA) stops when two consecutive runs are identical.

3.4.1 Synthetic Image Experiments

Since real images generally do not have complete ground truth about the endmembers, we must rely on synthetic images which are simulated by complete knowledge to conduct quantitative analysis for performance evaluation of various endmember extraction algorithms. Recently, several synthetic images developed in [39] can be used for this purpose. These synthetic images were custom-designed and simulated based on the Cuprite image data, which is available on the USGS website http://aviris.jpl.nasa.gov/. This scene is a 224-band image with size of 350×350 pixels and was collected over the Cuprite mining site, Nevada, in 1997. It is well understood mineralogically. As a result, a total of 189 bands were used for experiments where bands 1-3, 105-115 and 150-170 have been removed prior to the analysis due to water absorption and low SNR in those bands. Although there are more than five minerals in the data set, the ground truth available for this region only provides the locations of the pure pixels: Alunite (A), Buddingtonite (B), Calcite (C), Kaolinite (K) and Muscovite (M). The locations of these five pure minerals are labeled by A, B, C, K and M respectively and shown in Fig 3.1. Available from the image scene is a set of reflectance spectra shown in Fig 3.1 which will be used to simulate synthetic images.



(a) (b)

Figure 3.2 (a) Cuprite AVIRIS image scene; (b) Five mineral reflectance spectra and background signature which is the average of area BKG in the top right of (a)

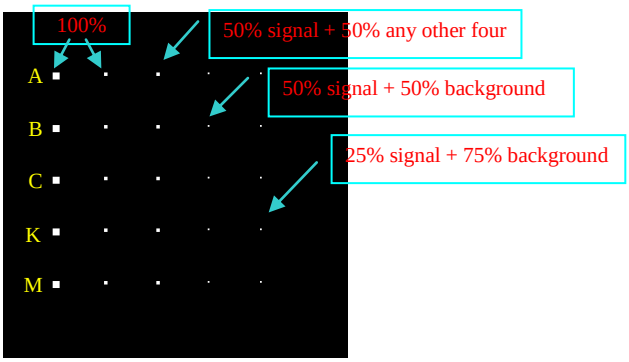


Figure 3.3 A set of 25 panels simulated by A,B,C,K,M

The synthetic image designed here simulates 25 panels shown in Fig 3.3 with 5 panels in each row simulated by the same mineral signature and 5 panels in each column having the same size.

Table 3.1 Mixing percent and pixels in each row and column in Fig 3.2

Mixing percent and pixels in each row and column		1st Column	2nd Column	3rd Column	4th Column	5th Column
		100% sig	100% sig	50% sig + 50% other four	50% sig + 50% background	25% sig + 75% background
1st Row	A	4×4	2×2	2×2	1×1	1×1
2nd Row	B	4×4	2×2	2×2	1×1	1×1
3rd Row	C	4×4	2×2	2×2	1×1	1×1
4th Row	K	4×4	2×2	2×2	1×1	1×1
5th Row	M	4×4	2×2	2×2	1×1	1×1

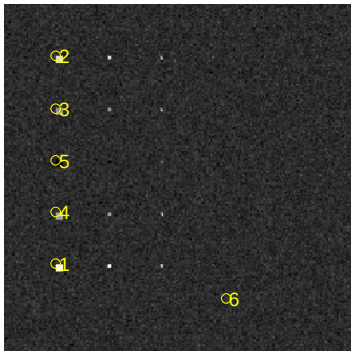
Among 25 panels are five 4×4 pure-pixel panels for each row in the 1st column and five 2×2 pure-pixel panels for each row in the 2nd column, the five 2×2-mixed pixel panels for each row in the 3rd column and both the five 1×1 subpixel panels for each row in the 4th column and the 5th column where the mixed and subpanel pixels were simulated according to legends in Fig 3.3. So, a total of 100 pure pixels (80 in the 1st column and 20 in 2nd column), referred to as endmember pixels were simulated in the data by the five endmembers, A, B, C, K, M. An area marked by "BKG" at the upper right corner of Fig 3.2(b) was selected to find its sample mean, i.e., the average of all pixel vectors within the area "BKG", denoted by **b** and plotted in Fig 3.2(b), to be used to simulate the background for image scene in Fig 3.3. The reason for this background selection is empirical since the selected area "BKG" seemed more homogeneous than other regions. Nevertheless, other areas can be also selected for the same purpose. This **b**-simulated image background was further corrupted by an additive noise to achieve a certain signal-to-noise ratio (SNR) which was defined as 50% signature (i.e., reflectance/radiance) divided by the standard deviation of the noise. Once target pixels and background are simulated, two types of target insertion can be designed to simulate experiments for various applications.

1. Scenario 1: TI Scenario

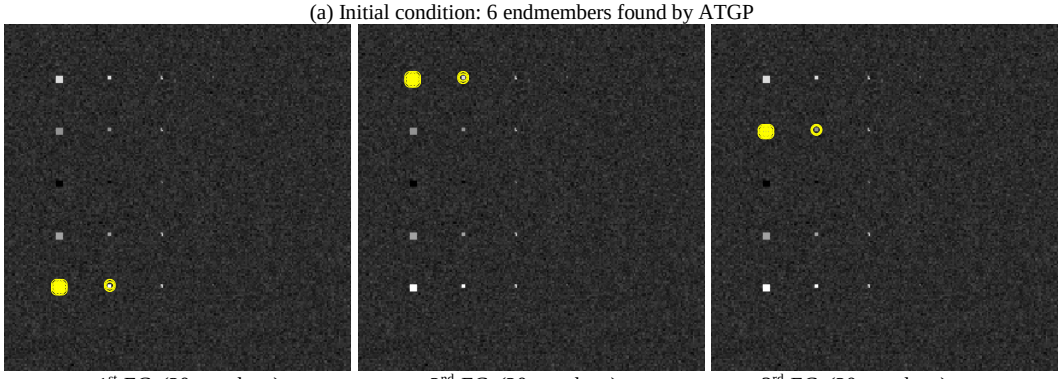
The first type of target insertion is Target Implantation (TI) which can be simulated by inserting clean target panels into the clean image background plus additive Gaussian noise by replacing their corresponding background pixels.

The value of VD is chosen to be 6 according to the ground truth where 5 mineral signatures plus a BKG signature are used to simulate the synthetic image scene. Fig.

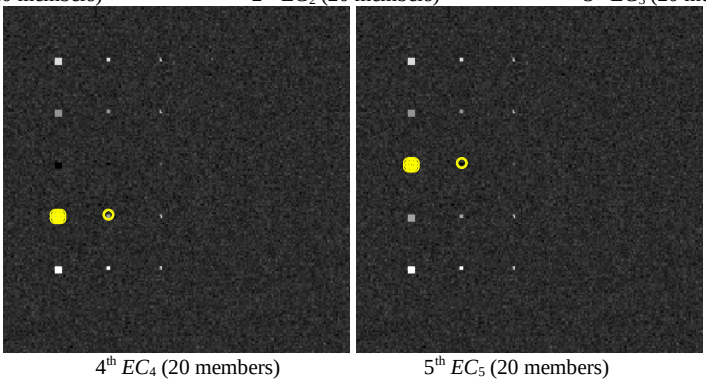
3.4(a) shows 6 initial endmembers generated by ATGP for TI and Fig. 3.4(b) shows the 5 endmember classes by the proposed EVA using the 6 initial signatures in Fig. 3.4(a) for TI. It should be noted that since we are not interested in BKG class, only 5 endmember classes corresponding to five mineral signatures are shown in Fig. 3.5. The algorithm is terminated at the 3rd run because the endmember classes generated at the 2nd and 3rd runs are the same where each endmember class indeed finds all its own corresponding pure panel signatures.



(a) Initial condition: 6 endmembers found by ATGP



1st EC_1 (20 members) 2nd EC_2 (20 members) 3rd EC_3 (20 members)

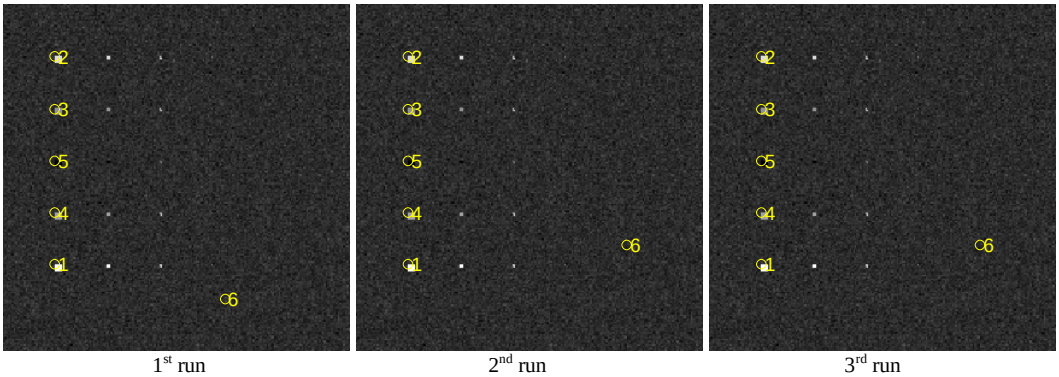


4th EC_4 (20 members) 5th EC_5 (20 members)

(b) Finding 5 endmember classes

Figure 3.4 TI: ATGP-generated endmembers and their endmember classes

Fig. 3.5 shows its final endmembers selected by EVA to represent each of endmember classes where these representatives happen to be the first panel pixels in each row that were identified by ATGP.



1st run 2nd run 3rd run

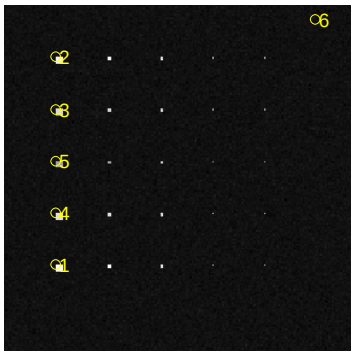
Figure 3.5 TI: final endmembers determined by EVA

Scenario 2: TE Scenario

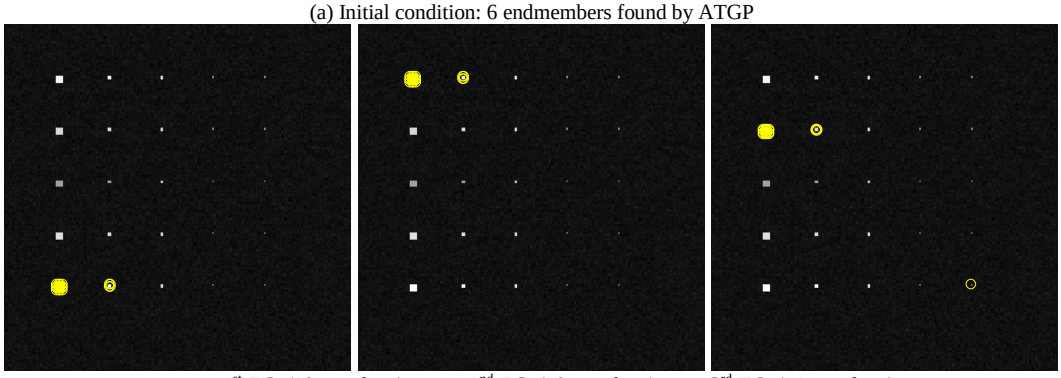
A second type of target insertion is Target Embeddedness (TE) which can be also simulated by embedding clean target panels into the clean image background plus additive Gaussian noise by superimposing target pixels over the background pixels.

Same experiments conducted for TI are also conducted for TE.

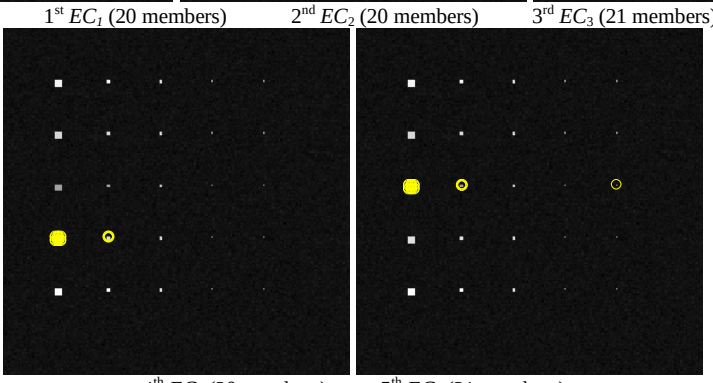
Fig. 3.6 and Fig. 3.7 shows two consecutive runs of endmembers (2^{nd} and 3^{rd}) are identical and EVA stops at the 3^{rd} run for TE data. If the stopping rule from endmember identification algorithm is applied, EVA stops at the 2^{nd} run. Although the 6^{th} endmember location changes in the 2^{nd} run, the endmember classes for 5 panel endmembers are the same as those shown in Fig 3.6.



(a) Initial condition: 6 endmembers found by ATGP



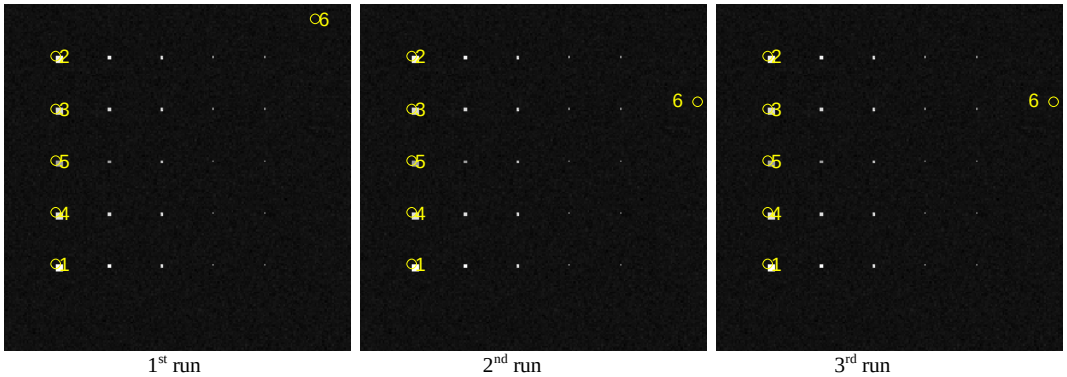
1st EC_1 (20 members) 2nd EC_2 (20 members) 3rd EC_3 (21 members)



4th EC_4 (20 members) 5th EC_5 (21 members)

(b) Five endmember classes

Figure 3.6 TE: ATGP generated endmembers and their endmember classes

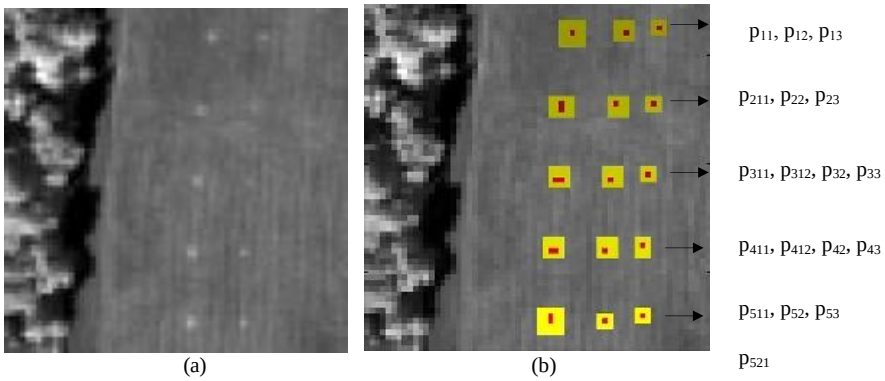


1st run 2nd run 3rd run

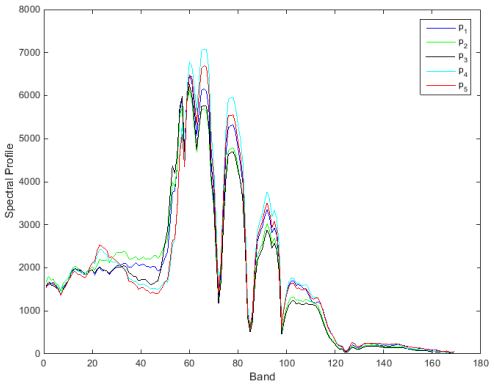
Figure 3.7 TE: 6 endmembers determined by EVA

3.4.2 Real Image Experiment

The Hyperspectral Digital Imagery Collection Experiment (HYDICE) image scene is shown in Fig. 3.8 which was collected in August 1995 from a flight altitude of 10000 feet with the ground sampling distance approximately 1.56 m [40]. It has size of 64×64 pixel vectors shown in Fig 3.8 (a) along with its ground truth provided in Fig 3.8 (b) where the center and boundary pixels of objects are highlighted by red and yellow respectively. It was acquired by 210 spectral bands with a spectral coverage from 0.4 μm to 2.5 μm. Low signal/high noise bands: bands 1-3 and bands 202-210; and water vapor absorption bands: bands 101-112 and bands 137-153 were removed. So, a total of 169 bands were used in the experiments. The spectral resolution of this image scene is 10 nm. Within the scene in Fig. 3.8 there is a large grass field and a forest on the left edge. Each element in this matrix is a square panel and denoted by p_{ij} with rows indexed by i and columns indexed by $j = 1, 2, 3$. For each row $i = 1, 2, \cdots, 5$, there are three panels painted by the same paint but with three different sizes.



(a) (b)

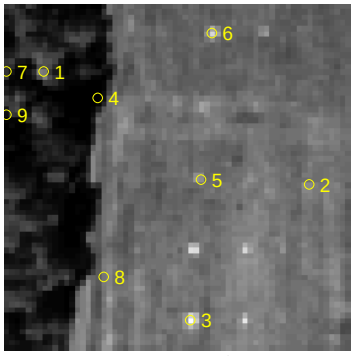


(c)

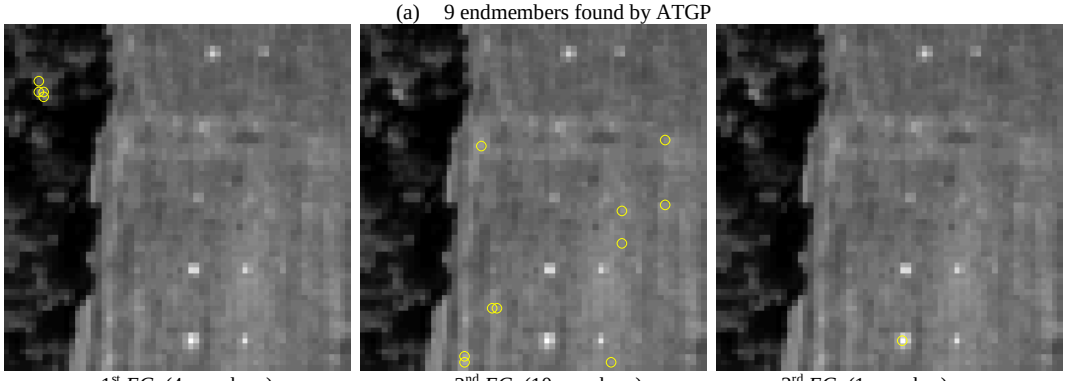
Figure 3.8 (a) A HYDICE panel scene which contains 15 panels; (b) Ground truth map of spatial locations of the 15 panels; (c) five panel signatures $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$, $\mathbf{p}_4$, $\mathbf{p}_5$

The sizes of the panels in the first, second and third columns are $3\,m \times 3\,m$, $2\,m \times 2\,m$ and $1\,m \times 1\,m$ respectively. Since the size of the panels in the third column is $1\,m \times 1\,m$, they cannot be seen visually from Fig 3.8 (a) because of their size less than the 1.56 m pixel resolution. For each column $j = 1,\ 2,\ 3$, the 5 panels have same sizes but in five different paint. However, it should be noted that the panels in rows 2 and 3 were made by the same material with two different paints. Similarly, it is also the case for panels in rows 4 and 5. Nevertheless, they were still considered as different panels but our experiments will demonstrate that detecting panels in row 5 (row 3) may also have effect on detection of panels in row 4 (row 2). The 1.56 m-spatial resolution of the image scene suggests that most of the 15 panels are one pixel in size except that the panels in the 1^{st} column with the 2^{nd}, 3^{rd}, 4^{th}, 5^{th} rows which are two-pixel panels, denoted by p_{211}, p_{221}, p_{311}, p_{312}, p_{411}, p_{412}, p_{511}, p_{521}. As a result, there are a total 19 panel pixels. Fig 3.8 (b) shows the precise spatial locations of these 19 panel pixels where red pixels (R pixels) are the panel center pixels and the pixels in yellow (Y pixels) are panel pixels mixed with the BKG. Fig

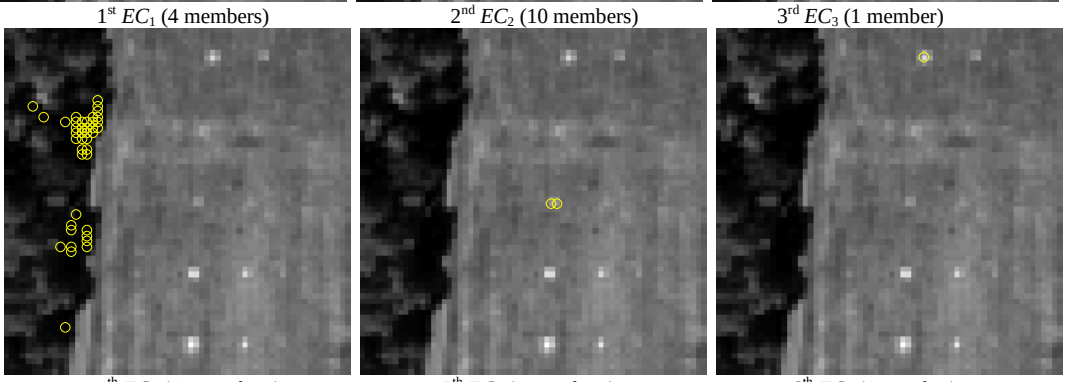
3.8 (c) shows the spectra of five panel signatures $\mathbf{p}_1$, $\mathbf{p}_2$, $\mathbf{p}_3$, $\mathbf{p}_4$, $\mathbf{p}_5$ obtained by averaging the center R panel pixels for each of five rows.



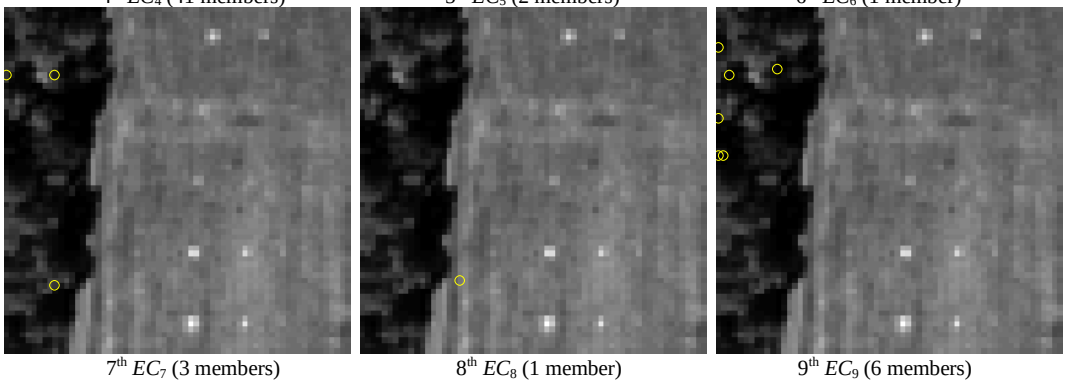
(a) 9 endmembers found by ATGP



1st EC_1 (4 members) 2nd EC_2 (10 members) 3rd EC_3 (1 member)



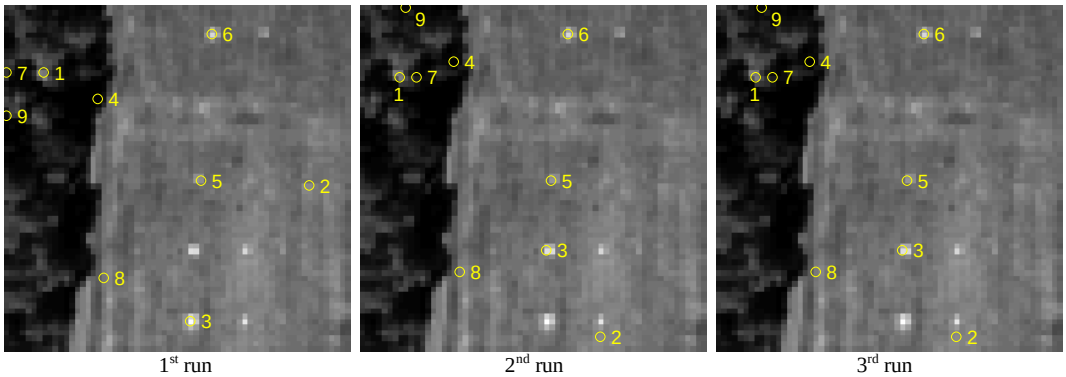
4th EC_4 (41 members) 5th EC_5 (2 members) 6th EC_6 (1 member)



7th EC_7 (3 members) 8th EC_8 (1 member) 9th EC_9 (6 members)

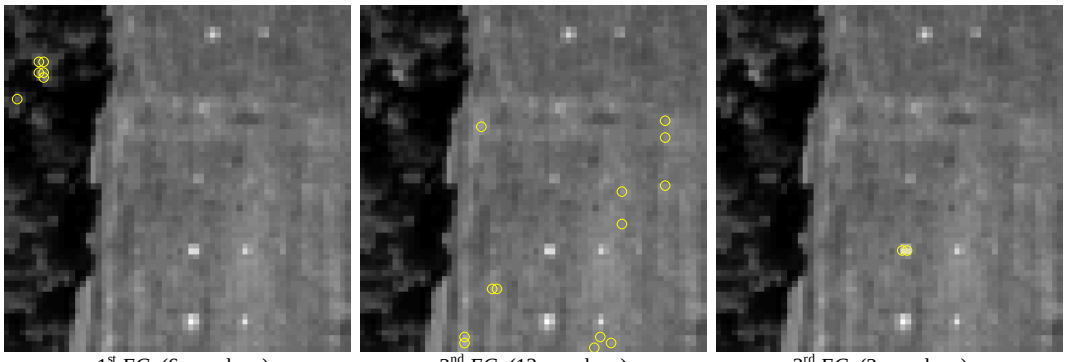
(b) Finding 9 endmember classes

Figure 3.9 HYDICE: ATGP-generated endmembers and their endmember classes

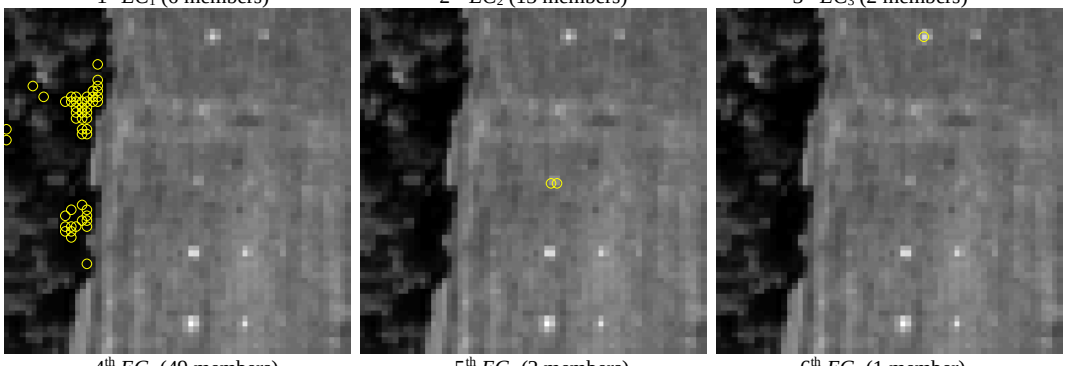


1st run 2nd run 3rd run

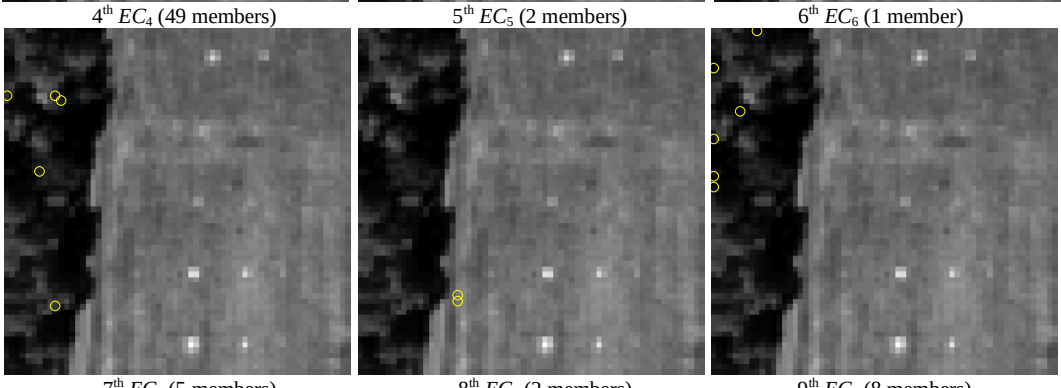
Figure 3.10 HYDICE: endmembers generated by EVA



1st EC_1 (6 members) 2nd EC_2 (13 members) 3rd EC_3 (2 members)



4th EC_4 (49 members) 5th EC_5 (2 members) 6th EC_6 (1 member)



7th EC_7 (5 members) 8th EC_8 (2 members) 9th EC_9 (8 members)

Figure 3.11 HYDICE: endmember classes of 3rd run by EVA

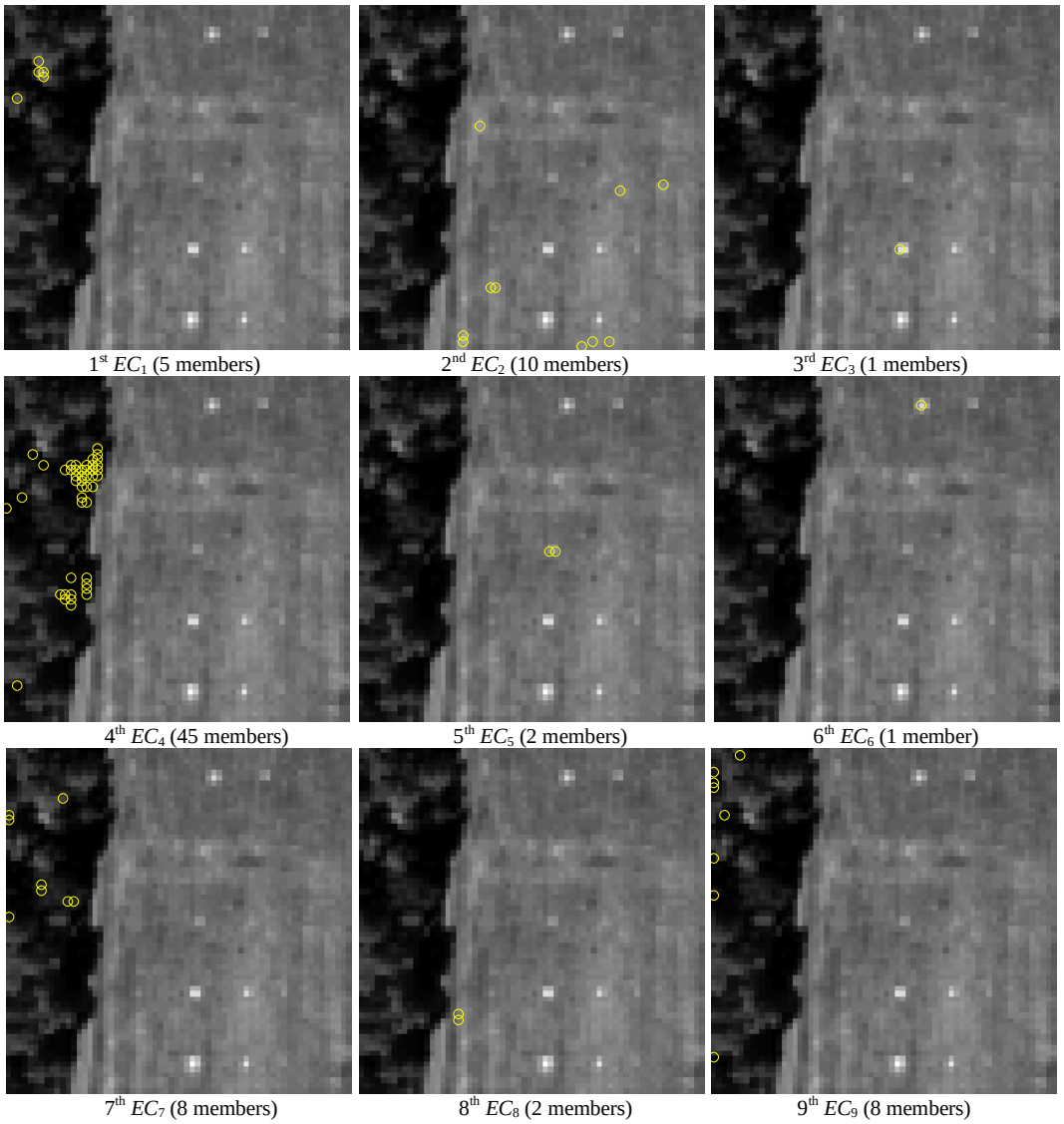
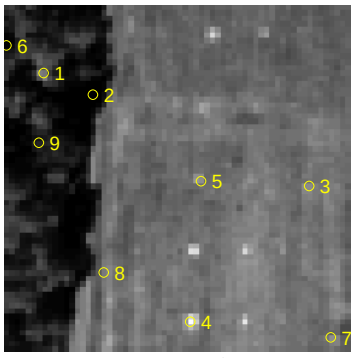


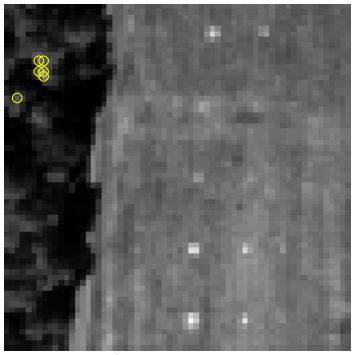
Figure 3.12 HYDICE: endmember classes by EVA using class means

Nine endmembers generated by ATGP and their corresponding endmember classes are shown in Fig 3.9. As shown in Fig 3.10, it takes EVA three runs to converge (two consecutive runs are the same). If the stopping rule from endmember identification algorithm is used, EVA stops at the 2nd run as shown in Fig 3.10. The final endmember classes are shown in Fig 3.11. Instead of using the closest data sample, endmember class mean can be also used as virtual signature. The final endmember classes are shown in Fig 3.12. Comparing results from Fig. 3.11 and Fig 3.12,

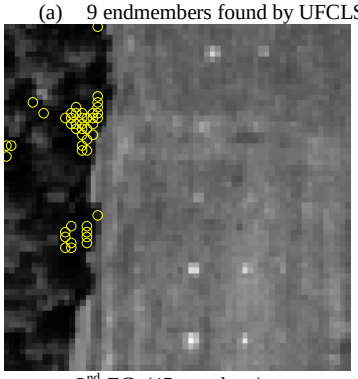
endmember classes results are almost the same. The only difference is that the third endmember class includes two members as shown in Fig. 3.11 while it only has only member as shown in Fig. 3.12 for panel pixels in the fourth row.



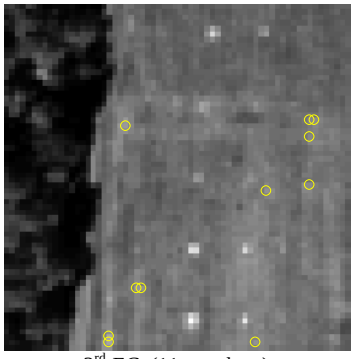
(a) 9 endmembers found by UFCLS



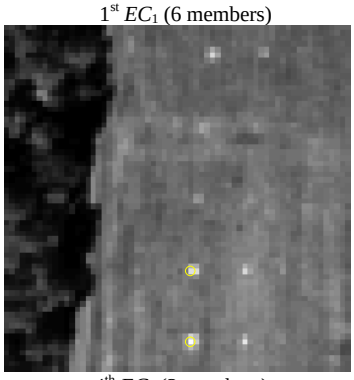
1st EC_1 (6 members)



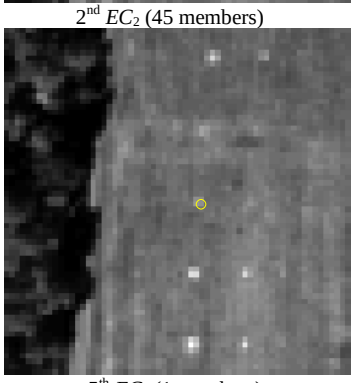
2nd EC_2 (45 members)



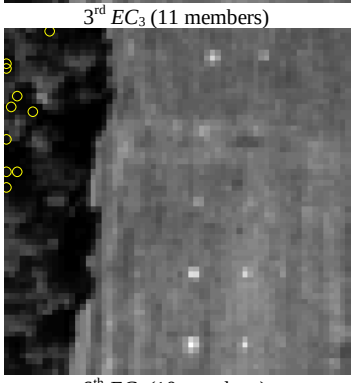
3rd EC_3 (11 members)



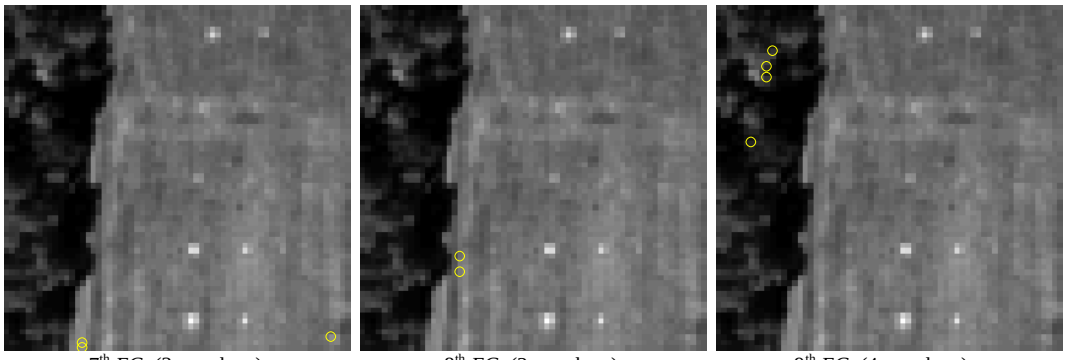
4th EC_4 (2 members)



5th EC_5 (1 members)



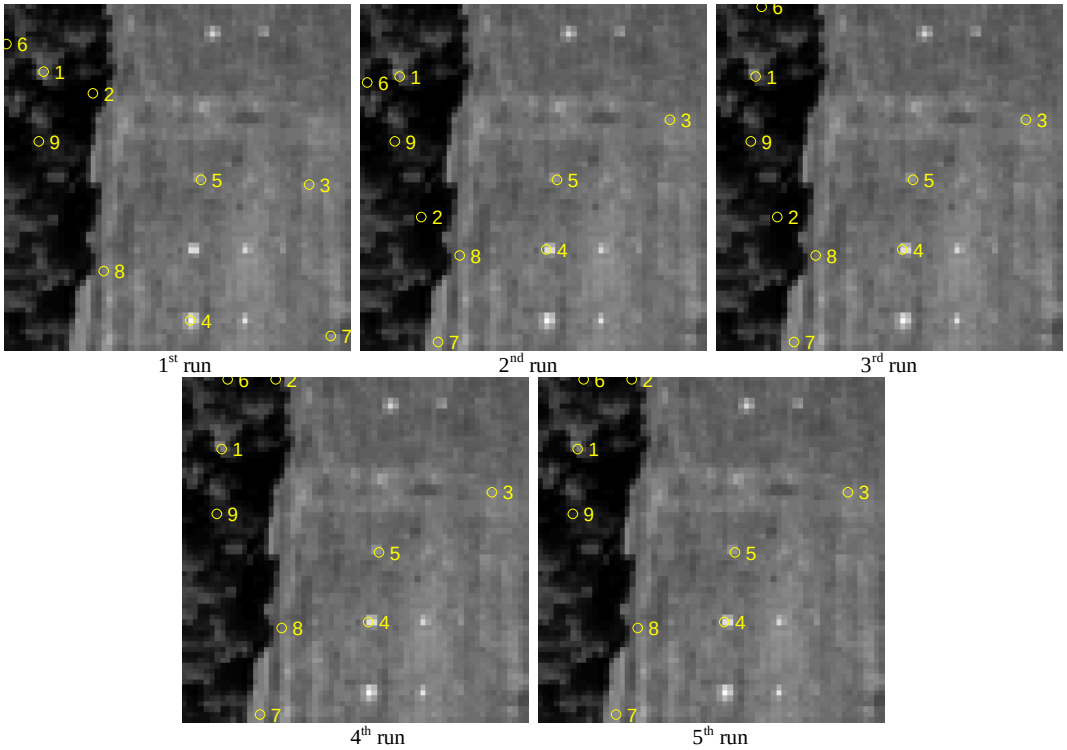
6th EC_6 (10 members)



7th EC_7 (3 members) 8th EC_8 (2 members) 9th EC_9 (4 members)

(b) Finding 9 endmember classes

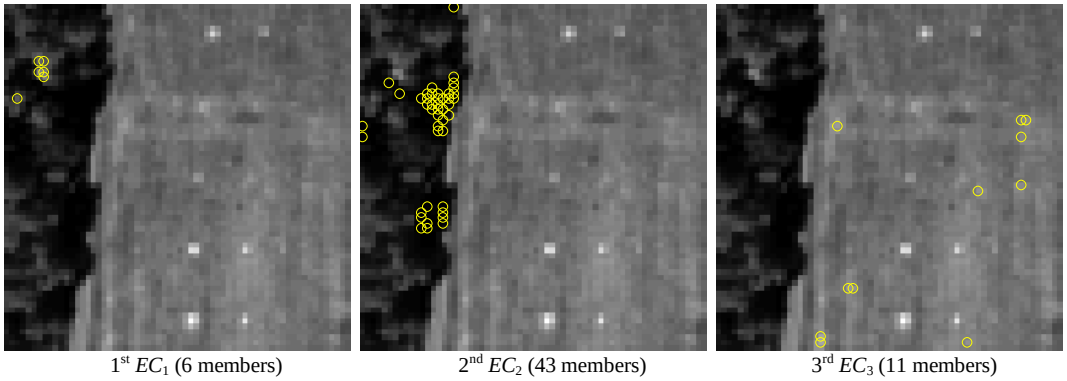
Figure 3.13 HYDICE: UFCLS-generated endmembers and their endmember classes



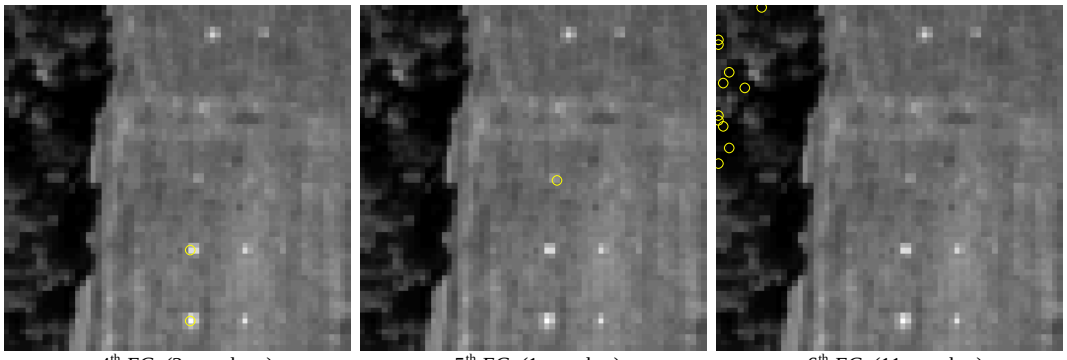
1st run 2nd run 3rd run

4th run 5th run

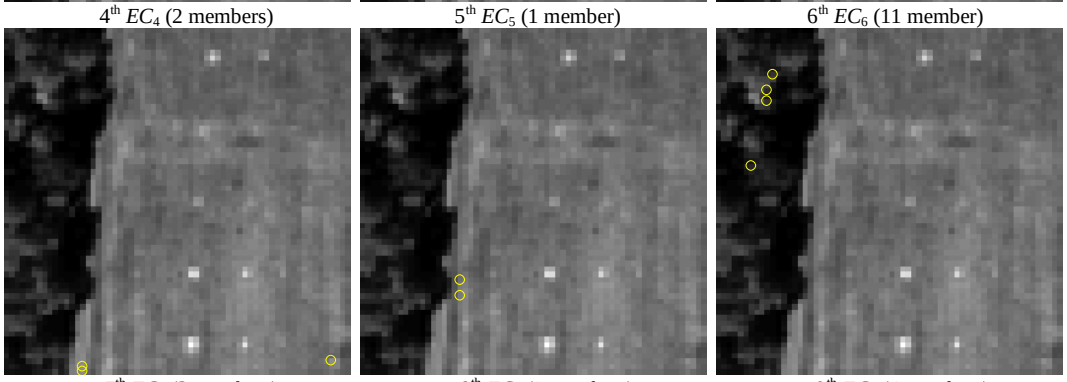
Figure 3.14 HYDICE: endmembers generated by EVA with UFCLS initialization



1st EC_1 (6 members) 2nd EC_2 (43 members) 3rd EC_3 (11 members)

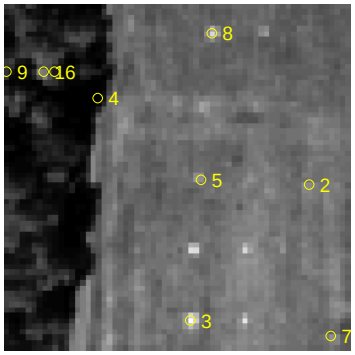


4th EC_4 (2 members) 5th EC_5 (1 member) 6th EC_6 (11 member)

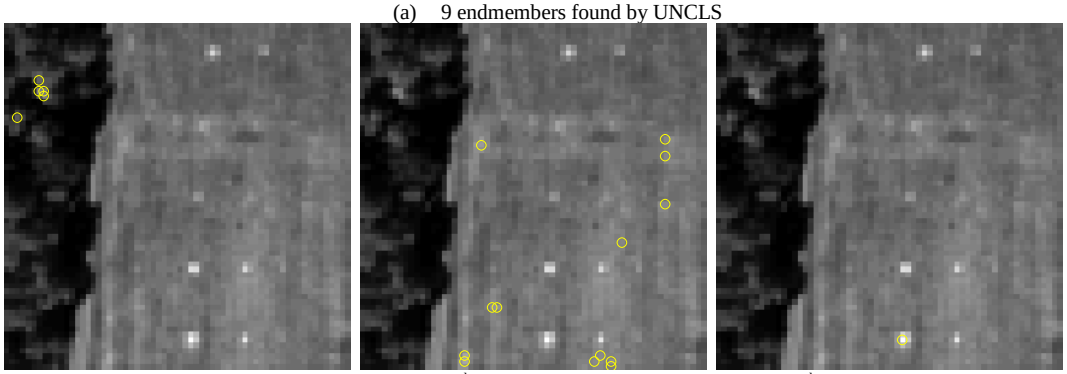


7th EC_7 (3 members) 8th EC_8 (2 members) 9th EC_9 (4 members)

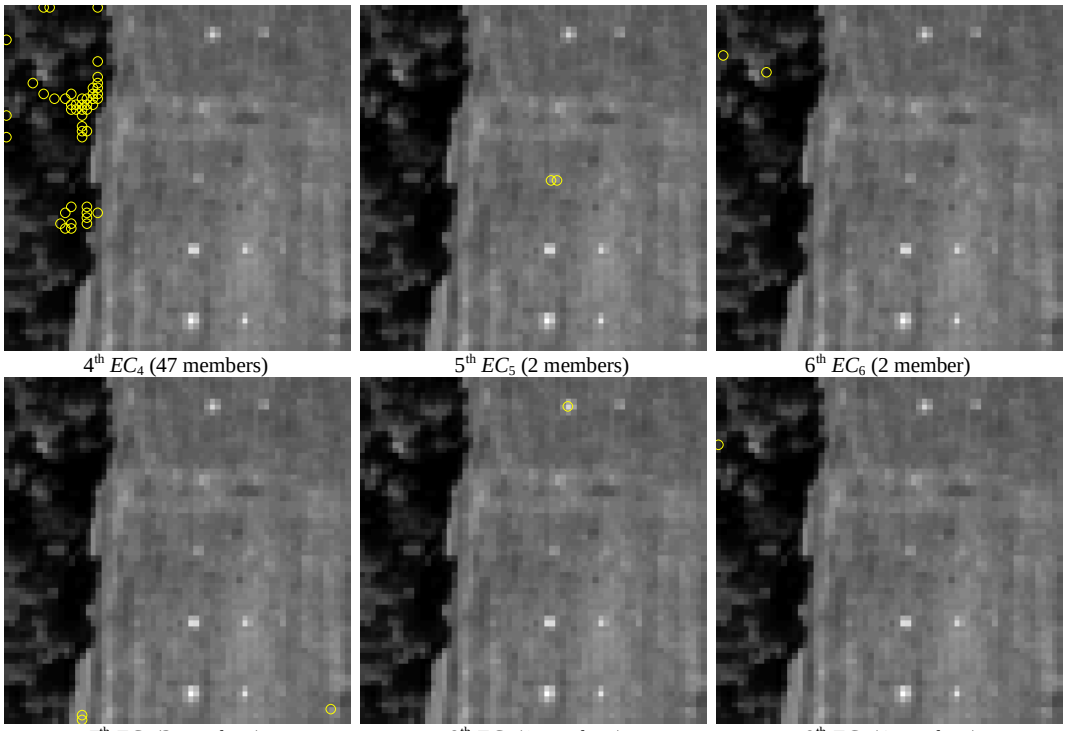
Figure 3.15 HYDICE: endmember classes of 5th run by EVA



(a) 9 endmembers found by UNCLS



1st EC_1 (5 members) 2nd EC_2 (13 members) 3rd EC_3 (1 members)

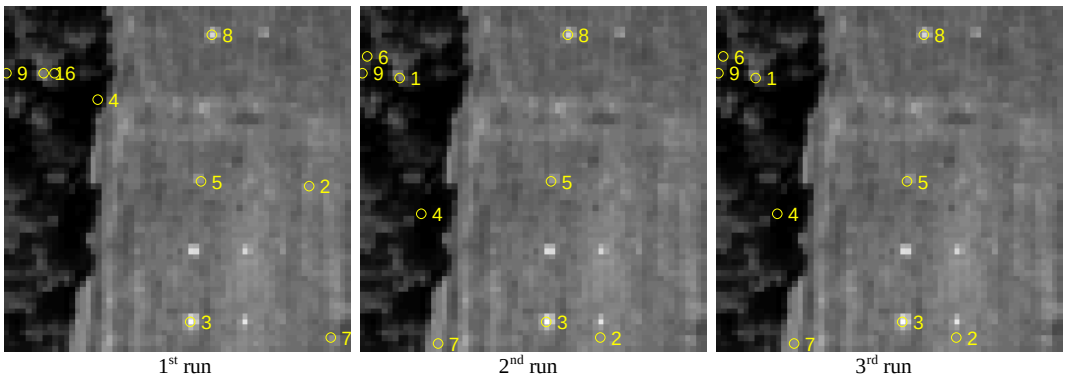


4th EC_4 (47 members) 5th EC_5 (2 members) 6th EC_6 (2 member)

7th EC_7 (3 members) 8th EC_8 (1 members) 9th EC_9 (4 members)

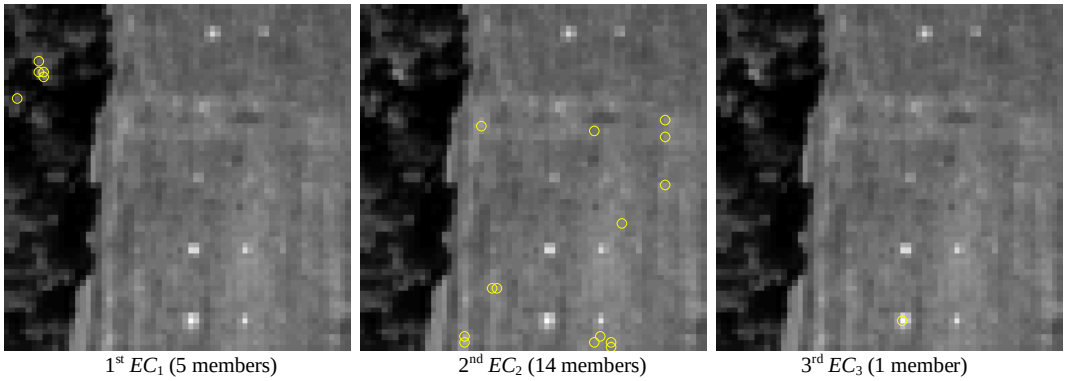
(b) Finding 9 endmember classes

Figure 3.16 HYDICE: UNCLS-generated endmembers and their endmember classes

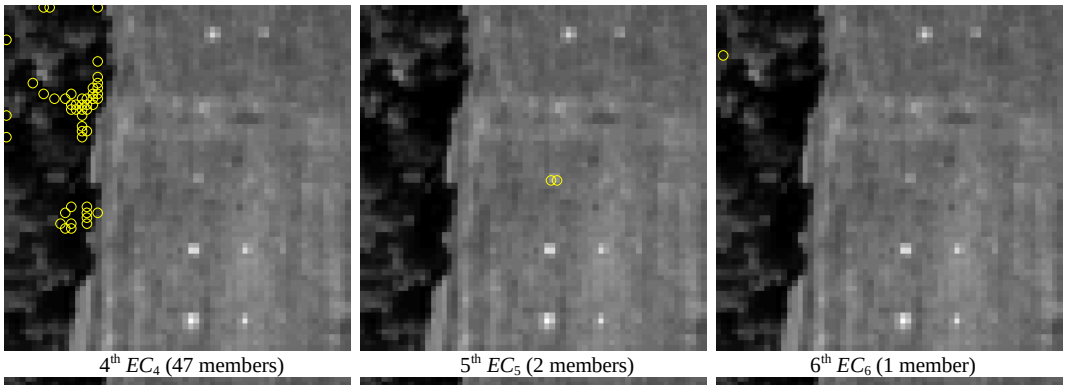


1st run 2nd run 3rd run

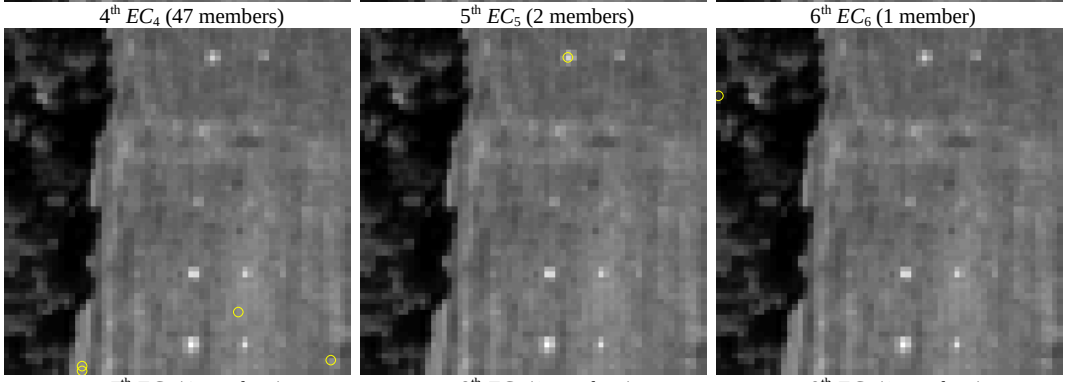
Figure 3.17 HYDICE: endmembers generated by EVA with UNCLS initialization



1st EC_1 (5 members) 2nd EC_2 (14 members) 3rd EC_3 (1 member)



4th EC_4 (47 members) 5th EC_5 (2 members) 6th EC_6 (1 member)

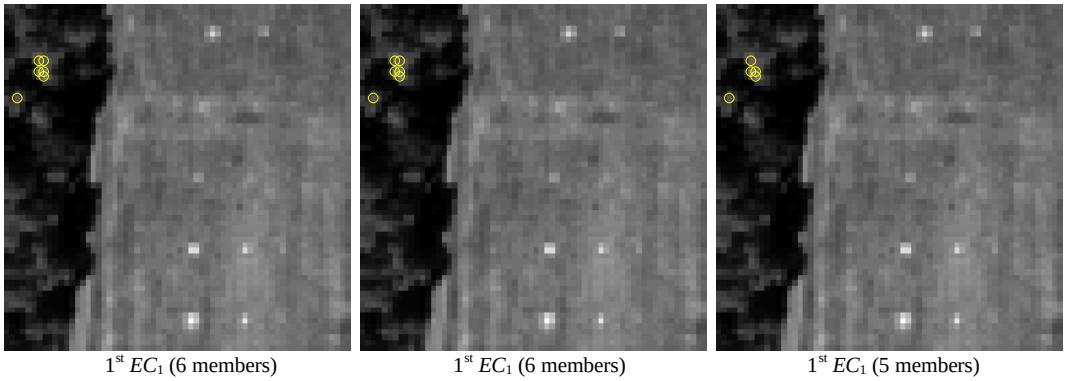


7th EC_7 (4 members) 8th EC_8 (1 members) 9th EC_9 (1 members)

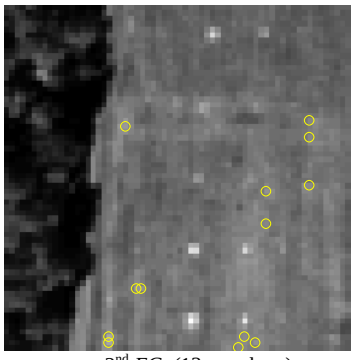
Figure 3.18 HYDICE: endmember classes of 3rd run by EVA

Endmember classes results initialized by UFCLS and UNCLS are shown in Fig 3.13-Fig 3.15 and Fig 3.16-Fig 3.18, respectively. With two different sets of initial conditions, endmember classes are different. However, they have some similar endmember classes shown as follows.

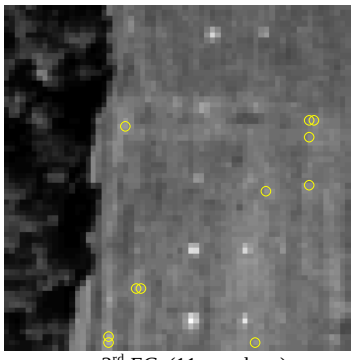
ATGP UFCLS UNCLS



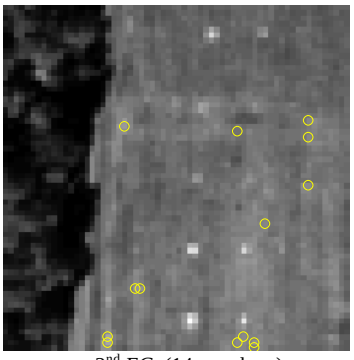
1st EC_1 (6 members) 1st EC_1 (6 members) 1st EC_1 (5 members)



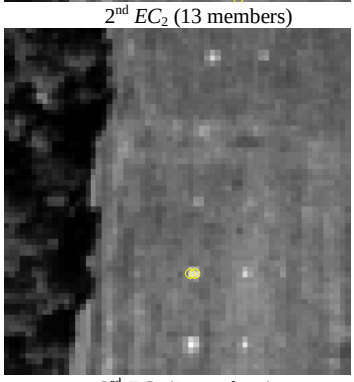
2nd EC_2 (13 members)



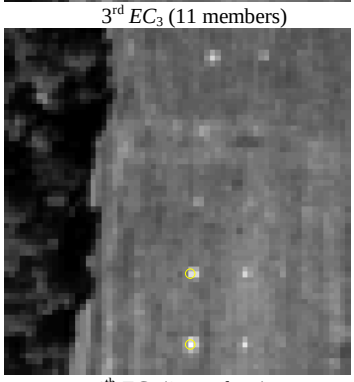
3rd EC_3 (11 members)



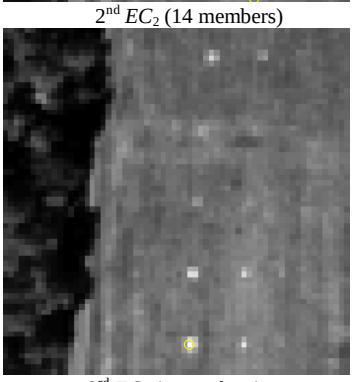
2nd EC_2 (14 members)



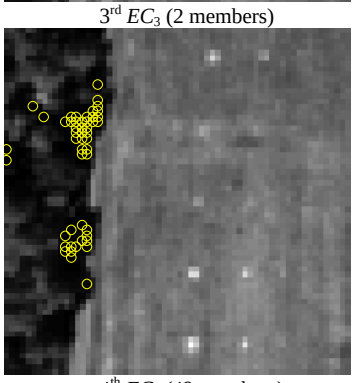
3rd EC_3 (2 members)



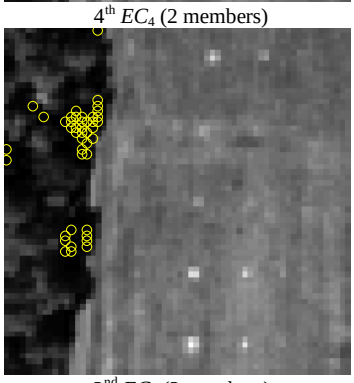
4th EC_4 (2 members)



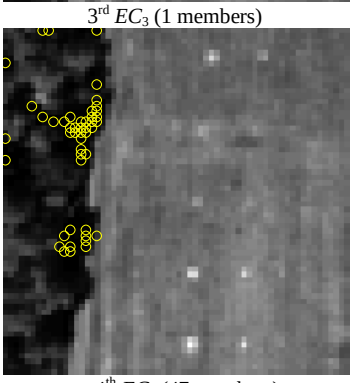
3rd EC_3 (1 members)



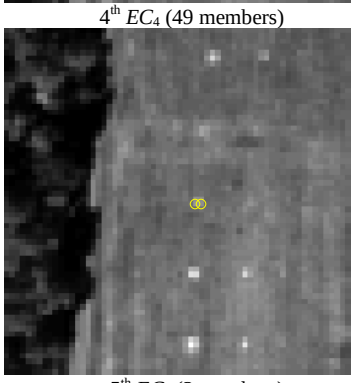
4th EC_4 (49 members)



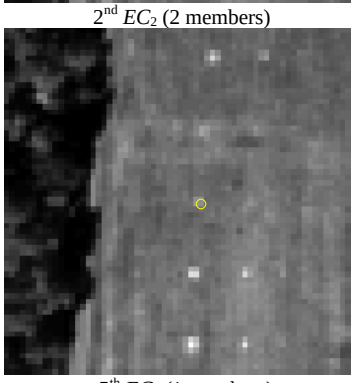
2nd EC_2 (2 members)



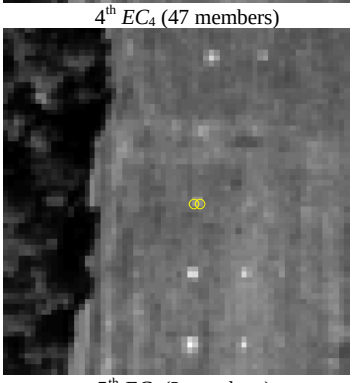
4th EC_4 (47 members)



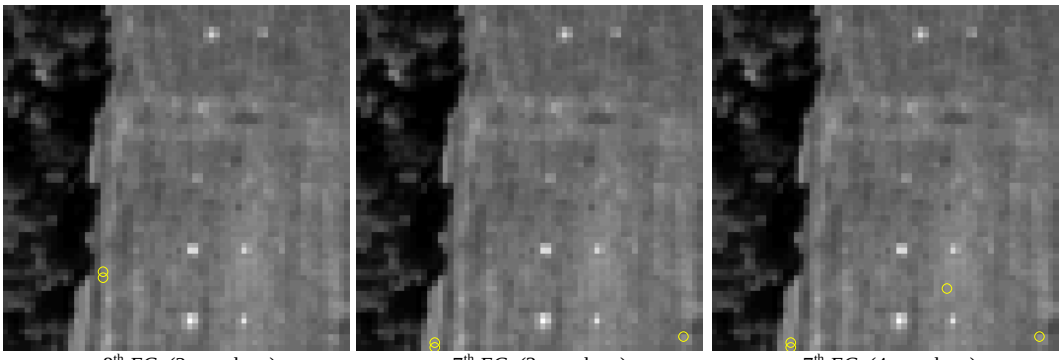
5th EC_5 (2 members)



5th EC_5 (1 members)



5th EC_5 (2 members)



8th EC_8 (2 members) 7th EC_7 (3 members) 7th EC_7 (4 members)

Figure 3.19 Endmember classes comparison of different initialization algorithms ATGP, UFCLS and UNCLS

As a second example an Airborne Visible/InfraRed Image Spectrometer (AVIRIS) image scene shown in Fig 3.20 was used for experiments. It is the Lunar Crater Volcanic Field (LCVF) located in Northern Nye County, NV. Atmospheric water bands and low SNR bands have been removed from the data, reducing the image cube from 224 to 158 bands. The image in Fig 3.20 has 10nm spectral resolution and 20m spatial resolution. There are five target of interest, the radiance spectra of red oxidized basaltic cinders, rhyolite, playa (dry lake), vegetation and shade along with a two-pixel wide anomaly.

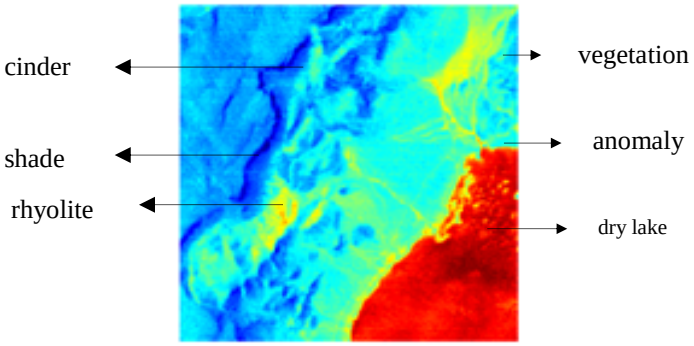


Figure 3.20 AVARIS LCVF scene

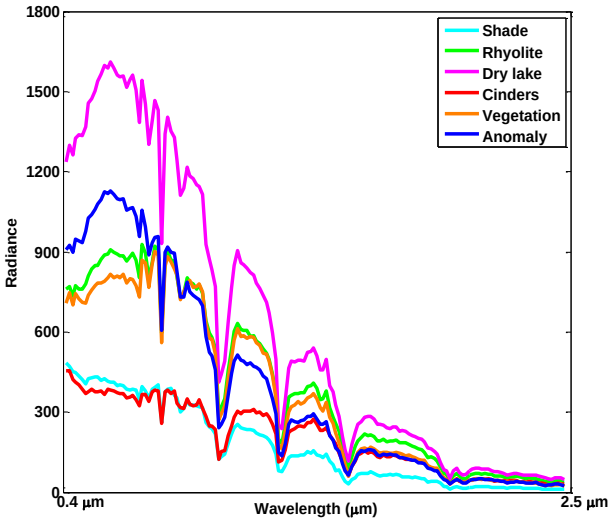
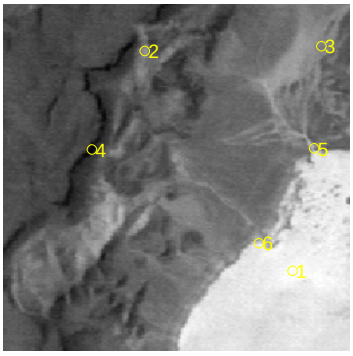
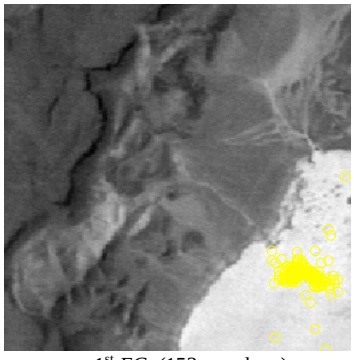


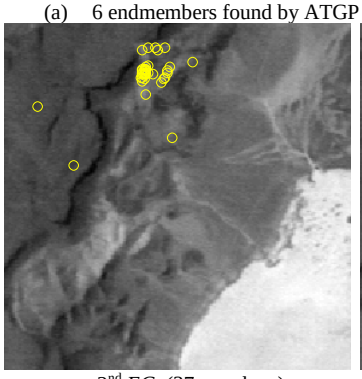
Figure 3.21 Spectral profile AVARIS LCVF scene



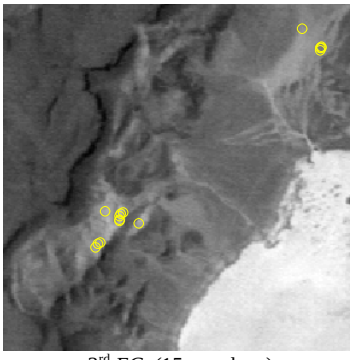
(a) 6 endmembers found by ATGP



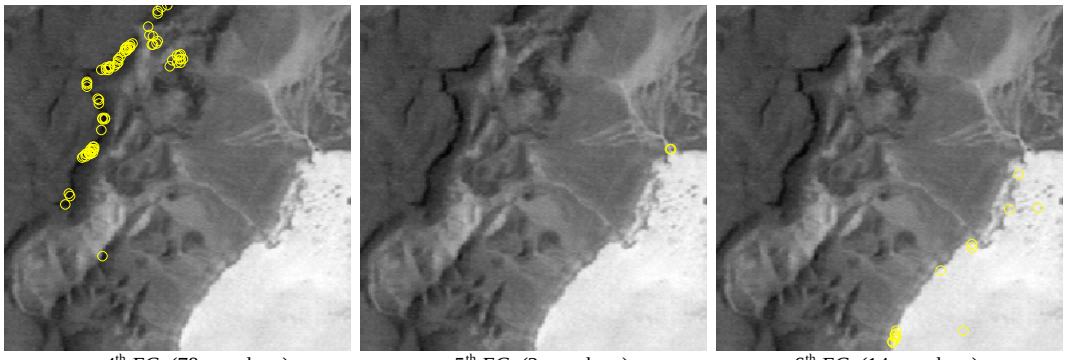
1st EC_1 (153 members)



2nd EC_2 (37 members)



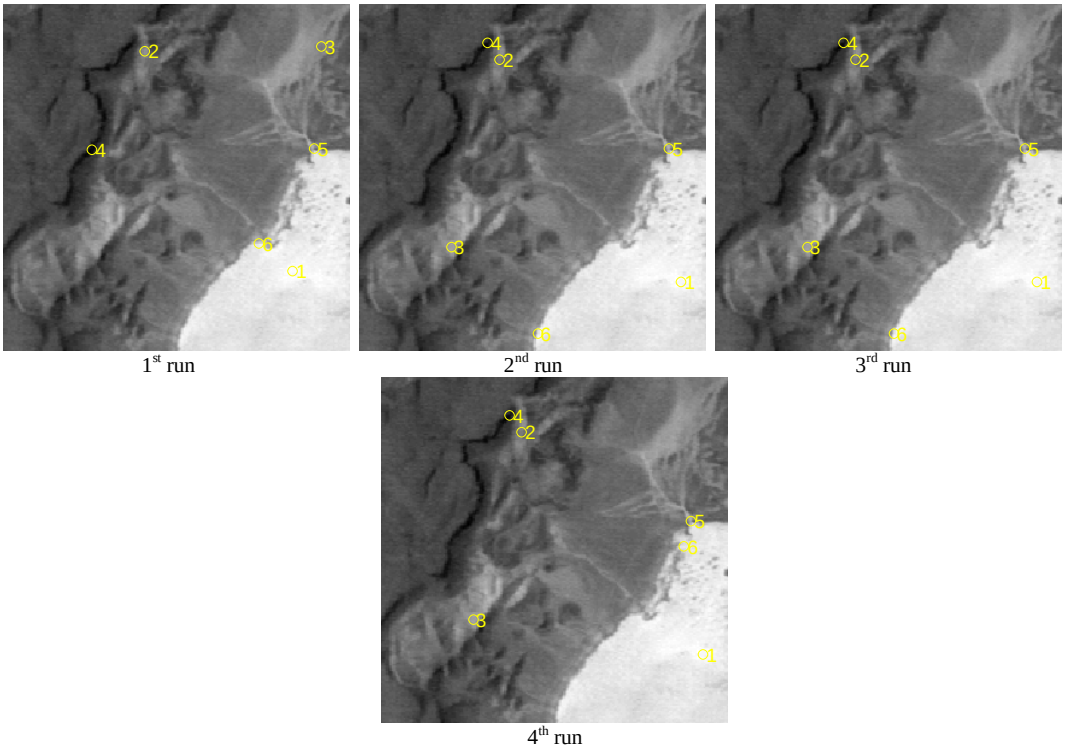
3rd EC_3 (15 members)



4th EC_4 (78 members) 5th EC_5 (2 members) 6th EC_6 (14 members)

(b) Finding 6 endmember classes

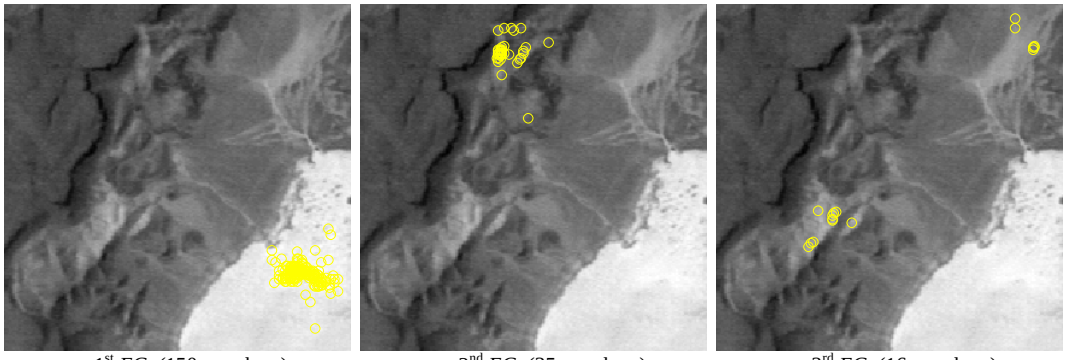
Figure 3.22 LCVF: ATGP-generated endmembers and their endmember classes



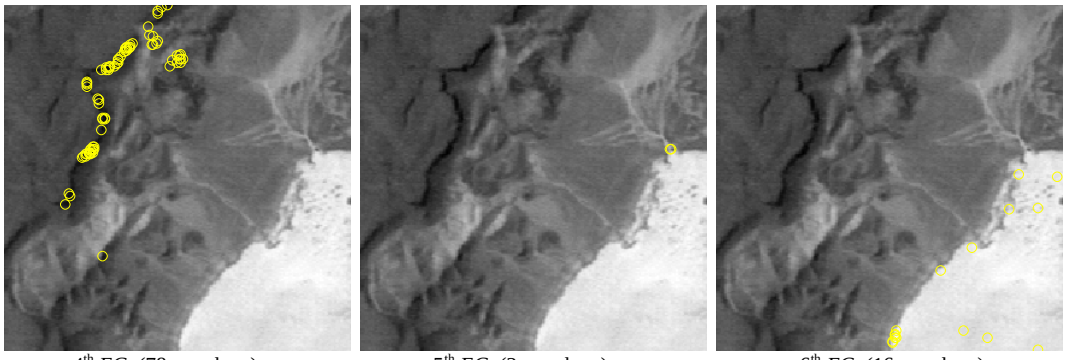
1st run 2nd run 3rd run

4th run

Figure 3.23 LCVF: endmembers generated by EVA with ATGP initialization

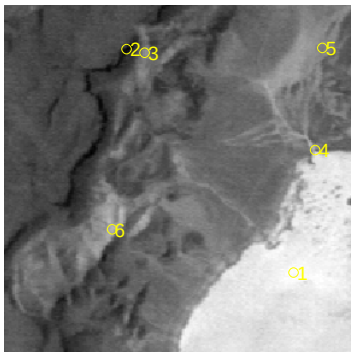


1st EC_1 (150 members) 2nd EC_2 (35 members) 3rd EC_3 (16 members)

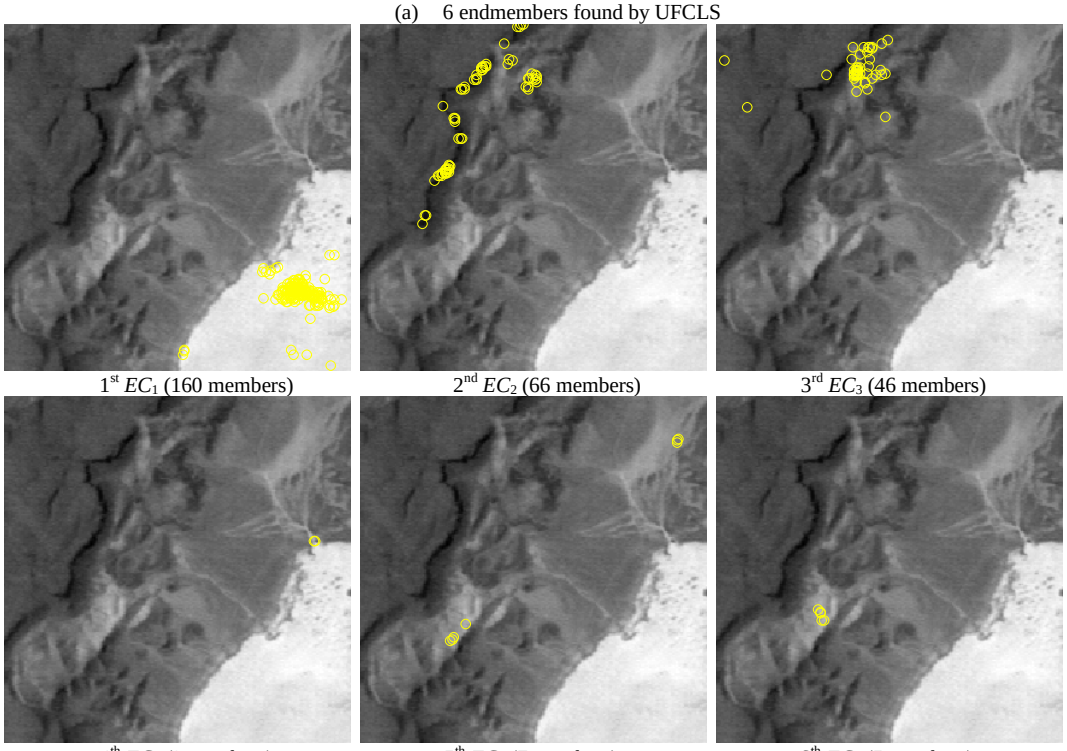


4th EC_4 (78 members) 5th EC_5 (2 members) 6th EC_6 (16 members)

Figure 3.24 LCVF: endmember classes of 4th run by EVA



(a) 6 endmembers found by UFCLS



1st EC_1 (160 members) 2nd EC_2 (66 members) 3rd EC_3 (46 members)

4th EC_4 (2 members) 5th EC_5 (7 members) 6th EC_6 (5 members)

(b) Finding 6 endmember classes

Figure 3.255 LCVF: UFCLS-generated endmembers and their endmember classes

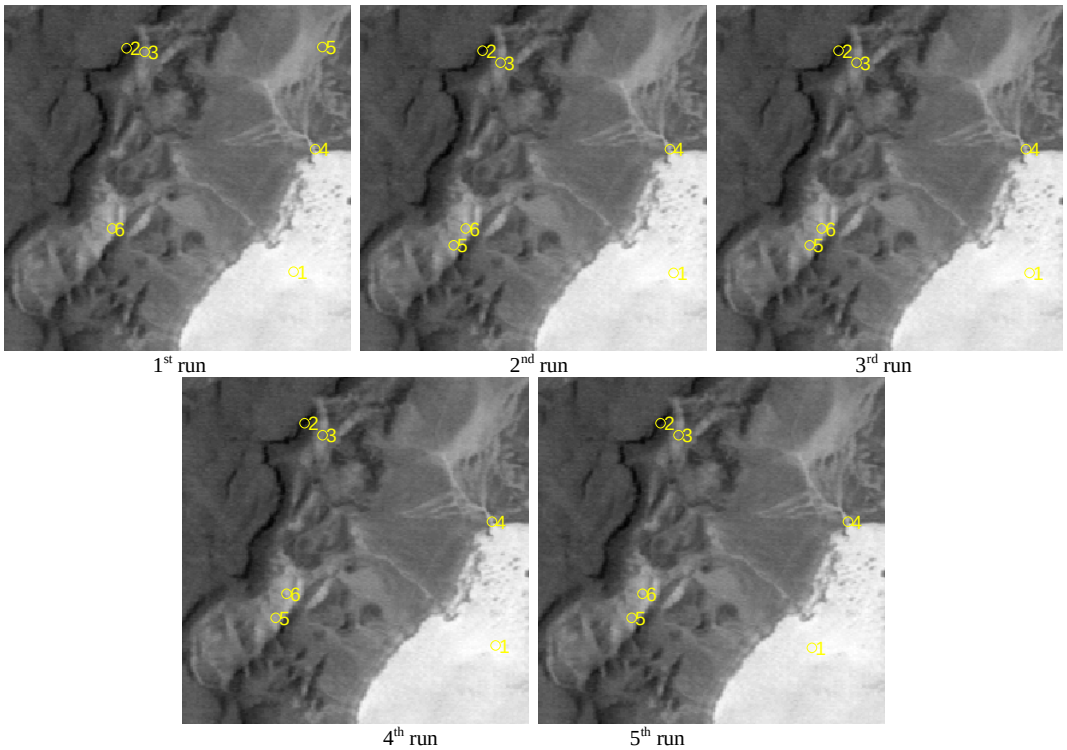


Figure 3.26 LCVF: endmembers generated by EVA with UFCLS initialization

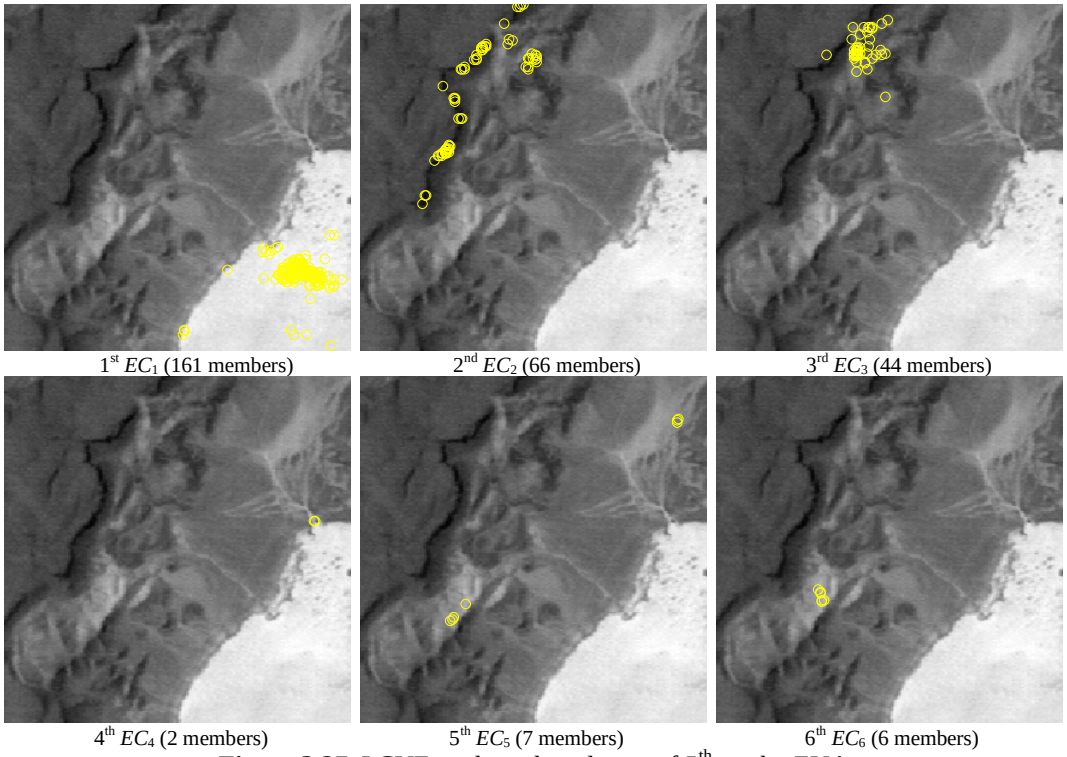
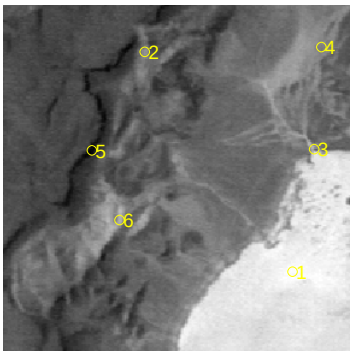
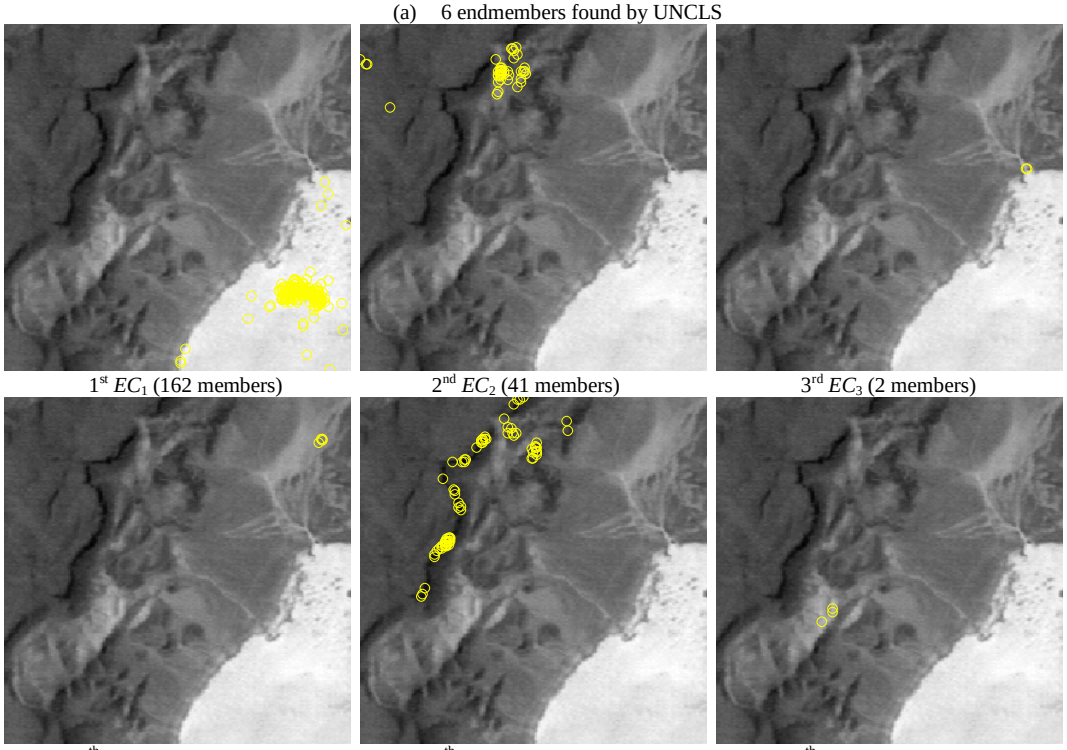


Figure 3.27 LCVF: endmember classes of 5th run by EVA



(a) 6 endmembers found by UNCLS

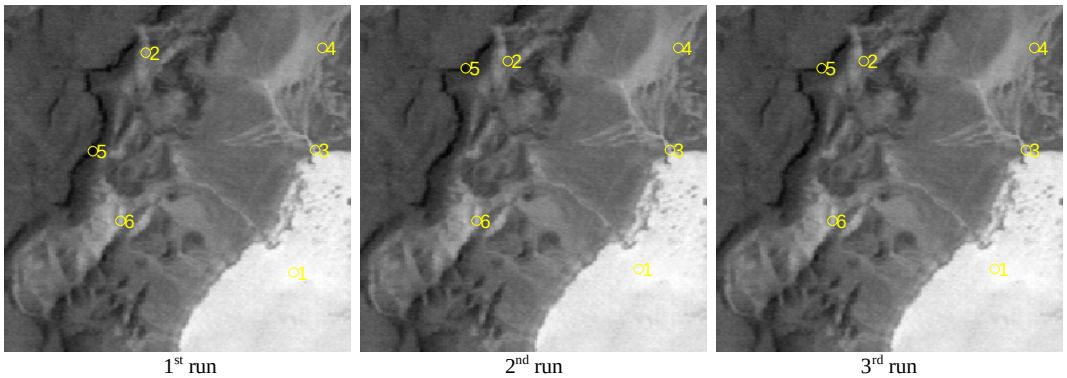


1st EC_1 (162 members) 2nd EC_2 (41 members) 3rd EC_3 (2 members)

4th EC_4 (4 members) 5th EC_5 (73 members) 6th EC_6 (3 members)

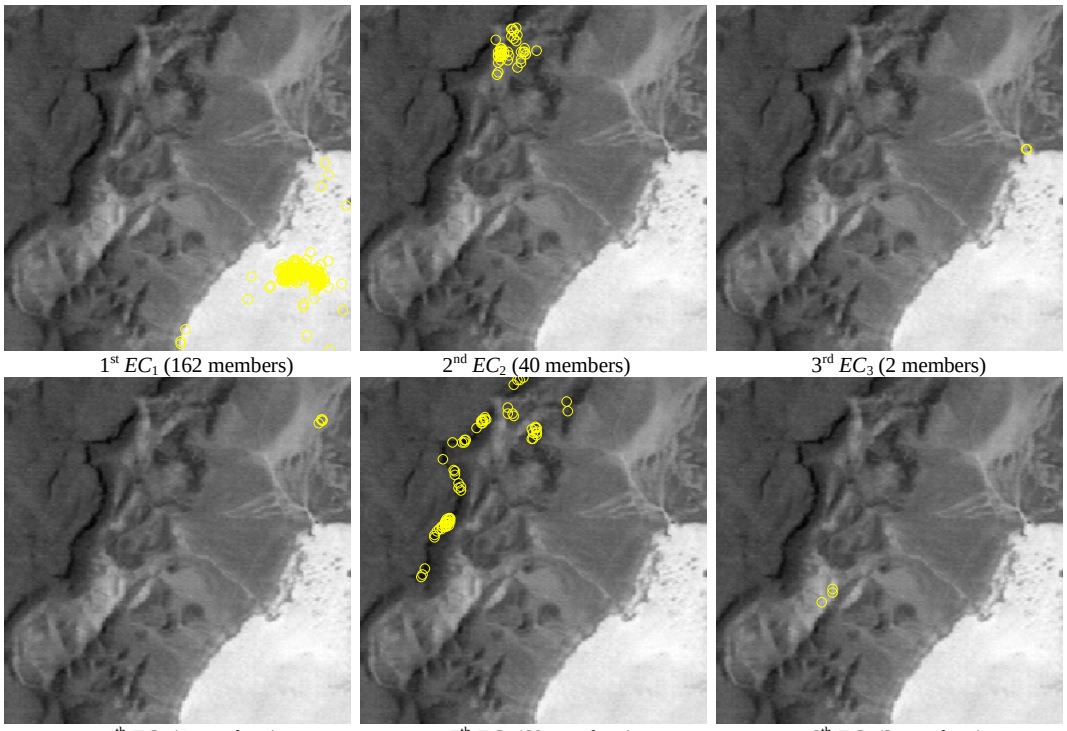
(b) Finding 6 endmember classes

Figure 3.28 LCVF: UNCLS-generated endmembers and their endmember classes



1st run 2nd run 3rd run

Figure 3.29 LCVF: endmembers generated by EVA with UNCLS initialization



1st EC_1 (162 members) 2nd EC_2 (40 members) 3rd EC_3 (2 members)

4th EC_4 (4 members) 5th EC_5 (69 members) 6th EC_6 (3 members)

Figure 3.30 LCVF: endmember classes of 3rd run by EVA

The same experiments done for HYDICE data were also conducted on LCVF data. Results for endmember classes are shown in Fig 3.22-Fig 3.24, Fig 3.25-Fig 3.27 and Fig 3.28-Fig 3.29 which used ATGP, UFCLS and UNCLS as initialization algorithms to produce initial endmembers for EVA, respectively. Endmember class results with different initial conditions are not identical but they are very similar except that the order of classes is different.

3.5 Conclusion

In this chapter, an unsupervised method called Endmember Variability (EVA) is developed that does not require any prior knowledge or assumptions. Furthermore, finding endmember class algorithm is an iterative process, where an unsupervised endmember finding algorithm such as ATGP, UFCLS or UNCLS, is used to find

initial endmembers prior to the implementation of EVA. The set of endmember classes are refined by an iterative process implemented in EVA.

Chapter 4 Fisher's Ratio-Based Approach to Finding Endmember Classes and Endmembers

4.1 Introduction

As mentioned in Chapter 1 and Chapter 2, the existence of endmembers will be unlikely guaranteed due to many unforeseeable effects resulting from physical phenomena in real world problems. Under such a circumstance, endmember variability needs to be considered when it comes to endmember finding.

Since an endmember class is to group a set of data samples with their spectral signatures very similar to a true endmember, a logical approach is to assume the true endmember to be the class center and all other samples in this class are simply considered as its variants. This further suggests that this class can be used to describe endmember variability around the endmember. From a pattern classification point of view, such a class endmember variability corresponds to the within-class variance that is used in one of most widely used classification technique, Fisher's linear discriminant analysis (FLDA) [23] - which makes use of Fisher's ratio (FR) or Raleigh quotient defined as the ratio of between-class scatter matrix to within-class scatter matrix to perform classification.

According to FLDA, Fisher's ratio defined by

$$FR = \max_{\mathbf{w}} \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}} \tag{4.1}$$

is an effective criterion widely used for pattern classification where $\mathbf{S}_B$ and $\mathbf{S}_W$ are between-class and within-class scatter matrices. $\mathbf{w}$ is the linear transformation.

In light of this interpretation we can define two types of endmember variability, between-endmember variability (BEV) and within-endmember variability (WEV) which can represent intra-endmember class variability and inter-endmember class variability very similar to within-class variance and between-class variance commonly used in pattern classification. By virtue of BEV and WEV a criterion similar to FR can be also defined as ratio of BEV to WEV, referred to as endmember variability ratio (EVR). As a result, EVR provides an alternative criterion to many other commonly used criteria for finding endmembers, orthogonal projection used by pixel purity index (PPI), simplex volume used by minimum volume transform (MVT), N-FINDR. In other words, a new criterion for finding endmembers can be derived to maximize EVR, which is equivalent to maximizing BEV while also minimizing WEV. An algorithm using EVR as an optimal criterion for finding endmembers is called endmember variability ratio-based endmember finding algorithm (EVR-EFA).

4.2 Criteria for Endmember Variability

Assume that $\{\mathbf{e}_j\}_{j=1}^{p}$ is a set of desired endmembers and $\{EC_j\}_{j=1}^{p}$ are their corresponding endmember classes where EC_j is the spectral class specified by the j^{th} endmember $\mathbf{e}_j$ which is a set made up of $\{\mathbf{s}_i^j\}_{i=1}^{n_j}$ where $\mathbf{s}_i^j$ is an element in EC_j and considered to be a spectral variant of $\mathbf{e}_j$, and n_j is the total number of elements in

EC_j. As will be described in the following, two types of endmember variability are of interest, between endmember variability (BEV) and within endmember variability (WEV).

4.2.1 Between Endmember Variability (BEV)

While WEV has been studied extensively in the literature, between-endmember variability (BEV) has never been defined and explored in the past. It can be measured by

$$BEV\left(\mathbf{e}_1,\ \mathbf{e}_2,\ \cdots,\ \mathbf{e}_p\right)=\sum_{m=1}^{p}\sum_{n=1,n\neq m}^{p}\left(\mathbf{e}_m-\mathbf{e}_n\right)^T\left(\mathbf{e}_m-\mathbf{e}_n\right) \tag{4.2}$$

which is similar to between-class variance used by FLDA.

4.2.2 Within Endmember Variability (WEV)

Within-endmember variability (WEV) can be measured by variances within each of p endmember classes, $\{EC_j\}_{j=1}^{p}$ via a distance measure such as Euclidean distance (ED) given by

$$WEV\left(\mathbf{e}_1,\ \mathbf{e}_2,\ \cdots,\ \mathbf{e}_p\right)=\sum_{m=1}^{p}\sum_{n=1}^{n_m}\left(\mathbf{s}_n^m-\mathbf{e}_m\right)^T\left(\mathbf{s}_n^m-\mathbf{e}_m\right) \tag{4.3}$$

Following an idea similar to (3.1) we can define a new criterion as

$$EVR\left(\mathbf{e}_1,\cdots,\mathbf{e}_j,\cdots,\mathbf{e}_p\right)=\frac{BEV\left(\mathbf{e}_1,\cdots,\mathbf{e}_j,\cdots,\mathbf{e}_p\right)}{WEV\left(\mathbf{e}_1,\cdots,\mathbf{e}_j,\cdots,\mathbf{e}_p\right)} \tag{4.4}$$

4.3 Finding Endmember Classes

Endmember classes need to be found in order to calculate within-endmember variability (WEV). According to previous literature, a predefined threshold is used to find endmember classes. There are two major issues about the predefined threshold.

(1) Without any prior knowledge, it is usually very difficult to determine the threshold. (2) For each endmember class, it is supposed to have its own variability. We cannot simply apply one threshold to all endmember classes. The following method can be used to solve those two issues. Let $\{\mathbf{e}_j\}_{j=1}^{p}$ be the desired set of endmembers and $\{EC_j\}_{j=1}^{p}$ be the corresponding endmember classes. Endmember classes can be found using the following method.

$$\mathbf{r} \in \{EC\}_{j^*} \Leftarrow j^* = \arg\{\min_{1\le j\le p} \mathrm{SAM}(\mathbf{r}, \mathbf{e}_j)\} \tag{4.5}$$

where $\mathbf{r}$ is a data sample and $\mathbf{e}_j$ is endmember class center. Similarity measurement in (4.5) SAM (Spectral Angle Mapper) can be replaced by Euclidean Distance (ED) or Spectral Information Divergence (SID) [1]. In this way, there is no need to specify any threshold.

4.4 Design and Development of Endmember Variability – Endmember Finding Algorithm (EVR-EFA)

Two algorithms can be developed for EVR-based endmember finding approach. One is derived from an idea similar to successive N-FINDR and is called SuCcessive EVR-EFA (SC EVR-EFA). Another one is also derived from sequential N-FINDR and is called SeQuential EVR-EFA (SQ EVR-EFA) which interchanges the order of implementing inner loop and outer loop in SC EVR-EFA.

4.4.1 Successive EVR-EFA

Despite SC EVR-EFA and SC N-FINDR share similar ideas their implementations are different in terms of their criteria where N-FINDER maximizes simplex volume and EVR-EFA maximizes EVR specified by (4.4).

SuCcessive EVR-EFA (SC EVR-EFA)

Preprocessing:

1. Let p be the number of endmembers required to generate.
2. Initialization:

   Let $\{\mathbf{r}_i^{PPI}\}_{i=1}^{M}$ be all data sample vectors from Pixel Purity Index (PPI). Randomly select p pixels as an initial set of endmembers $\{\mathbf{e}_j^{(0)}\}_{j=1}^{p}$ and their corresponding endmember classes, $\{EC(\mathbf{e}_j^{(0)})\}_{j=1}^{p}$ by the method developed in Section 4.3 and perform data sphering.
3. Outer loop

   For $\{\mathbf{r}_i^{PPI}\}_{i=1}^{M}$, find $\mathbf{e}_j^{(*)}$ which is a desired endmember to replace $\mathbf{e}_j^{(0)}$. (Here we use index j as a counter to keep track of the number of endmembers already and currently being processed)
4. Inner loop (Here we use index i as a counter to keep track of data sample vector)

   For $1 \le i \le M$ we calculate the Fisher's ratio $\mathrm{EVR}\left(\mathbf{e}_1^{(*)}, \cdots, \mathbf{e}_{j-1}^{(*)}, \mathbf{r}_i, \mathbf{e}_{j+1}^{(0)}, \cdots, \mathbf{e}_p^{(0)}\right)$ defined in (3.2), (3.3) and (3.4) while

fixing other endmembers $\mathbf{e}_l^{(*)}$ with $l < j$ and $\mathbf{e}_l^{(0)}$ with $l > j$. Let $\mathbf{e}_j^{(*)} = arg\left\{\max_{\mathbf{r}_i} \mathrm{EVR}\left(\mathbf{e}_1^{(*)}, \cdots, \mathbf{e}_{j-1}^{(*)}, \mathbf{r}_i, \mathbf{e}_{j+1}^{(0)}, \cdots, \mathbf{e}_p^{(0)}\right)\right\}$.

5. Stopping rule:

   If $j < p$ then $j \leftarrow j+1$ and go to step 3. Otherwise, the final set of $\left\{\mathbf{e}_1^{(*)}, \mathbf{e}_2^{(*)}, \cdots, \mathbf{e}_p^{(*)}\right\}$ is a desired set of p endmembers.

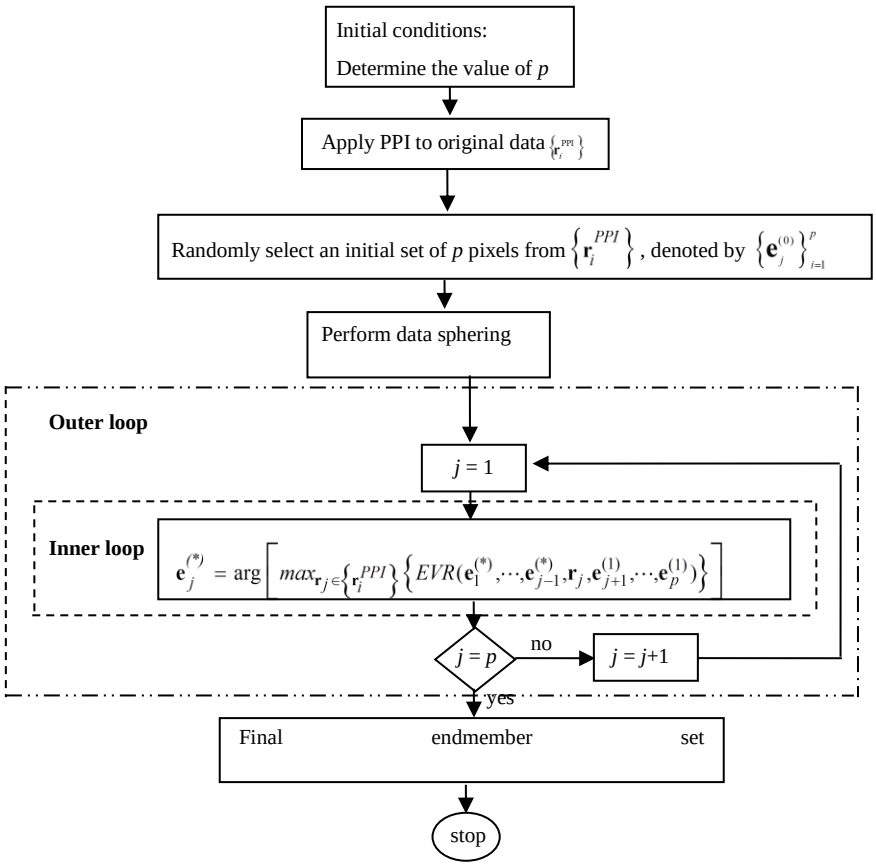


Figure 4.1 Flowchart of SC EVR-EFA using random initial conditions

4.4.2 Sequential EVR-EFA

A similar version to SQ N-FINDR can be also derived, called SQ EVR-EFA as follows.

SQ EVR-EFA

1. Preprocessing:

   Let p be the number of endmembers required to generate.

   Let $\{\mathbf{r}_i\}_{i=1}^{M}$ be all data sample vectors from Pixel Purity Index (PPI)

2. Initialization:

   Let $\{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \cdots, \mathbf{e}_p^{(0)}\}$ be a set of initial vectors randomly selected from the data.

   Perform data sphering.

   Set $i=1$ and go to step 3

3. Outer loop: (Here we use index i as a counter to keep track of data sample vector $\mathbf{r}_i$)

   a. Check if $i=N$. If yes, the algorithm is terminated. Otherwise, let $i \leftarrow i+1$ and continue.

   b. Input the i^{th} data sample vector $\mathbf{r}_i$. (Note that $\mathbf{r}_i$ now is the i+1$^{\text{st}}$ data sample vector $\mathbf{r}_{i+1}$).

4. Inner loop: (Here we use j as counter to keep track of the j^{th} endmember $\mathbf{e}_j$)

   a. Using $\mathbf{r}_i$ to replace each of $\{\mathbf{e}_1^{(j)}, \mathbf{e}_2^{(j)}, \cdots, \mathbf{e}_p^{(j)}\}$, calculate all the Fisher's ratios $\text{EVR}(\mathbf{r}_i, \mathbf{e}_2^{(j)}, \cdots, \mathbf{e}_p^{(j)})$, $\text{EVR}(\mathbf{e}_1^{(j)}, \mathbf{r}_i, \mathbf{e}_3^{(j)}, \cdots, \mathbf{e}_p^{(j)})$, $\text{EVR}(\mathbf{e}_1^{(j)}, \cdots, \mathbf{e}_{p-1}^{(j)}, \mathbf{r}_i)$. If there exists one of them is greater than $\text{EVR}(\mathbf{e}_1^{(j)}, \mathbf{e}_2^{(j)}, \cdots, \mathbf{e}_p^{(j)})$, go to step 3.

   b. Replacement rule:

The endmember which is absent in the largest Fisher's ratio among the p Fisher's ratios $\mathrm{EVR}\left(\mathbf{r}_i, \mathbf{e}_2^{(j)}, \cdots, \mathbf{e}_p^{(j)}\right)$, $\mathrm{EVR}\left(\mathbf{e}_1^{(j)}, \mathbf{r}_i, \mathbf{e}_3^{(j)}, \cdots, \mathbf{e}_p^{(j)}\right)$, $\mathrm{EVR}\left(\mathbf{e}_1^{(j)}, \cdots, \mathbf{e}_{p-1}^{(j)}, \mathbf{r}_i\right)$, denoted by $\mathbf{e}_l^{(j)}$ for index l will be replaced by the i^{th} sample vector $\mathbf{r}_i$, A new set of endmembers is then produced by letting $\mathbf{e}_l^{(j+1)} = \mathbf{r}_i$ and $\mathbf{e}_l^{(j+1)} = \mathbf{e}_l^{(j)}$ for $j \neq l$ and go to step 3.

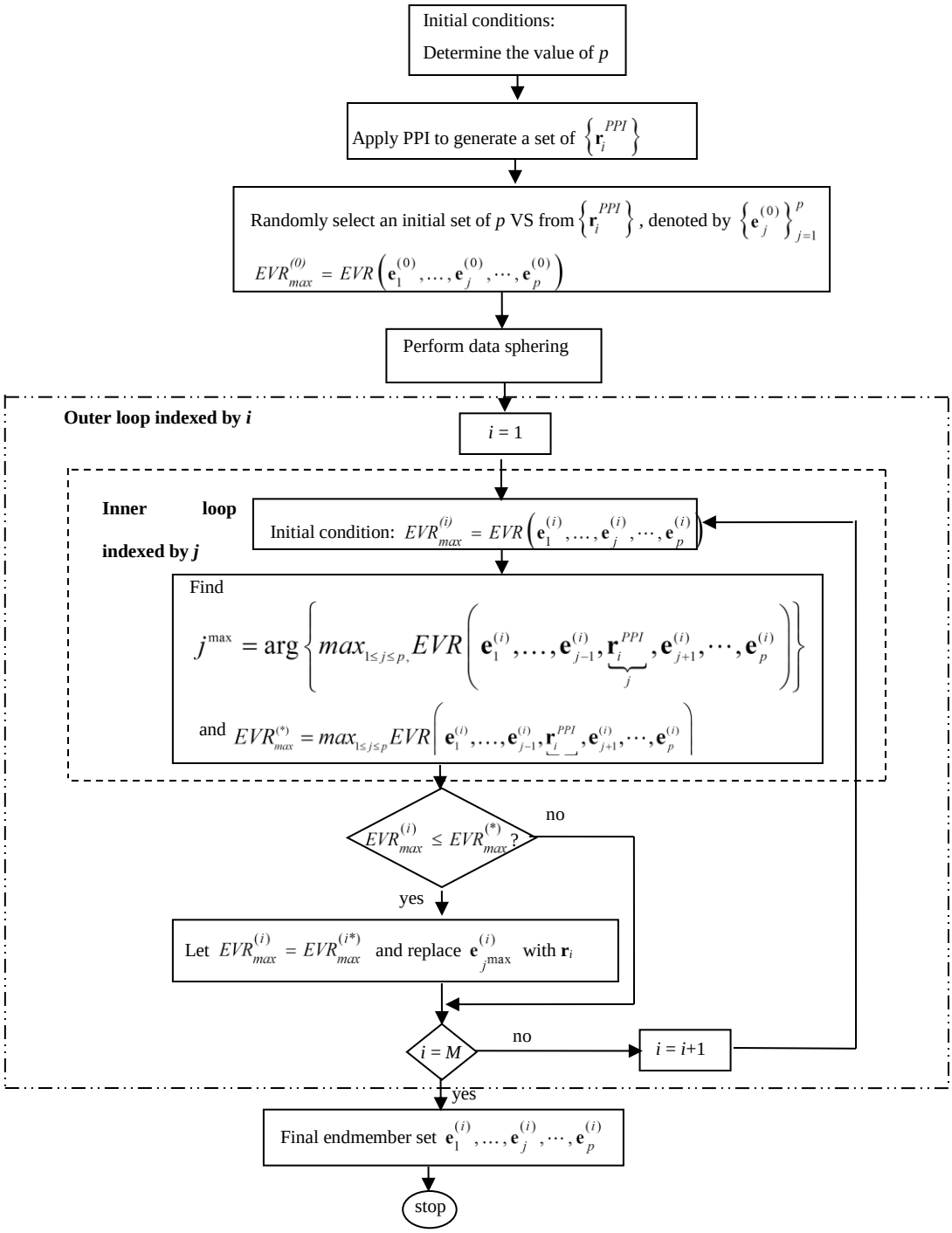


Figure 4.2 Flowchart of SQ EVR-EFA using random initial conditions

4.4.3 Growing EVR-EFA

The SQ EVR-EFA and SC EVR-EFA developed in previous sections deal with *p* endmembers at the same time. From computational as well real-time processing points of view they may not quite fit real world applications. This section we develop a growing EVR-EFA which grows endmembers via EVR, referred to as G-EVR-EFA and can be considered an EVR version of SGA. With a slight modification of SC EVR-EFA an algorithm to implement G-EVR-EFA is detailed as follows.

G-EVR-EFA

Preprocessing:

1. Let p be the number of endmembers required to generate.

   Perform data sphering.

2. Initialization:

   Let $\mathbf{e}_1^{(0)} = \arg\{\max_{\mathbf{r}} \mathbf{r}^T\mathbf{r}\}$ be the first endmember. Let $\{\mathbf{r}_i\}_{i=1}^{M}$ be all data sample vectors from Pixel Purity Index (PPI) and $\{EC_j\}_{j=1}^{p}$ be the j^{th} endmember class.

3. Outer loop

   For $2 \le j \le p$, find $\mathbf{e}_j^{(*)}$ which is a new desired endmember to be added. (Here we use index *j* as a counter to keep track of the number of endmembers already)

4. Inner loop (Here we use index *i* as a counter to keep track of data sample vector)

For $1 \le i \le N$ we calculate the Fisher's ratio $\text{EVR}\left(\mathbf{e}_1^{(*)}, \cdots, \mathbf{e}_{j-1}^{(*)}, \mathbf{r}_i\right)$ defined in (3.4).

5. Let $\mathbf{e}_j^{(*)} = arg\left\{\max_{\mathbf{r}_i} \text{EVR}\left(\mathbf{e}_1^{(*)}, \cdots, \mathbf{e}_{j-1}^{(*)}, \mathbf{r}_i\right)\right\}$.

6. Stopping rule:

   If $j < p$ then $j \leftarrow j+1$ and go to step 3. Otherwise, the final set of $\left\{\mathbf{e}_1^{(*)}, \mathbf{e}_2^{(*)}, \cdots, \mathbf{e}_p^{(*)}\right\}$ is a desired set of p endmembers.

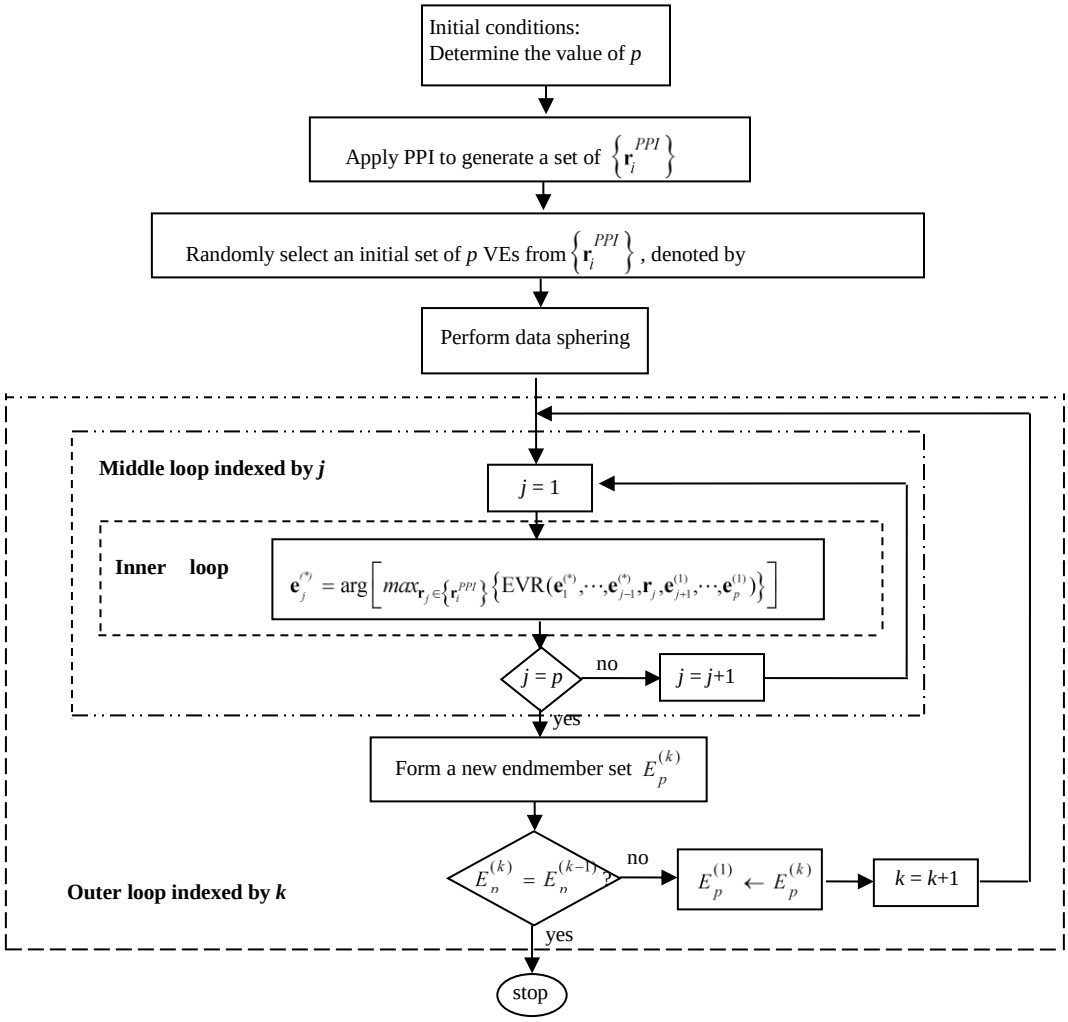


Figure 4.3 Flowchart of iterative SC EVR-EFA using random initial conditions

4.4.4 Iterative EVR-EFA

The resulting endmember sets may be different using random initial conditions by SC EVR-EFA or SQ EVR-EFA. In order to solve randomness issue, iterative version of EVR-EFA (I-EVR-EFA) is developed as follows.

I-EVR-EFA

1. Randomly generate a set of p initial endmembers, $VE^{(0)} = \left\{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \cdots, \mathbf{e}_p^{(0)}\right\}$.
2. Let k = 1 which is a counter to keep track of the number of times the outermost loop is executed.
3. For any k, apply either SC EVR-EFA or SQ EVR-EFA to generate a set of endmembers, denoted by $VE^{(k)} = \left\{\mathbf{e}_1^{(k)}, \mathbf{e}_2^{(k)}, \cdots, \mathbf{e}_p^{(k)}\right\}$.
4. Check if $VE^{(k-1)} = VE^{(k)}$. If it is, then the algorithm is terminated. Otherwise, continue. Let $k \leftarrow k+1$ and go to step 3.

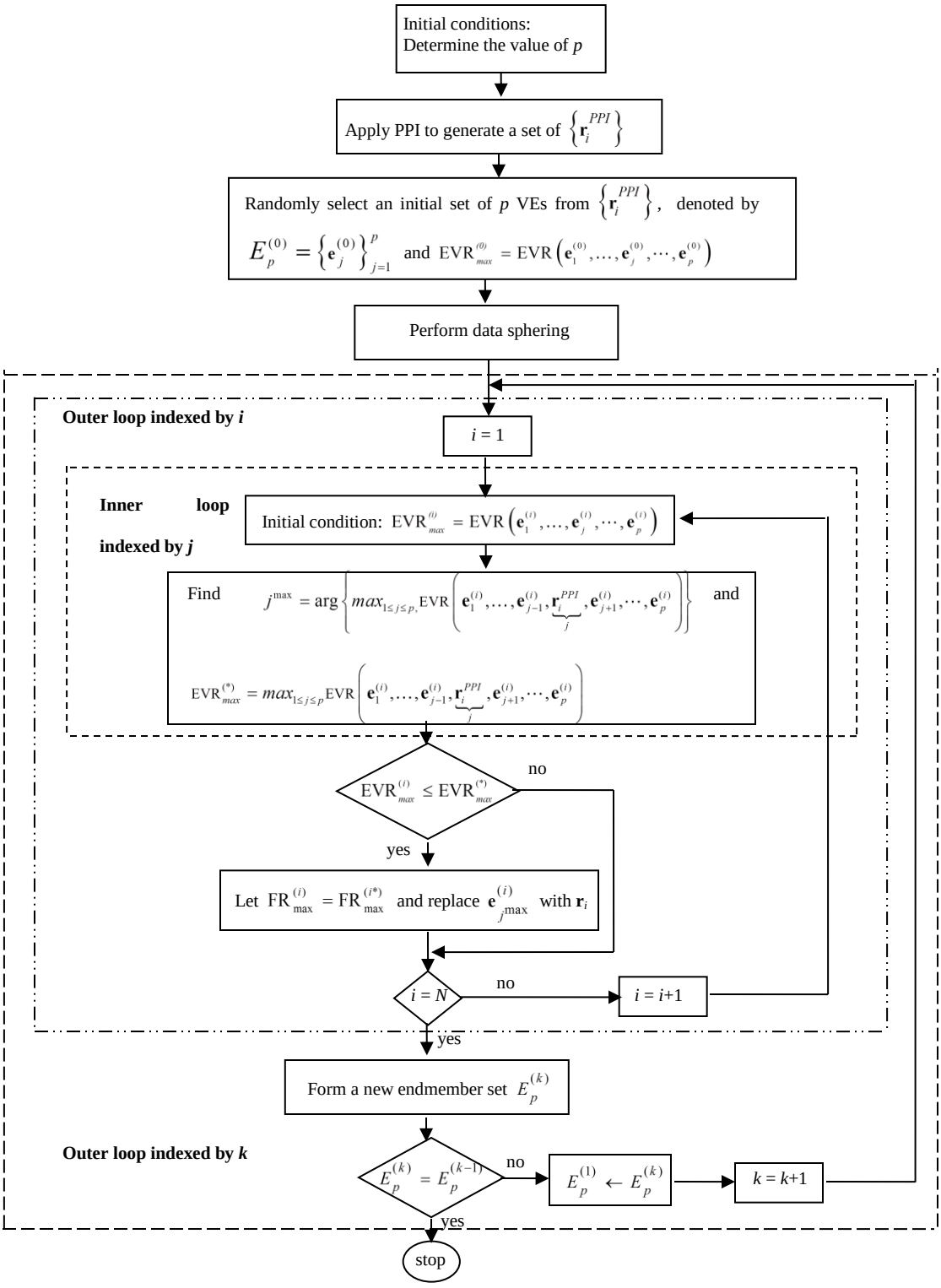


Figure 4.4 Flowchart of iterative SQ EVR-EFA using random initial conditions

4.5 Experiment

4.5.1 Synthetic Image Experiments

The same set of synthetic images, including TI and TE, are used for endmember variability algorithm (EVA). The detailed introduction of TI and TE can be found in Section 3.5.1.

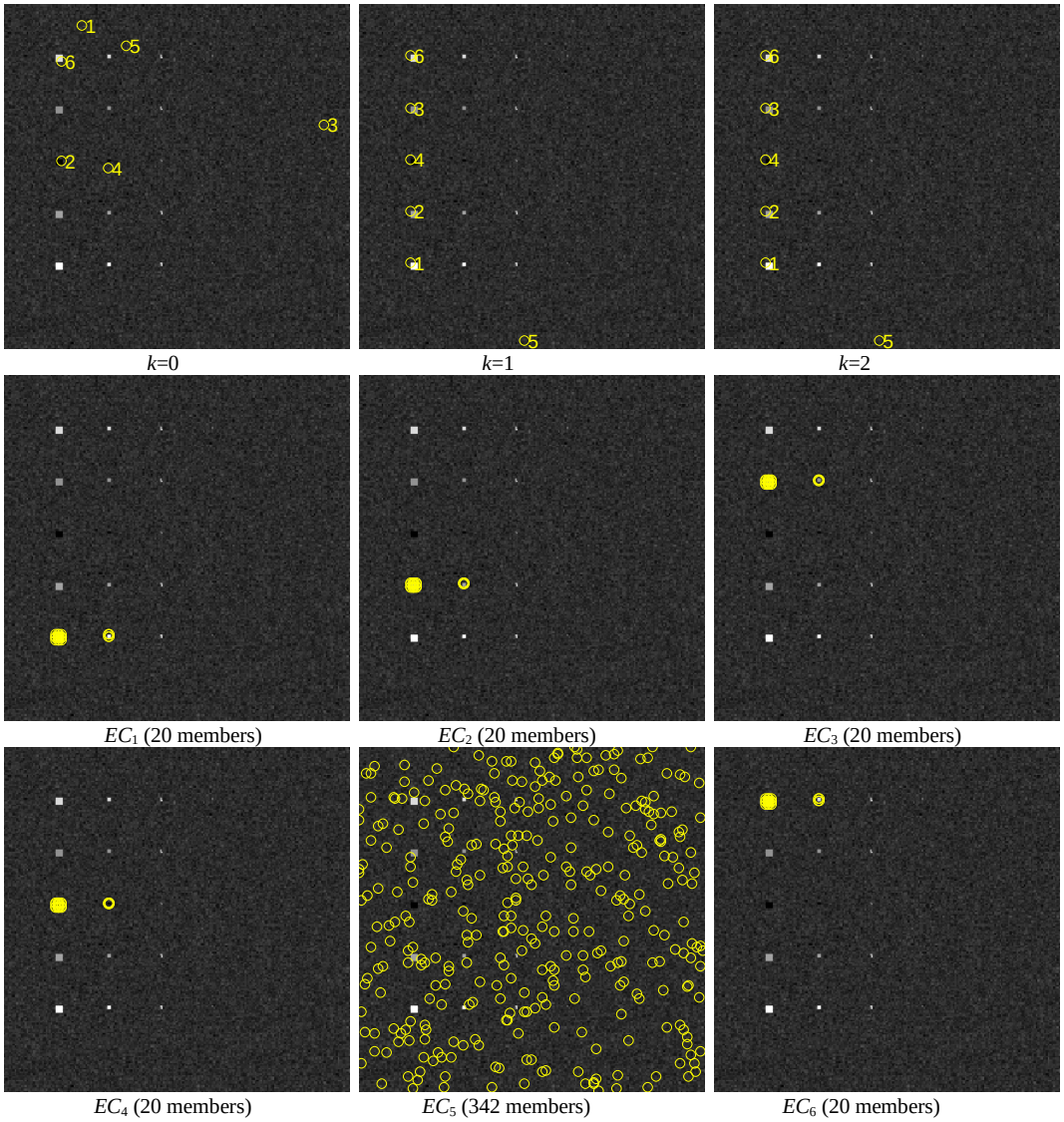
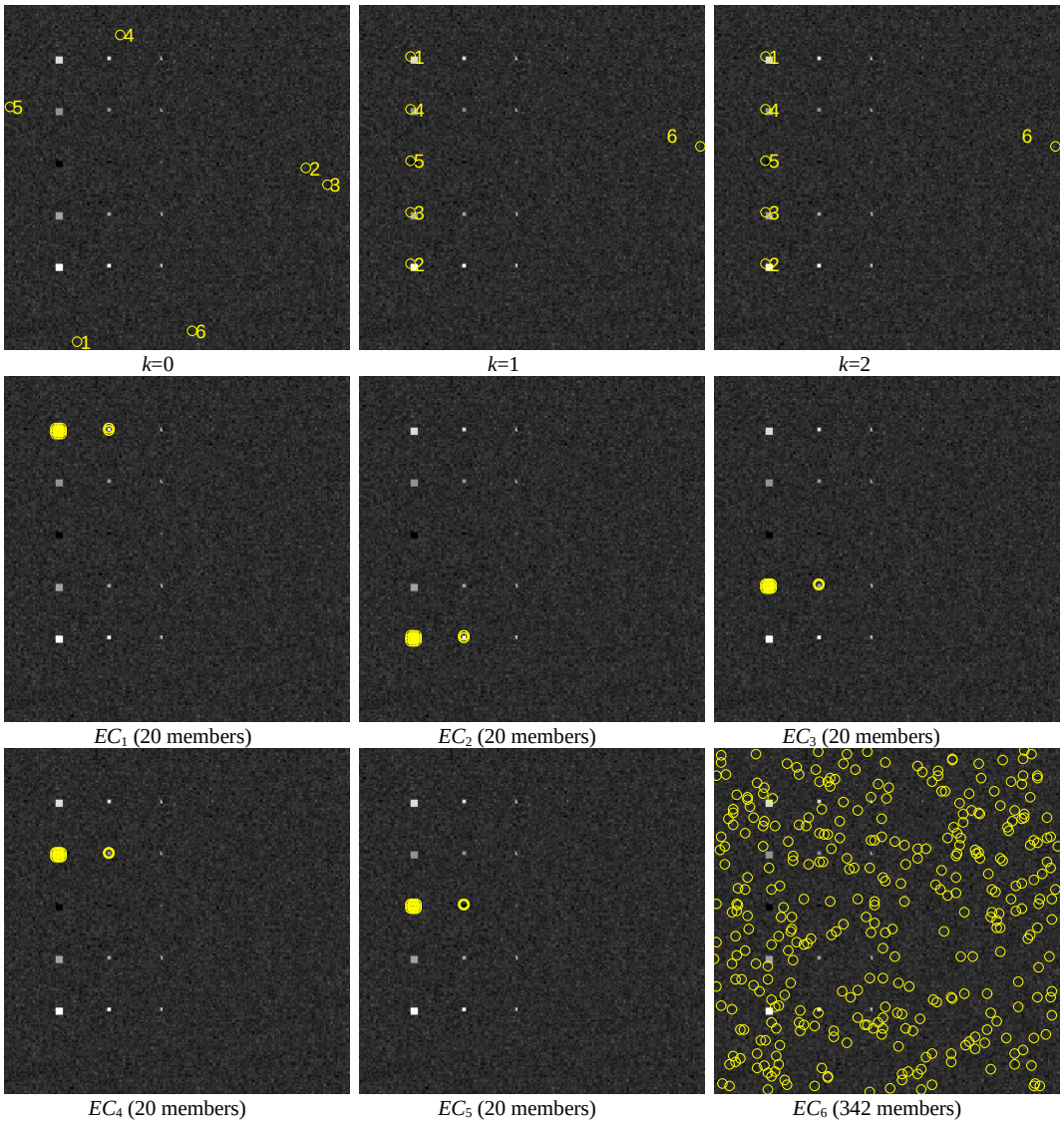


Figure 4.5 TI data: 6 endmembers and their classes by iterative SC EVA-EFA (one random initial condition)

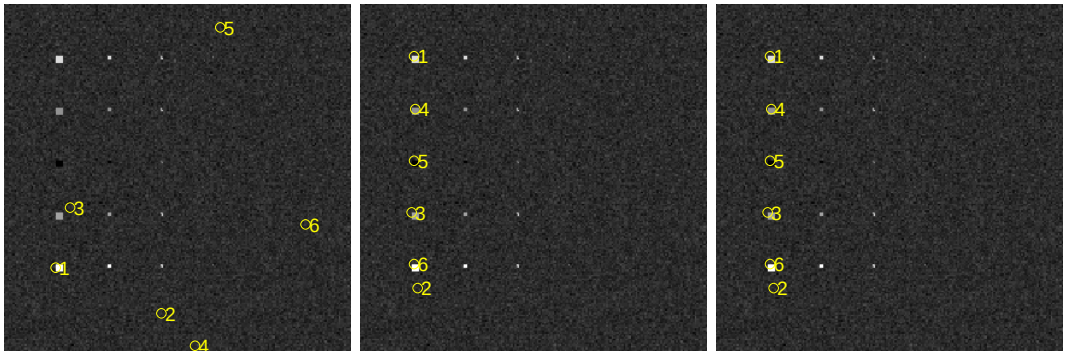


k=0 *k*=1 *k*=2

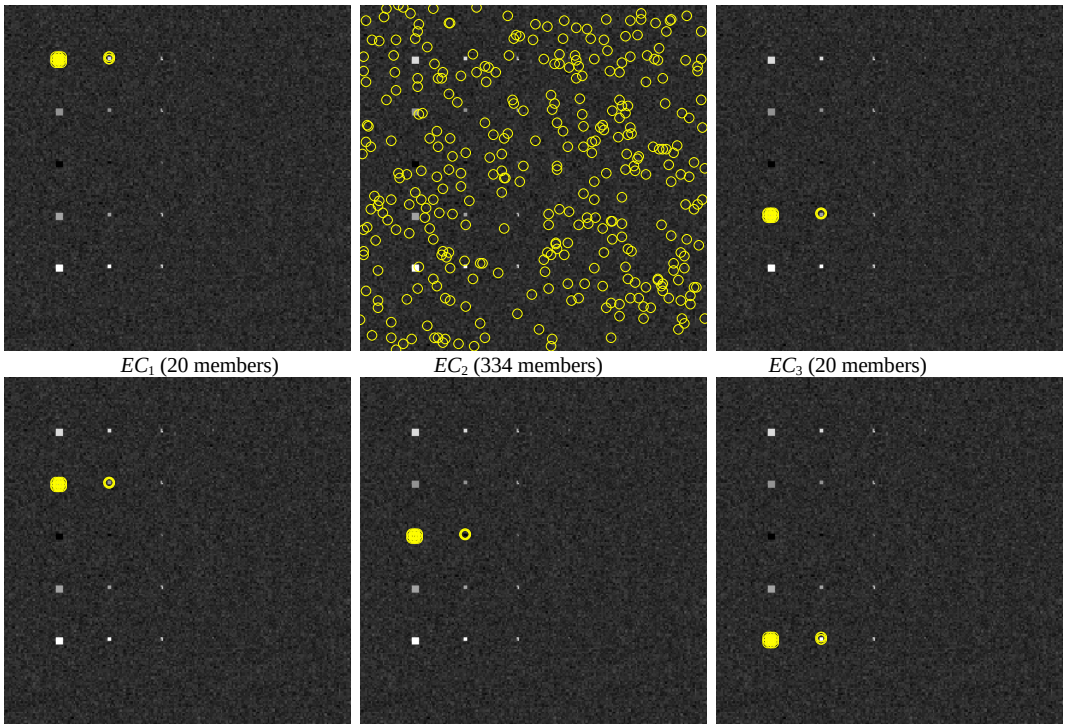
EC_1 (20 members) EC_2 (20 members) EC_3 (20 members)

EC_4 (20 members) EC_5 (20 members) EC_6 (342 members)

Figure 4.6 TI data: 6 endmembers and their classes by iterative SC EVA-EFA (another random initial condition)



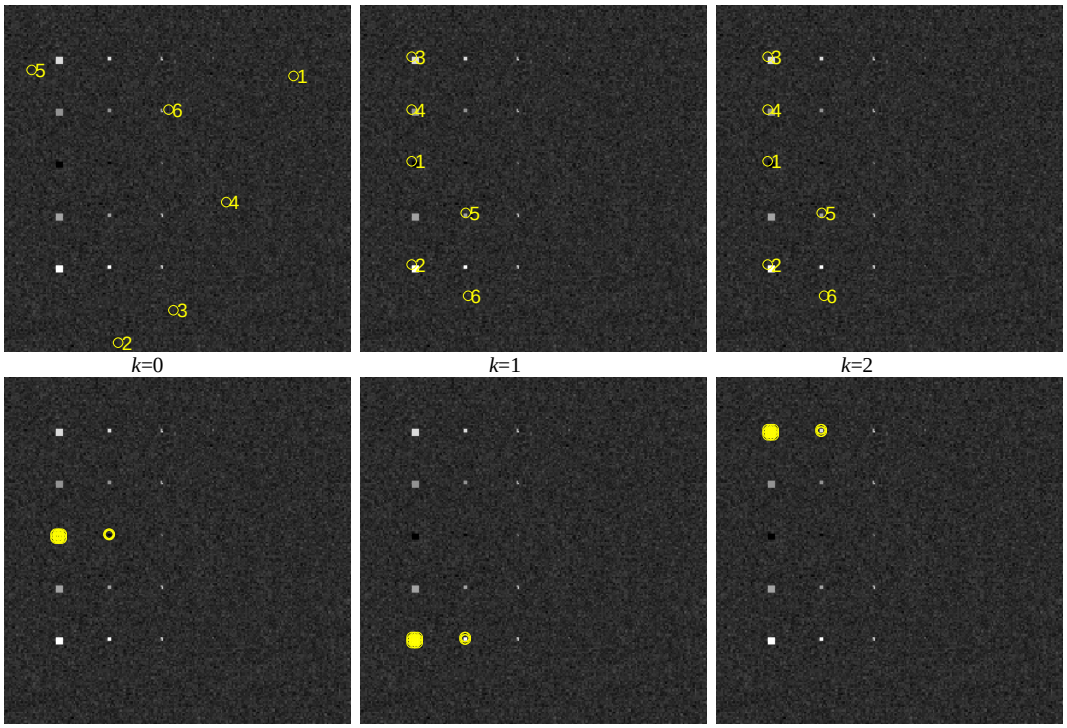
k=0 *k*=1 *k*=2



EC_1 (20 members) EC_2 (334 members) EC_3 (20 members)

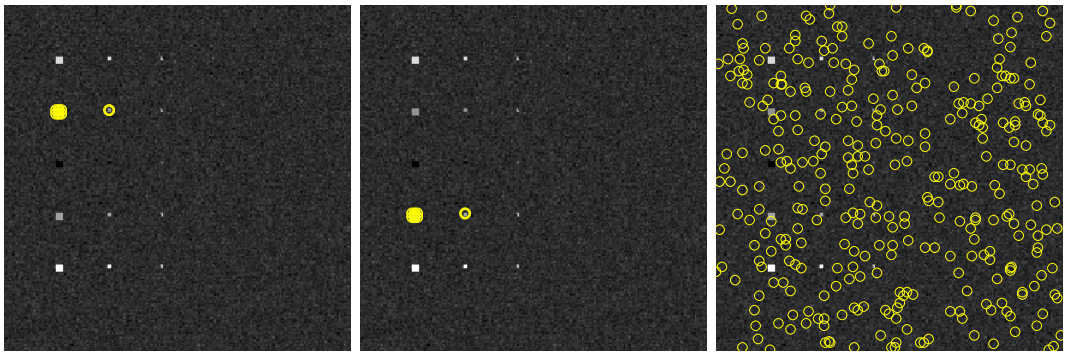
EC_4 (20 members) EC_5 (20 members) EC_6 (20 members)

Figure 4.7 TI data: 6 endmembers and their classes by iterative SQ EVA-EFA (one random initial condition)



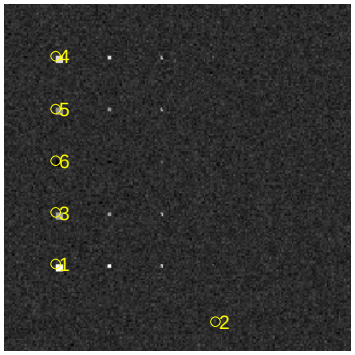
k=0 k=1 k=2

EC_1 (20 members) EC_2 (20 members) EC_3 (20 members)

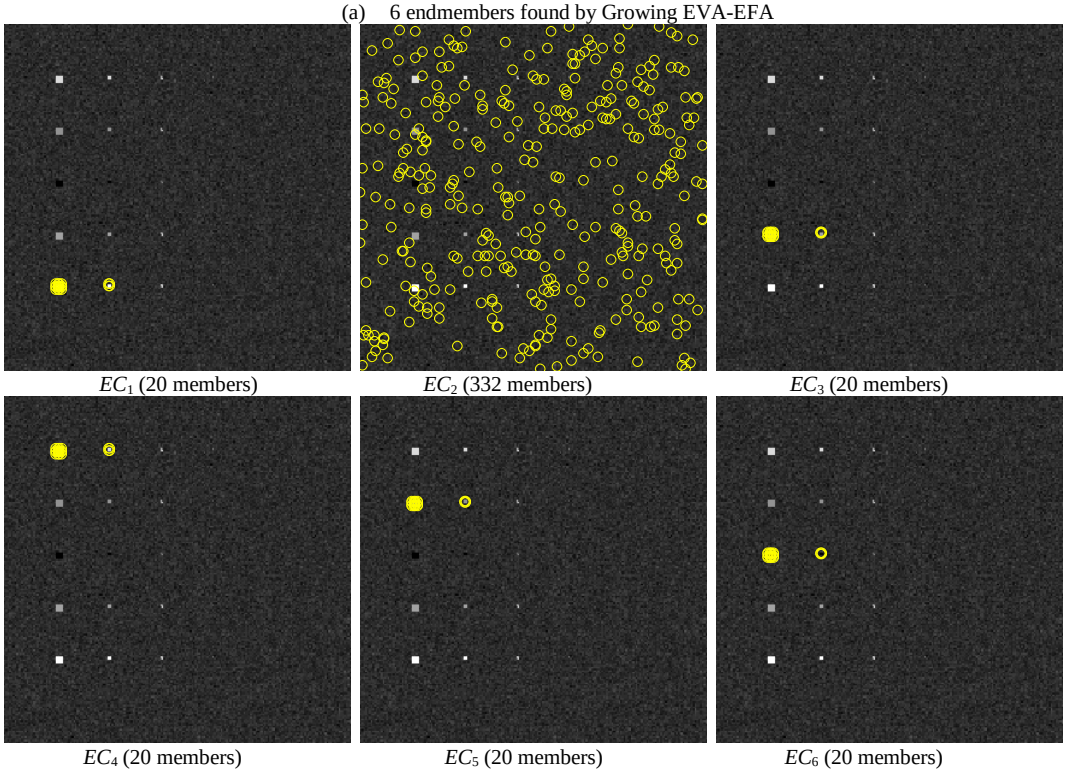


EC_4 (20 members) EC_5 (20 members) EC_6 (332 members)

Figure 4.8 TI data: 6 endmembers and their classes by iterative SQ EVA-EFA (another random initial condition)



(a) 6 endmembers found by Growing EVA-EFA



EC_1 (20 members) EC_2 (332 members) EC_3 (20 members)

EC_4 (20 members) EC_5 (20 members) EC_6 (20 members)

(b) Finding 6 endmember classes

Figure 4.9 TI data: 6 endmembers and their classes by G EVA-EFA

For TE data, the same experiments performed for TI were also conducted. Three methods, including iterative SC EVA-EFA, iterative SQ EVA-EFA and growing EVA-EFA, can found five panel pixels as well as their corresponding endmember classes.

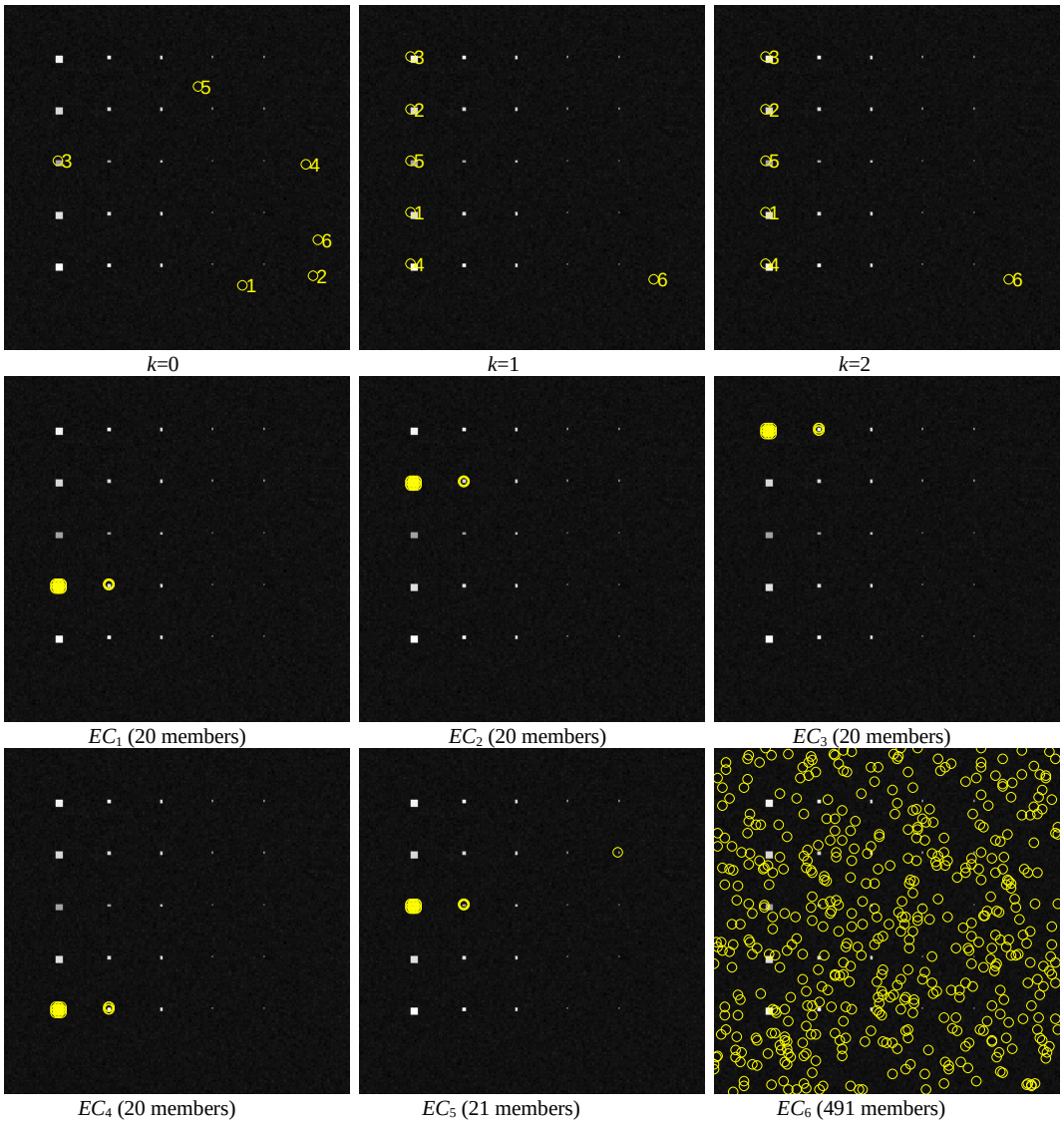


Figure 4.10 TE data: 6 endmembers and their classes by iterative SC EVA-EFA (one random initial condition)

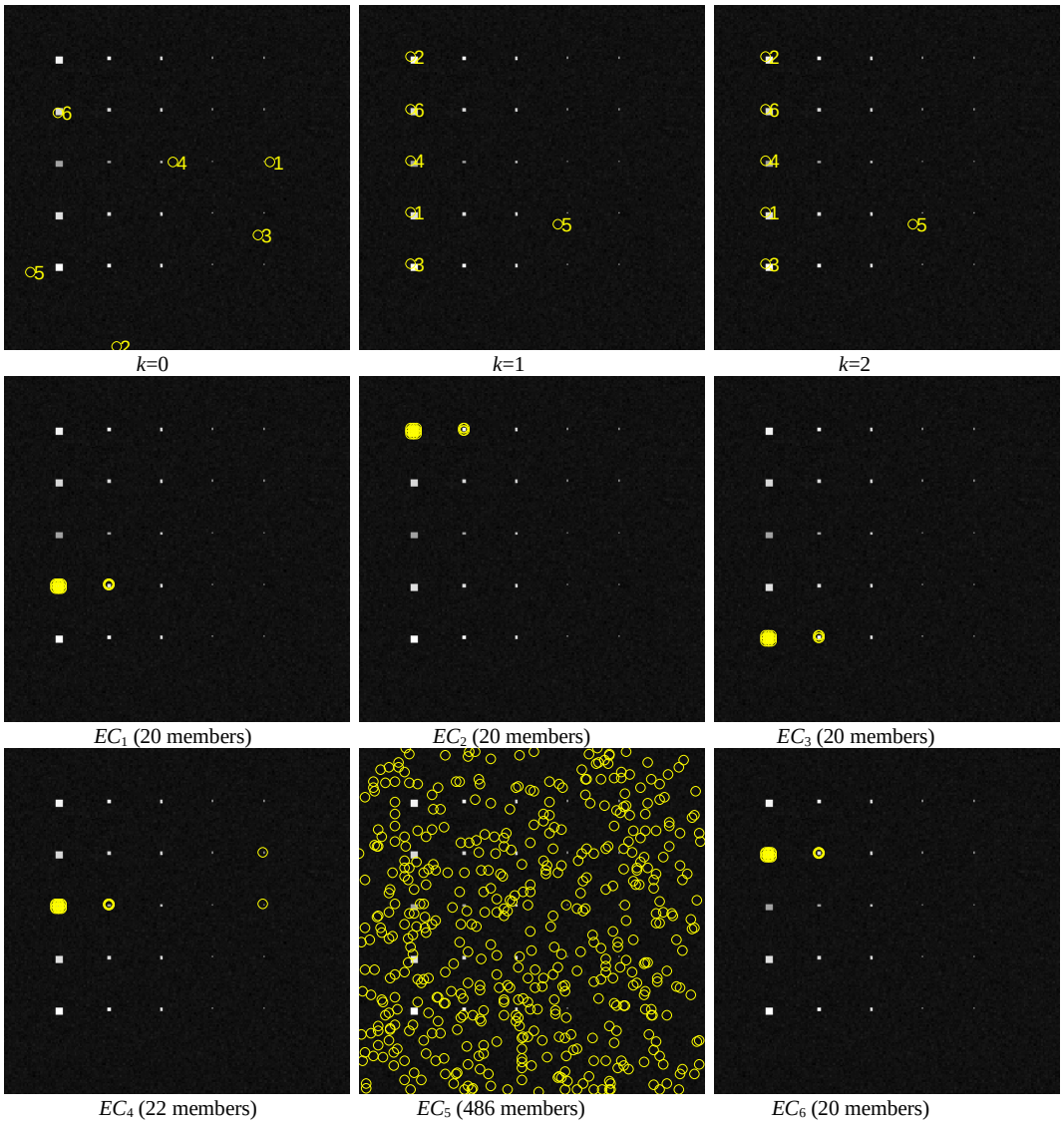
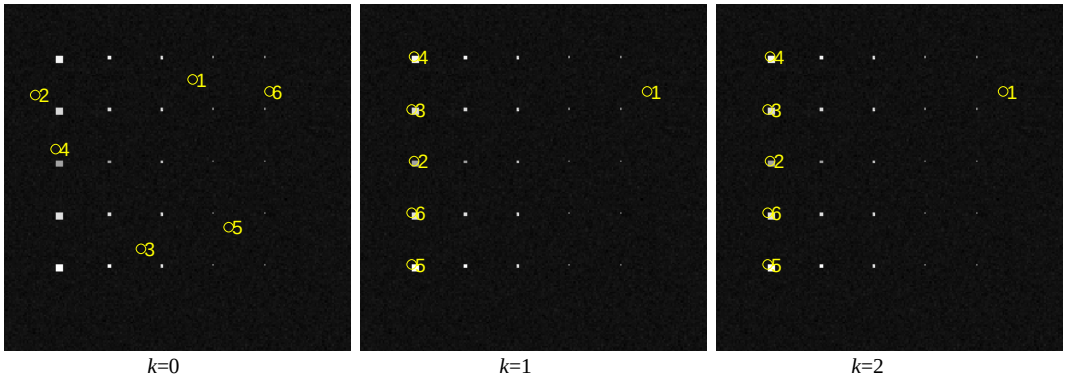
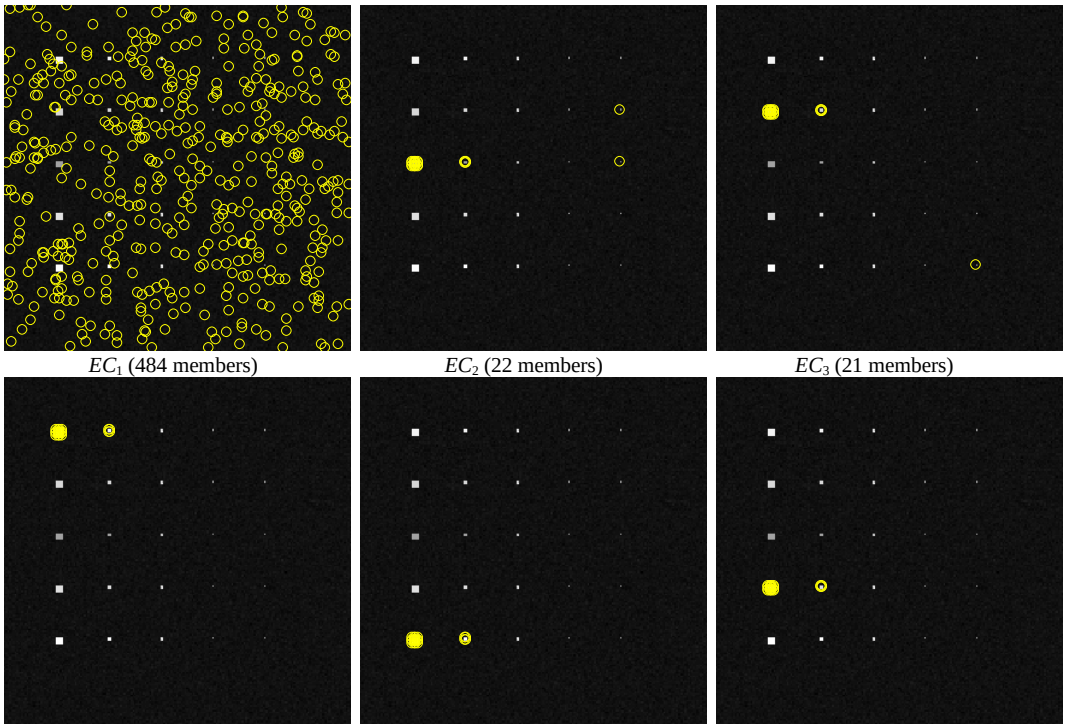


Figure 4.11 TE data: 6 endmembers and their classes by iterative SC EVA-EFA (another random initial condition)

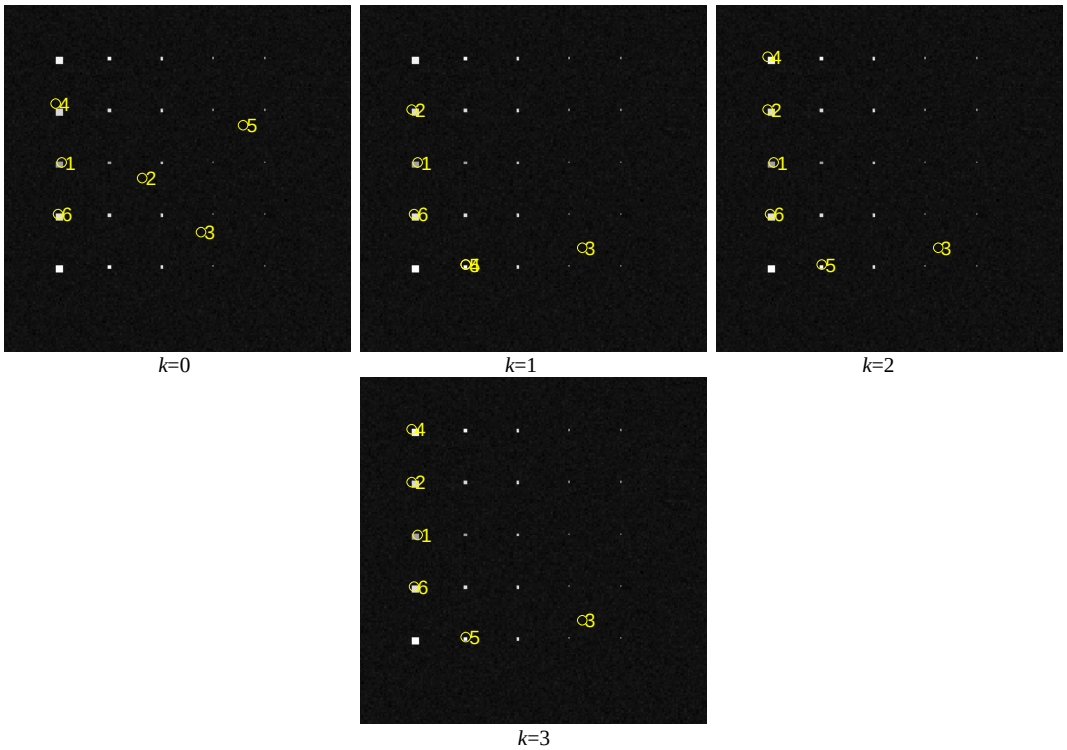




EC_1 (484 members) EC_2 (22 members) EC_3 (21 members)

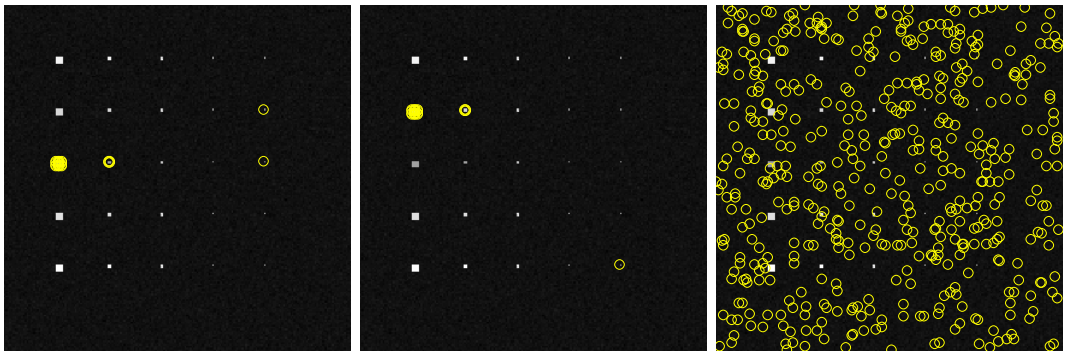
EC_4 (20 members) EC_5 (20 members) EC_6 (20 members)

Figure 4.12 TE data: 6 endmembers and their classes by iterative SQ EVA-EFA (one random initial condition)

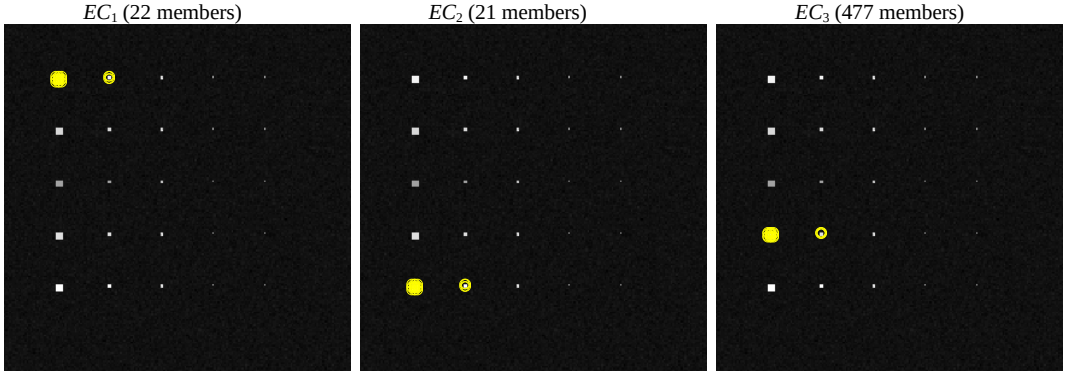


k=0 k=1 k=2

k=3

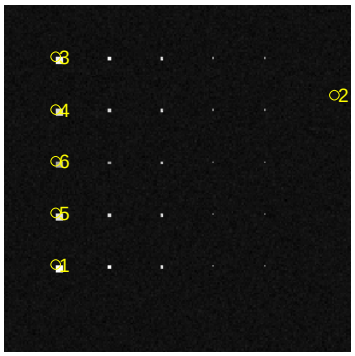


EC_1 (22 members) EC_2 (21 members) EC_3 (477 members)

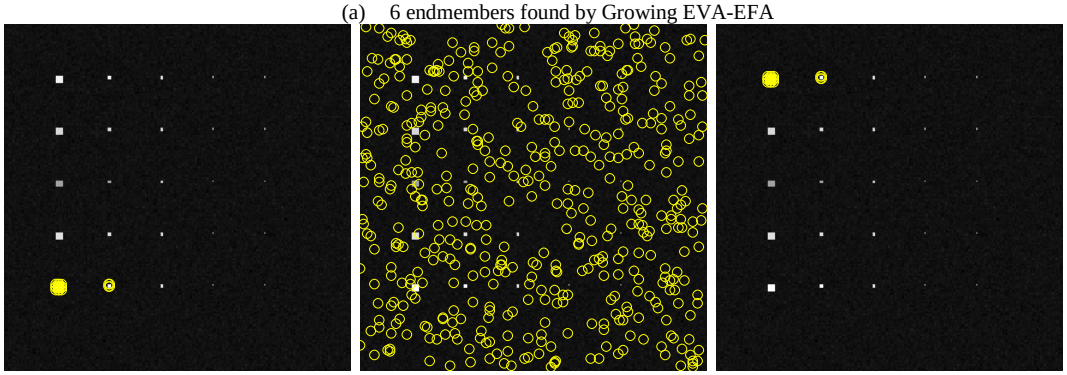


EC_4 (20 members) EC_5 (20 members) EC_6 (20 members)

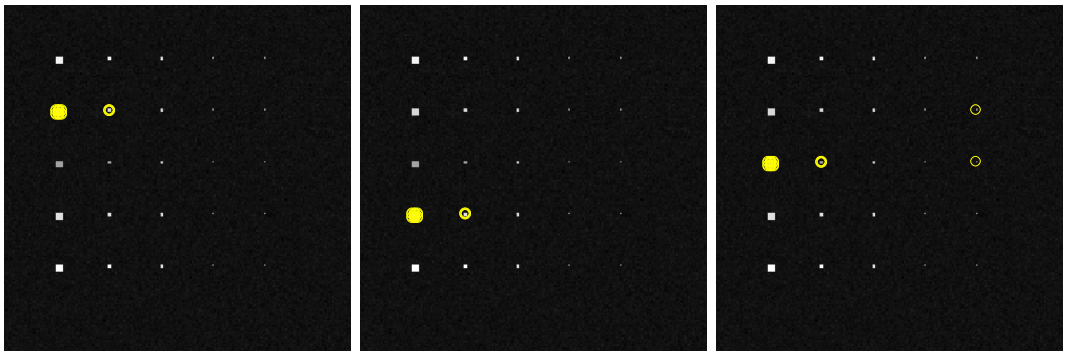
Figure 4.13 TE data: 6 endmembers and their classes by iterative SQ EVA-EFA (another random initial condition)



(a) 6 endmembers found by Growing EVA-EFA



EC_1 (20 members) EC_2 (562 members) EC_3 (20 members)



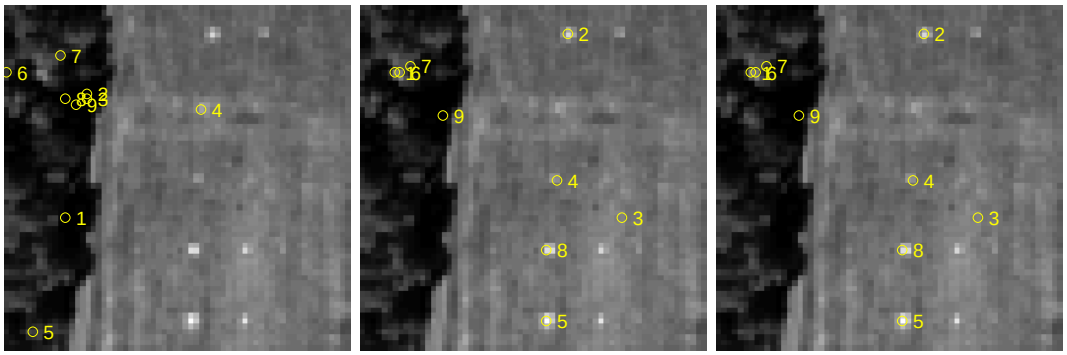
EC_4 (20 members) EC_5 (20 members) EC_6 (22 members)

(b) Finding 6 endmember classes

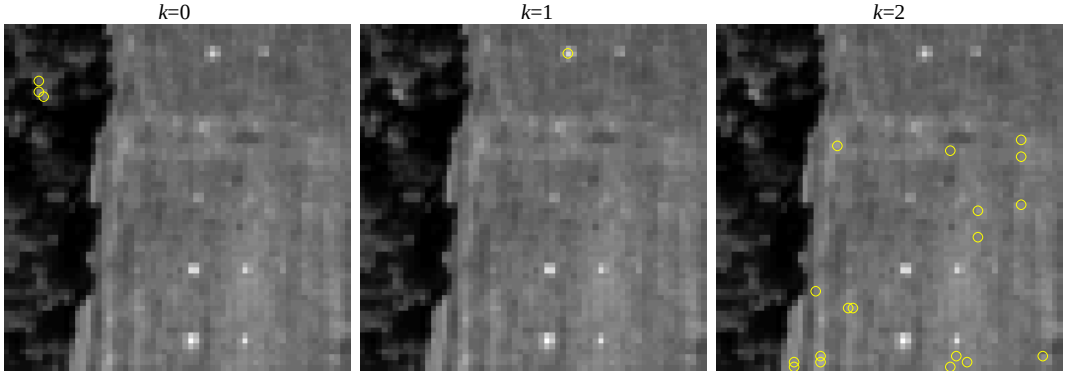
Figure 4.14 TE data: 6 endmembers and their classes by G EVA-EFA

4.5.2 Real Image Experiments

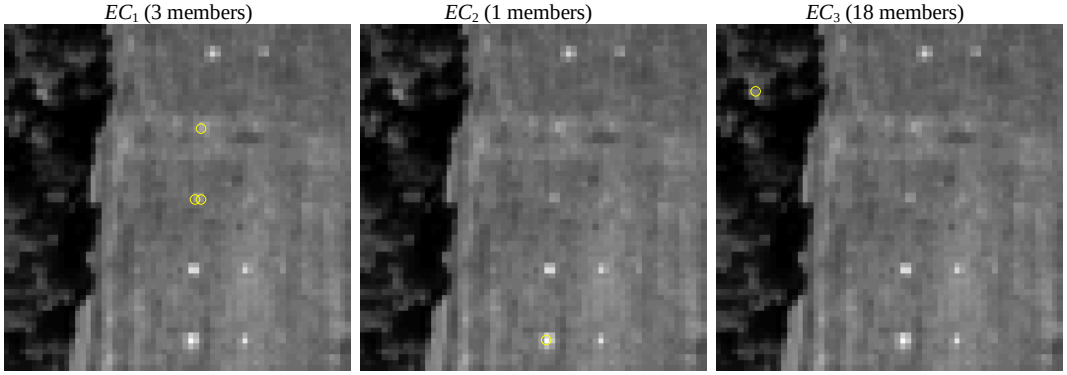
Real image, HYDICE, as introduced in Section 3.5.2, is used for testing effectiveness of algorithms developed in this chapter. As shown in Fig 4.15 and Fig 4.16, iterative SuCcessive endmember variability EFA was used to find endmembers using two different sets of random initial conditions where the final 9 endmembers were the same with different extracted orders but requires the number of iterations denoted by *k*. These results show that the iterative EFA can correct inconsistency caused by random initial conditions. The same experiments done by using iterative SuCcessive EVR EFA in Fig 4.15 and Fig 4.16 were also repeated for iterative SeQuential EVR EFA in Figure 4.17 and Fig 4.18. It takes 3 and 5 iterations to converge for two different initial conditions. Interestingly, comparing the results from iterative SC EVA EFA and iterative SQ EVA EFA, the final set of endmembers are the same and their corresponding endmember classes are also the same except that the order is different.



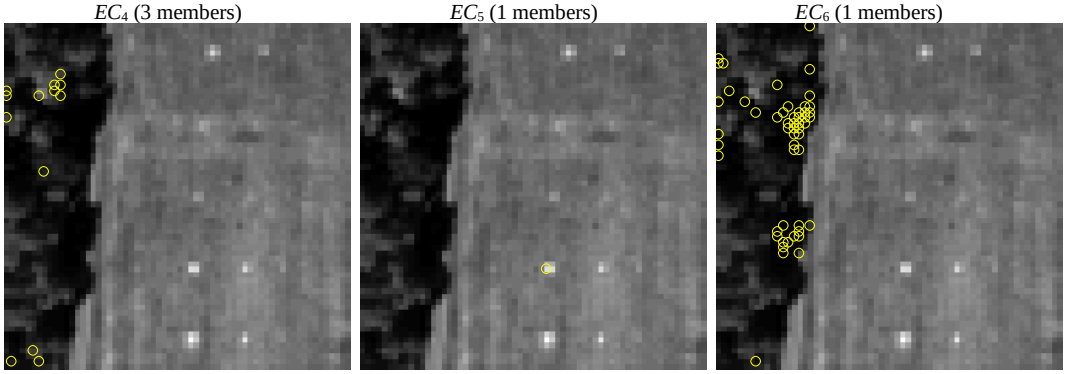
k=0 *k*=1 *k*=2



EC_1 (3 members) EC_2 (1 members) EC_3 (18 members)

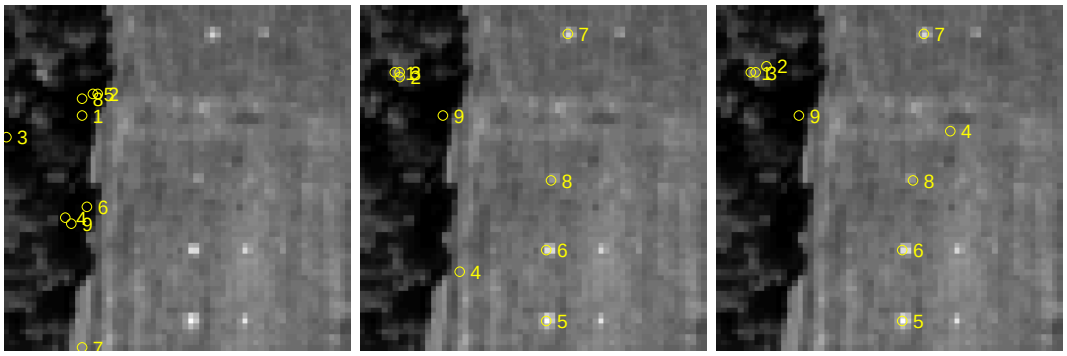


EC_4 (3 members) EC_5 (1 members) EC_6 (1 members)

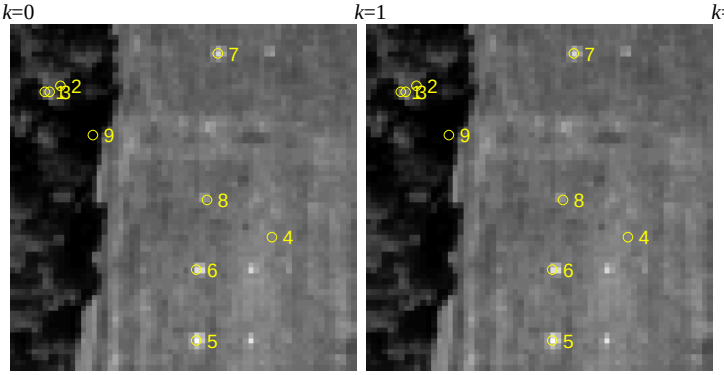


EC_7 (13 members) EC_8 (1 members) EC_9 (52 members)

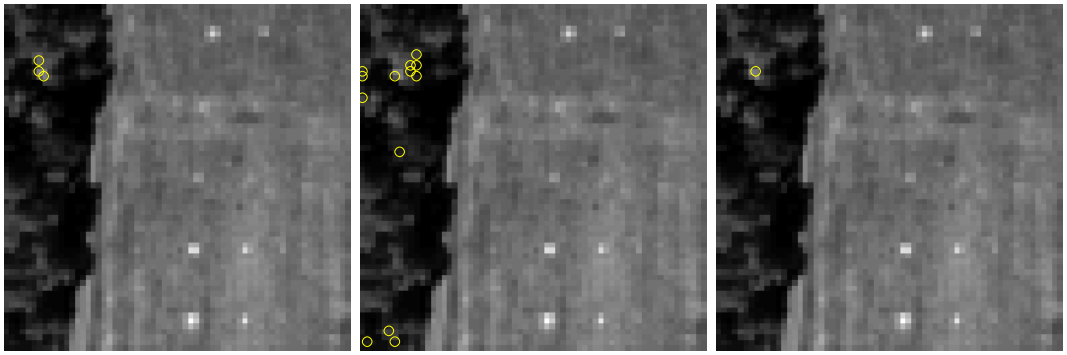
Figure 4.15 HYDICE: 9 endmembers and their classes by iterative SC EVA-EFA (one random initial condition)



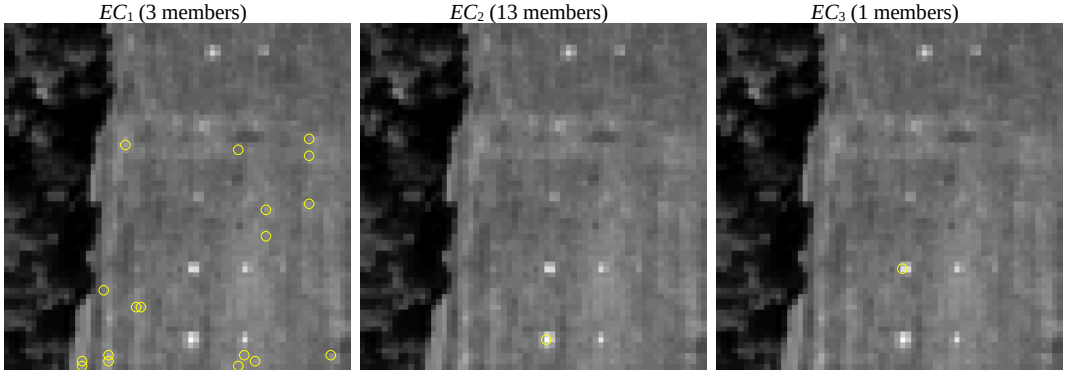
k=0 *k*=1 *k*=2



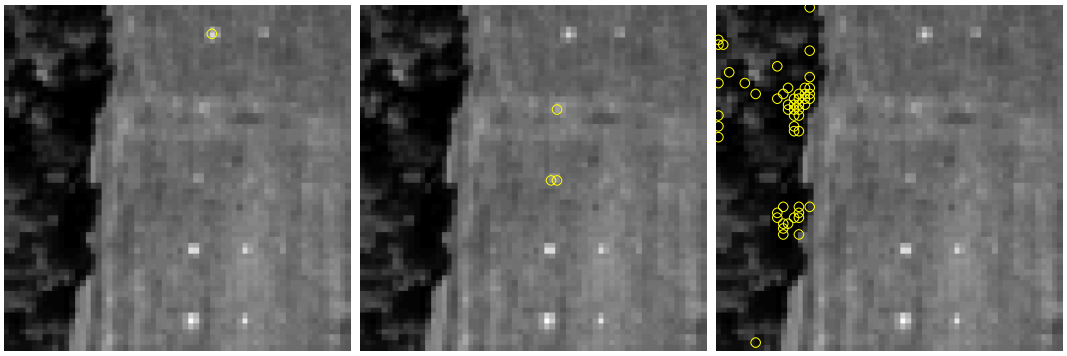
k=3 *k*=4



EC_1 (3 members) EC_2 (13 members) EC_3 (1 members)

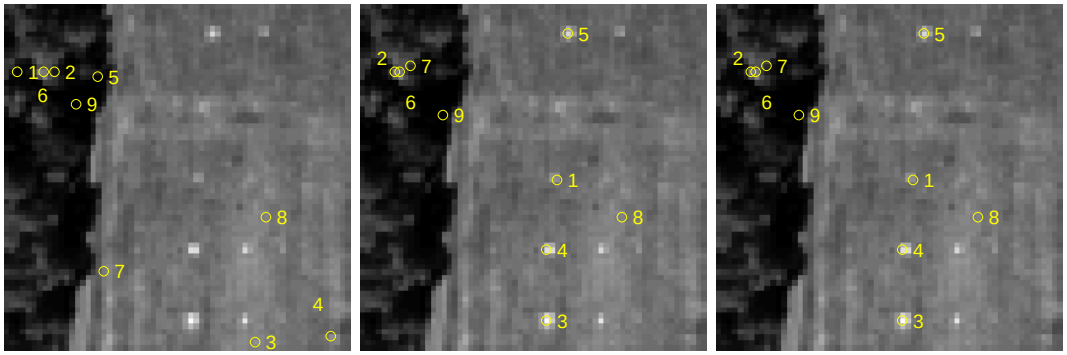


EC_4 (18 members) EC_5 (1 members) EC_6 (1 members)

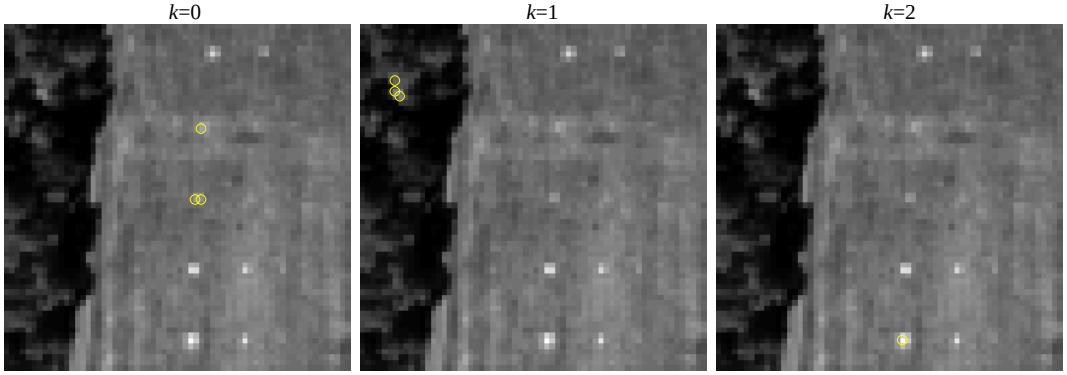


EC_7 (1 members) EC_8 (3 members) EC_9 (52 members)

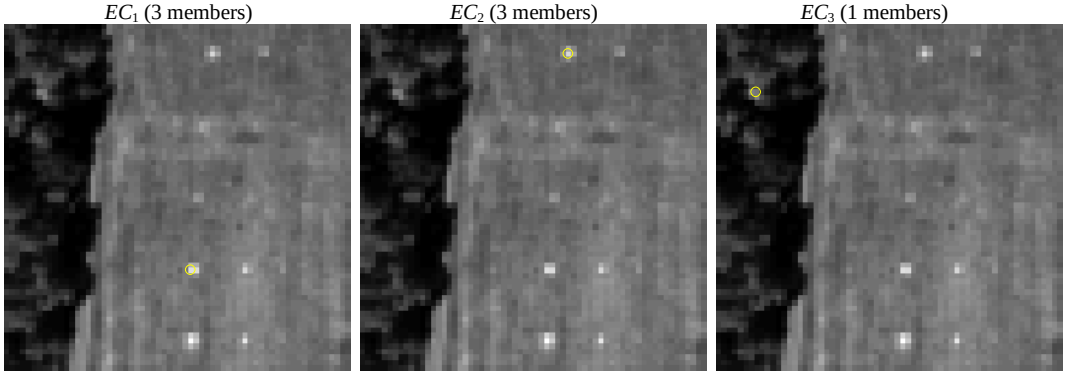
Figure 4.16 HYDICE: 9 endmembers and their classes by iterative SC EVA-EFA (another random initial condition)



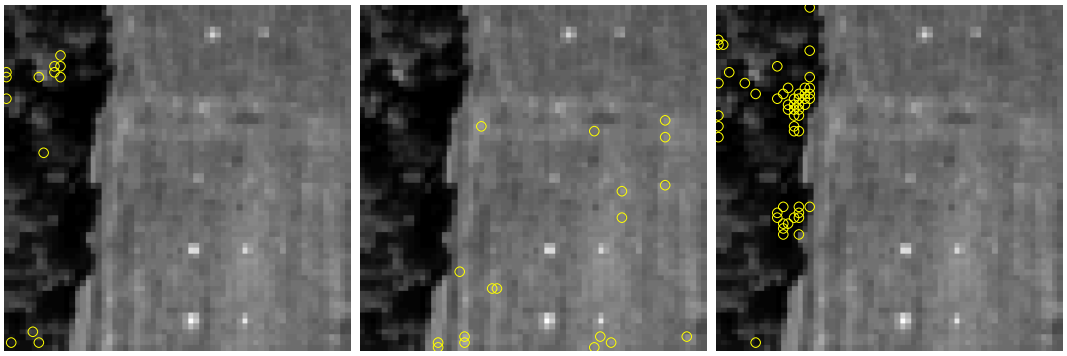
k=0 k=1 k=2



EC_1 (3 members) EC_2 (3 members) EC_3 (1 members)

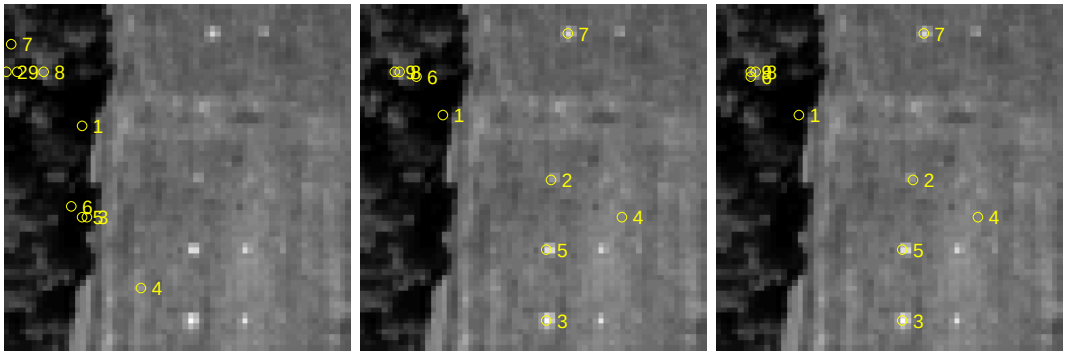


EC_4 (1 members) EC_5 (1 members) EC_6 (1 members)

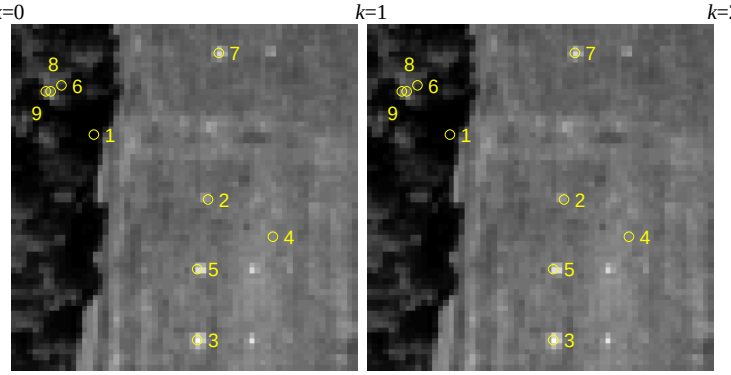


EC_7 (13 members) EC_8 (18 members) EC_9 (52 members)

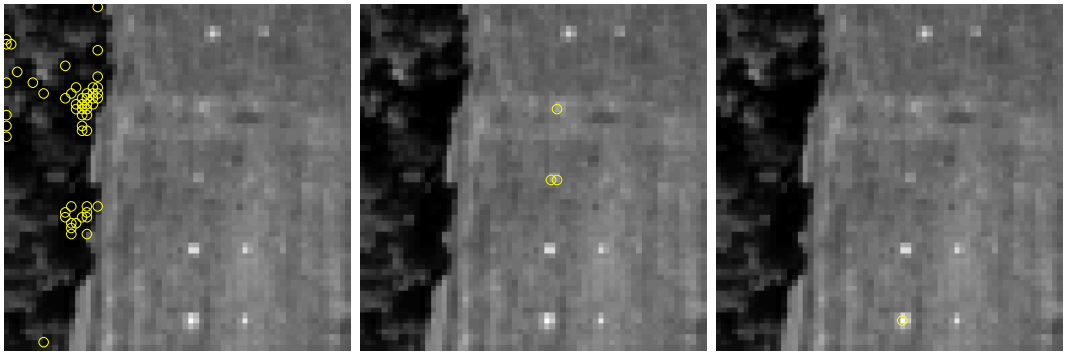
Figure 4.17 HYDICE: 9 endmembers and their classes by iterative SQ EVA-EFA (one random initial condition)



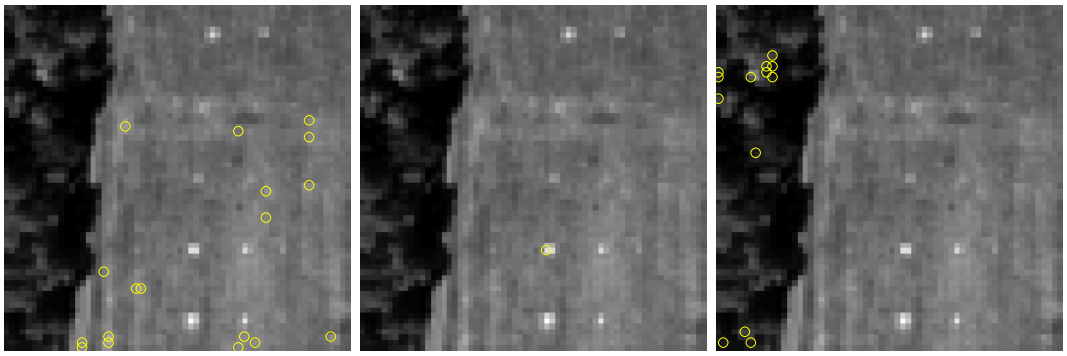
k=0 *k*=1 *k*=2



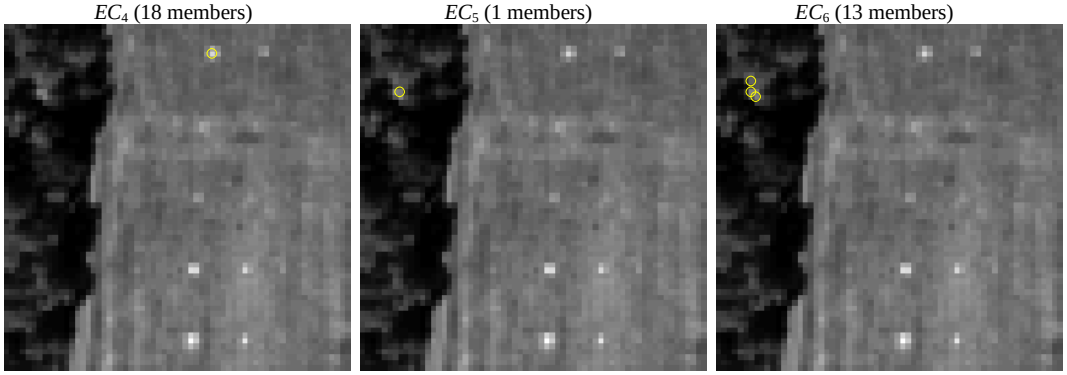
k=3 *k*=4



EC_1 (52 members) EC_2 (3 members) EC_3 (1 members)

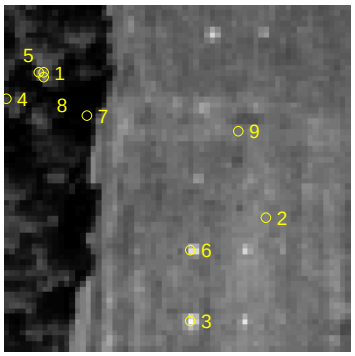


EC_4 (18 members) EC_5 (1 members) EC_6 (13 members)

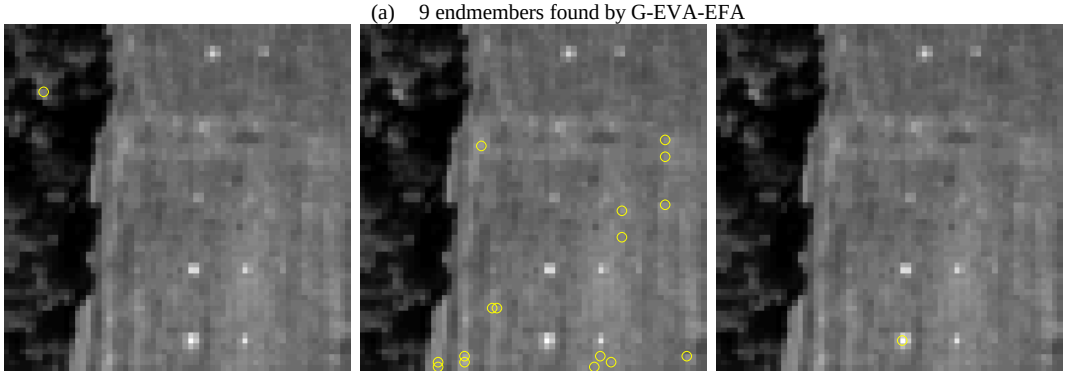


EC_7 (1 members) EC_8 (1 members) EC_9 (3 members)

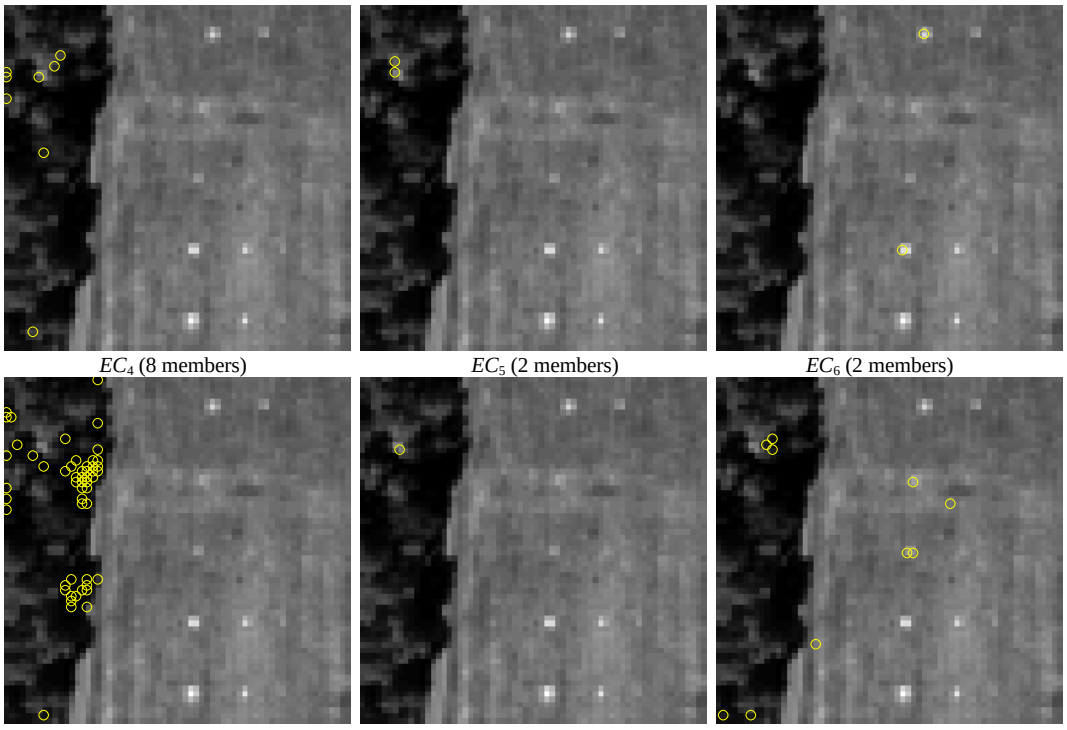
Figure 4.18 HYDICE: 9 endmembers and their classes by iterative SQ EVA-EFA (another random initial condition)



(a) 9 endmembers found by G-EVA-EFA



EC_1 (1 members) EC_2 (16 members) EC_3 (1 members)



EC_4 (8 members) EC_5 (2 members) EC_6 (2 members)

EC_7 (52 members) EC_8 (1 members) EC_9 (10 members)

Figure 4.19 HYDICE: 9 endmembers and their classes by G EVA-EFA

Another real image (LCVF) is used for testing effectiveness of iterative SC EVA EFA, iterative SQ EVA EFA and growing EVA EFA. The detailed introduction of LCVF is shown in chapter 3. The Iterative SC EVA-EFA, iterative SQ EVA-EFA and growing EVA-EFA were also tested on LCVF data. Results are shown in Fig. 4.23-Fig. 4.27. As we can see from these figures, dry lake and two anomaly pixels were found. However, the algorithms considered cinders and shade as a single endmember class , while vegetation and rhyolite as another single endmember class. The reason for this is that cinders and shade have very similar spectral profiles and so do the vegetation and rhyolite.

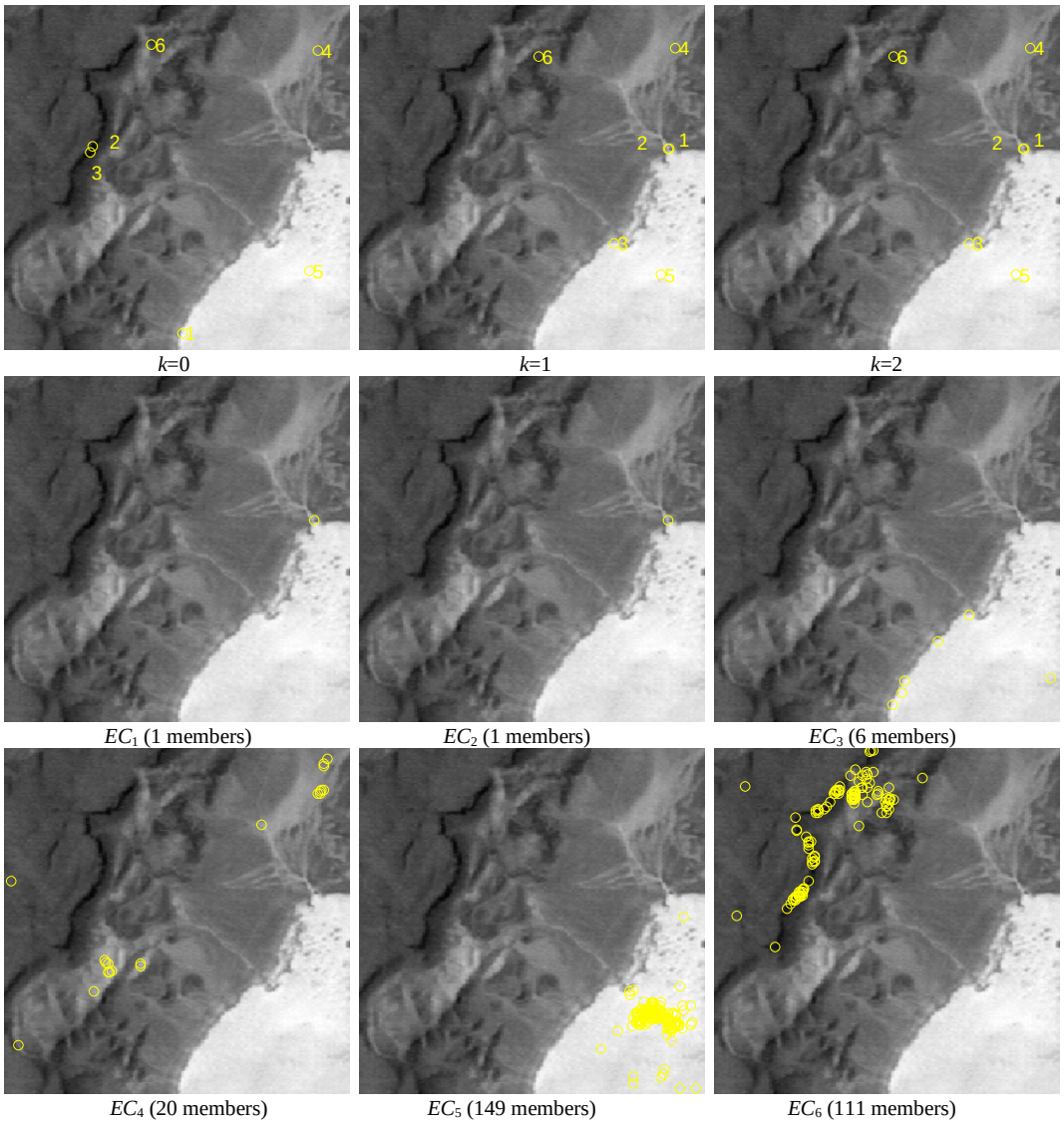
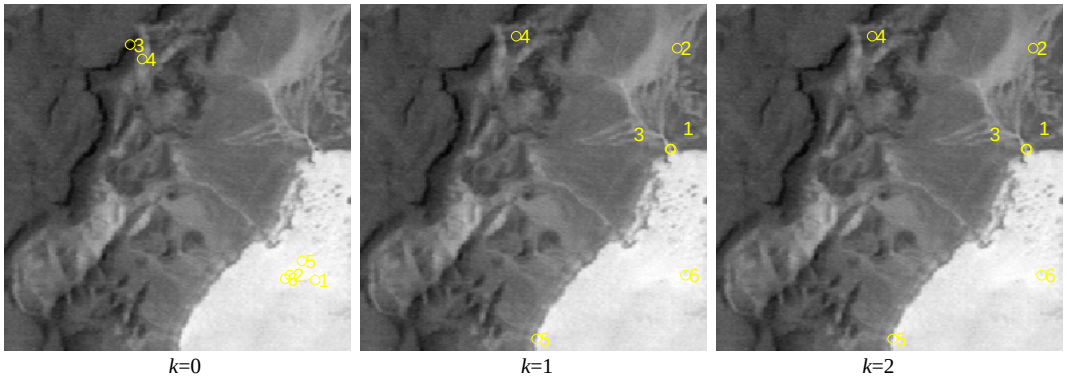
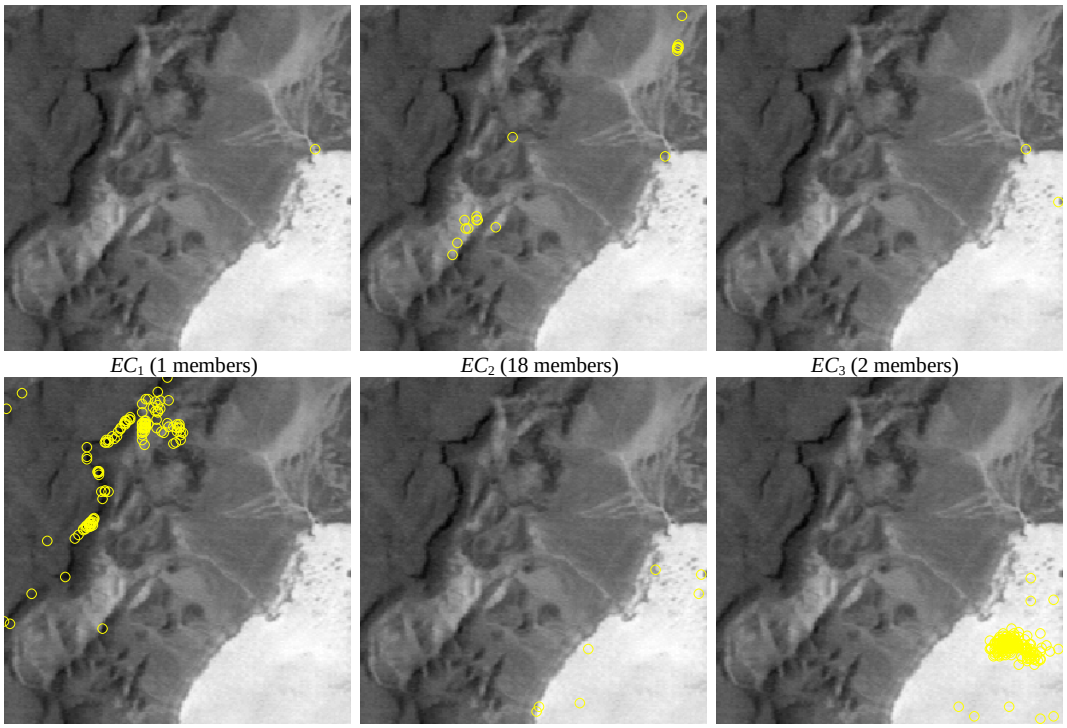


Figure 4.20 LCVF: 6 endmembers and their classes by iterative SC EVA-EFA (one random initial condition)

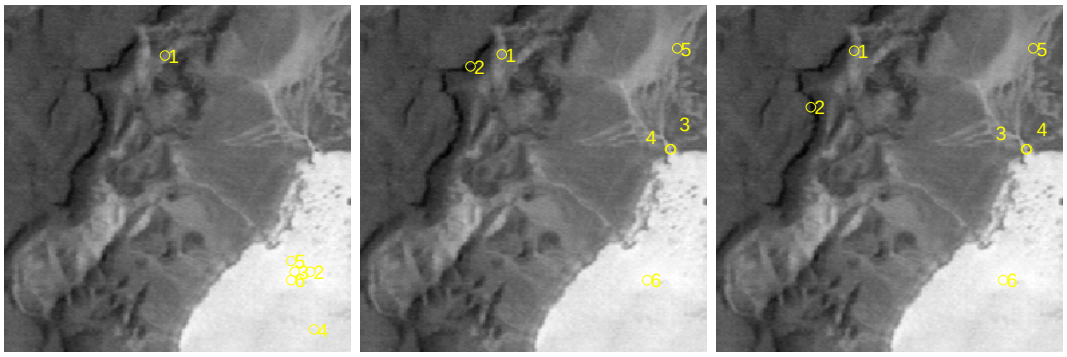




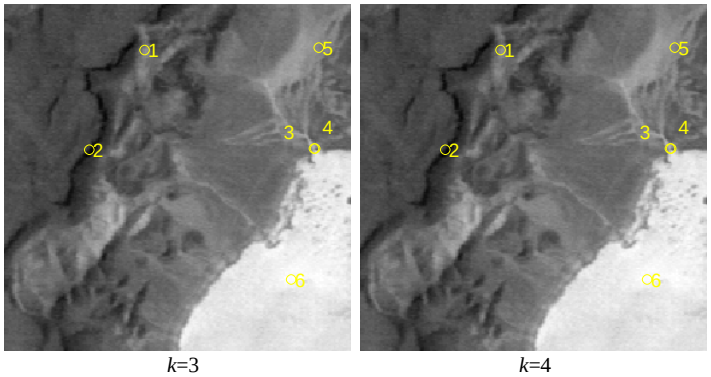
EC_1 (1 members) EC_2 (18 members) EC_3 (2 members)

EC_4 (117 members) EC_5 (7 members) EC_6 (152 members)

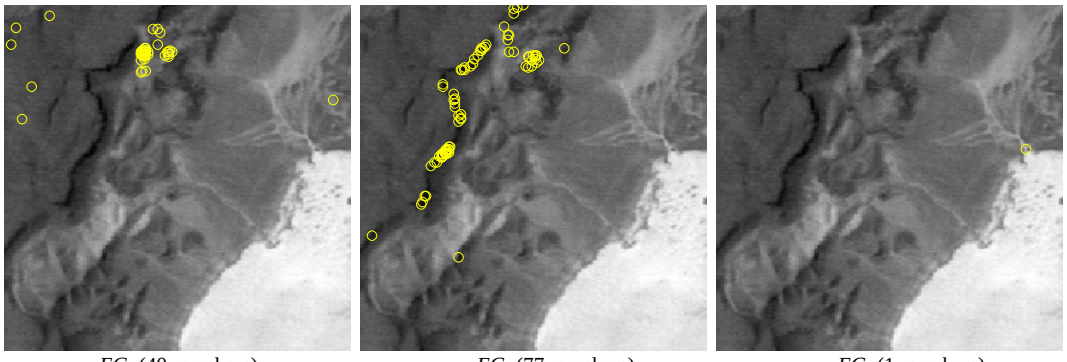
Figure 4.21 LCVF: 6 endmembers and their classes by iterative SC EVA-EFA (another random initial condition)



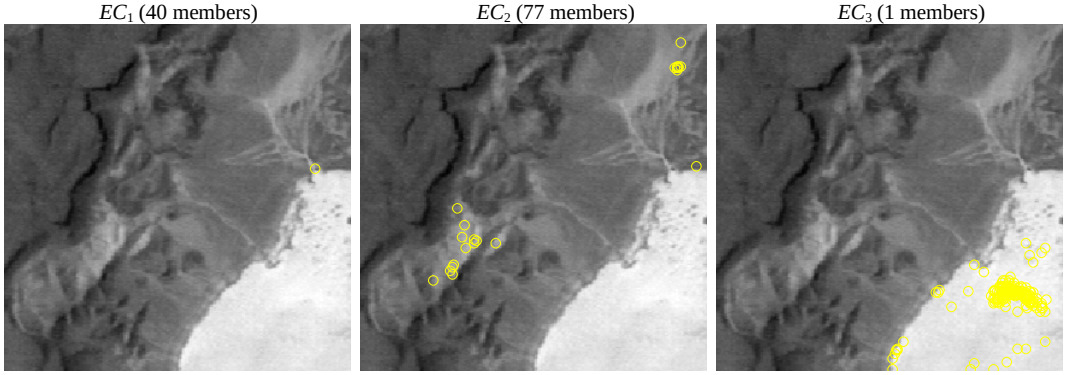
k=0 *k*=1 *k*=2



k=3 *k*=4

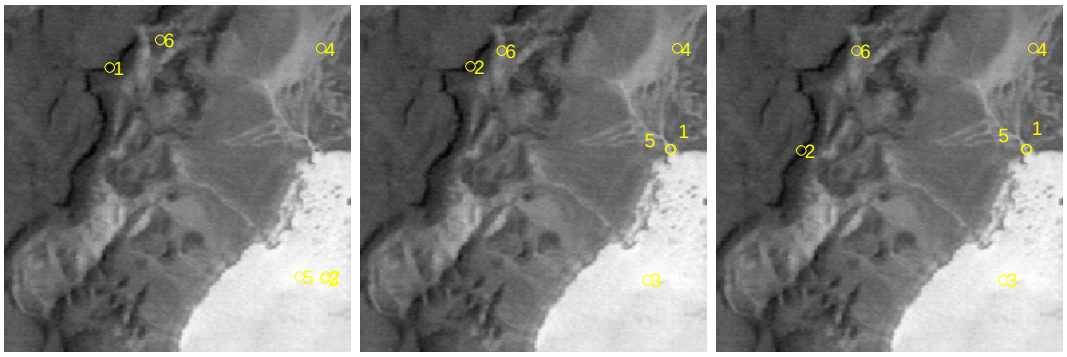


EC_1 (40 members) EC_2 (77 members) EC_3 (1 members)

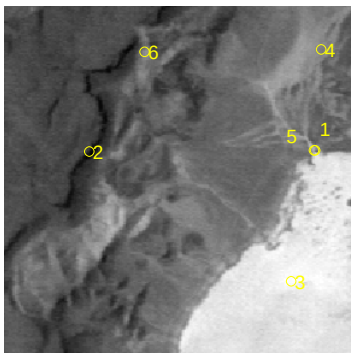


EC_4 (1 members) EC_5 (22 members) EC_6 (157 members)

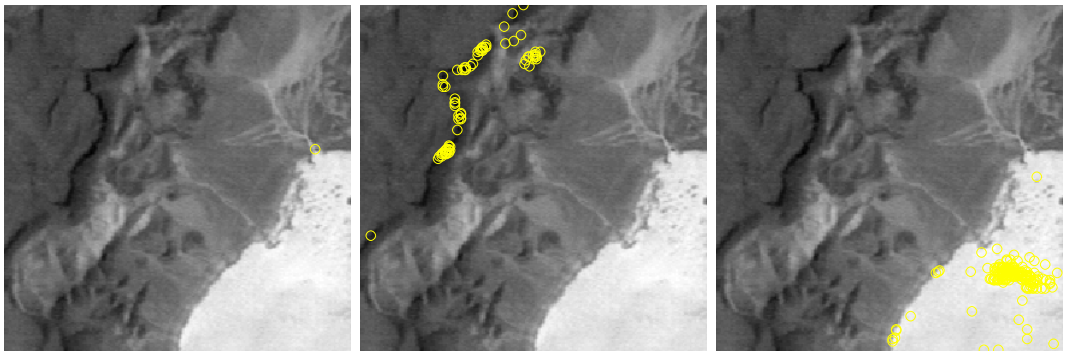
Figure 4.22 LCVF: 6 endmembers and their classes by iterative SQ EVA-EFA (one random initial condition)



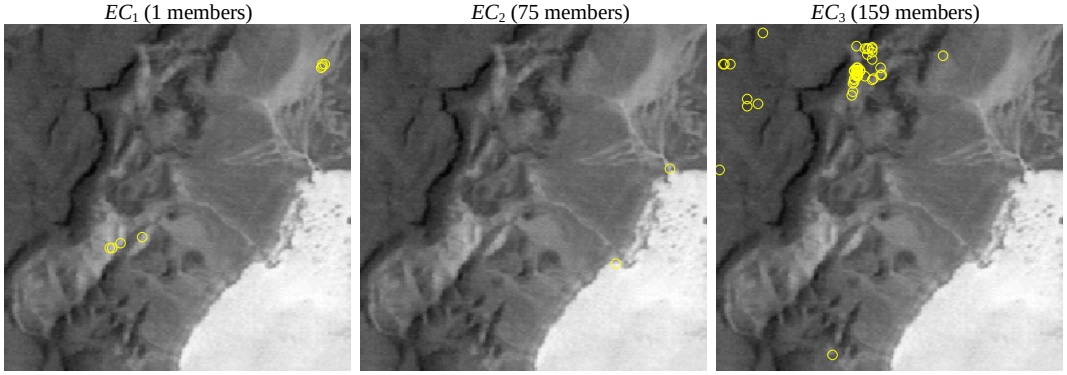
k=0 *k*=1 *k*=2



k=3

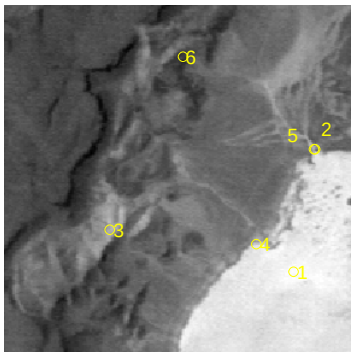


EC_1 (1 members) EC_2 (75 members) EC_3 (159 members)

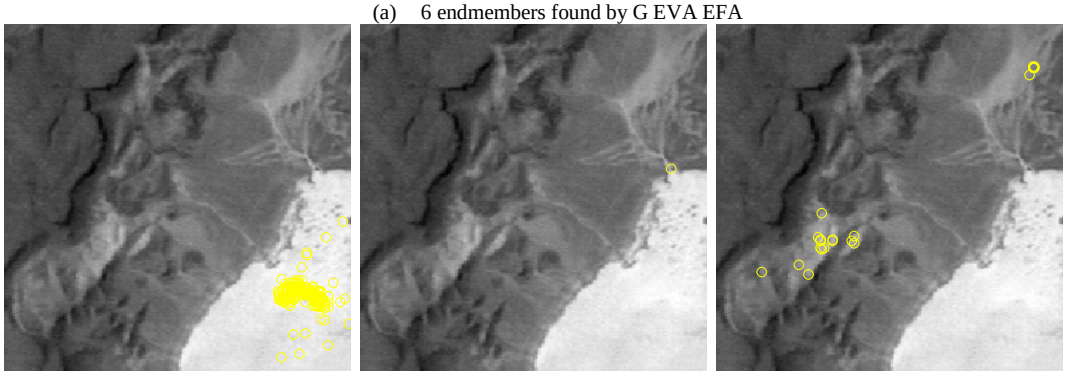


EC_4 (8 members) EC_5 (2 members) EC_6 (47 members)

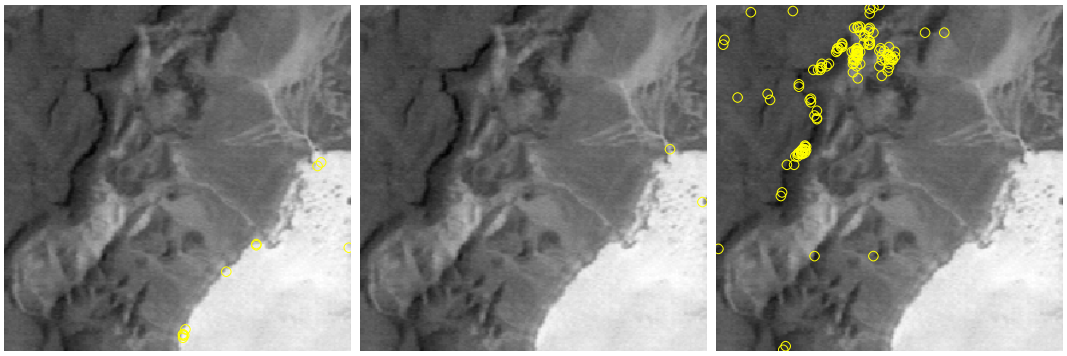
Figure 4.23 LCVF: 6 endmembers and their classes by iterative SQ EVA EFA (another random initial condition)



(a) 6 endmembers found by G EVA EFA



EC_1 (165 members) EC_2 (1 members) EC_3 (20 members)



EC_4 (11 members) EC_5 (2 members) EC_6 (30 members)

(b) Finding 6 endmember classes

Figure 4.24 LCVF: 6 endmembers and their classes by G EVA-EFA

4.6 Conclusion

Endmember variability has been an issue in finding endmembers since there is no guarantee that endmembers will be present in the data to be processed. This chapter develops a new concept to deal with the endmember variability issue, referred to as endmember variability ratio (EVR)-based approach which is derived from Fisher's ratio widely used in Fisher linear discriminant analysis. By taking advantage of EVR several EVR-based endmember algorithms (EVR-EFAs) algorithms can be designed and developed. In particular, the well-known N-FINDR can be further extended as a special case of EVR-EFAs.

Chapter 5 Fully Abundance-Constrained Linear Spectral Mixture Analysis for Finding Endmembers

5.1 Introduction

Linear spectral mixture analysis (LSMA) has been widely used for linear spectral unmixing. Its application to finding endmembers has recently received considerable interest. An early attempt to use LSMA to find endmembers is N-FINDR developed by Winter [39] who calculated the maximal simplex volume to find endmembers where data sample vectors embraced by a simplex correspond to data sample vectors that can be unmixed by Fully Constrained Least Squares (FCLS) with full abundance constraints, abundance sum-to-one constraint (ASC) and abundance non-negativity constraint (ANC). So, with this interpretation N-FINDR can be considered as unsupervised FCLS which finds a set of signatures to be used to unmix data sample vectors in an unsupervised manner. In this chapter, a different approach is taken to develop LSMA-based. Different versions of algorithms, including SeQuential FCLS (SQ FCLS) and SuCcessive FCLS (SC FCLS), are developed. In order to deal with random issues in initial conditions, two versions of FCLS, referred to as Iterative FCLS (IFCLS) and Random FCLS (RFCLS) are also developed.

5.2 Fully Constrained Least Squares

Assume that $\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p$ are signatures used to form a linear mixing model for LSMA. A linear mixture model for $\mathbf{r}$ models the spectral signature of $\mathbf{r}$ as a linear

combination of $\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p$ with appropriate abundance fractions specified by $\alpha_1, \alpha_2, \cdots, \alpha_p$, as follows

$$\mathbf{r} = \mathbf{M}\boldsymbol{\alpha} + \mathbf{n} \tag{5.1}$$

where $\mathbf{r}$ is an $L \times 1$ column vector, $\mathbf{M}$ is an $L \times p$ substance spectral signature matrix, denoted by $[\mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_p]$, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_p)^T$ is a $p \times 1$ abundance column vector associated with $\mathbf{r}$ with α_j being the abundance fraction of the j^{th} substance signature $\mathbf{e}_j$ present in the pixel vector $\mathbf{r}$ and $\mathbf{n}$ is noise or can be interpreted as a measurement or model error.

A classical approach to solving (5.1) is the least squares estimation given by

$$\hat{\boldsymbol{\alpha}}^{LS}(\mathbf{r}; \mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p) = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{r} \tag{5.2}$$

where $\hat{\boldsymbol{\alpha}}^{LS}(\mathbf{r}; \mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p) = (\hat{\alpha}_1^{LS}(\mathbf{r}; \mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p), \cdots, \hat{\alpha}_p^{LS}(\mathbf{r}; \mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p))$ and $\hat{\alpha}_j^{LS}(\mathbf{r}, \mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p)$ is the abundance fraction of j^{th} substance signature $\mathbf{e}_j$ estimated from the data sample vector $\mathbf{r}$. Here the abundance estimation is determined by $\mathbf{r}$ and $\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p$. By means of (5.2) can be reconstructed by

$$\hat{\mathbf{r}}(\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p) = \mathbf{M}\hat{\boldsymbol{\alpha}}(\mathbf{r}; \mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p) \tag{5.3}$$

By virtue of (5.3), we can calculate the least square error (LSE) resulting from using FCLS to unmix the data sample vector $\mathbf{r}$ using a set of p endmembers, $\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p$ as the averaged unmixing error (UME)

$$\mathrm{UME}(\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p) = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{r}_i - \hat{\mathbf{r}}_i(\mathbf{e}_1, \mathbf{e}_2, \cdots, \mathbf{e}_p))^2 \tag{5.4}$$

where $\{\mathbf{r}_i\}_{i=1}^{N}$ are all data sample vectors.

5.3 Fully Constrained Least Squares-Based Endmember Finding Algorithms

When FCLS is implemented for LSMA it assumes that the signatures used to form a linear mixing model for data unmixing are known and must be provided *a priori*, in which case there is no issue in finding signatures. Unfortunately, when such knowledge is not given, finding these signatures becomes challenging. In the past, convex geometry-based techniques such as N-FINDR and its variants have been used for this purpose [39]. Despite that unsupervised FCLS (UFCLS) was previously developed for finding signatures for LSMA it was designed to find one signature at a time but no technique particularly designed for finding signatures all together based on unmixed error criterion (5.4), referred to as FCLS-based endmember finding algorithm (FCLS-EFA) as N-FINDR does for all endmembers.

Technically speaking, finding an optimal set of p signatures to yield minimal unmixed error requires an exhaustive search for all possible $\binom{N}{p} = \frac{N!}{p!(N-p)!}$ combinations. Practically speaking, this is nearly impossible to be done. This is the very same issue that also arises in implementing N-FINDR. In order to mitigate this dilemma two sequential versions of N-FINDR that were developed can be also used to develop their counterparts for FCLS-EFA for finding all endmembers.

5.3.1 Sequential FCLS-EFA

Since FCLS-EFA does not have any prior knowledge about endmembers which can be used as signatures for data unmixing, it must be carried out by an unsupervised means. Developing sequential FCLS (SQ FCLS) is to make an exhaustive search a sequential search so that FCLS can be implemented to find all the p endmembers iteratively in a very effective manner. Its idea is derived from the sequential N-FINDR which can be described as follows.

SQ FCLS-EFA

1. Initialization:

   Let p be the number of endmembers required to generate and $\left\{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \cdots, \mathbf{e}_p^{(0)}\right\}$ be a set of initial vectors randomly selected from the data.

   Let $\left\{\mathbf{r}_i\right\}_{i=1}^{N}$ be all data sample vectors. Set $i=1$ and go to step 3.

2. Outer Loop: (using index i as a counter to keep track data sample vector $\mathbf{r}_i$)

   Check $i=N$. If it is, the algorithm terminated. Otherwise, let $i \leftarrow i+1$ and continue.

3. Input the i^{th} data sample vector $\mathbf{r}_i$.

4. Inner Loop: (using j as a counter to keep track the j^{th} endmember $\mathbf{e}_j$)

   For $1 \le j \le p$, we recalculate UME($\mathbf{e}_1^{(i)}, \cdots, \mathbf{e}_{j-1}^{(i)}, \mathbf{r}_i, \mathbf{e}_{j+1}^{(i)}, \cdots, \mathbf{e}_p^{(i)}$) according to (5.4) which is the LSE unmixed by $\mathbf{e}_1^{(i)}, \cdots, \mathbf{e}_{j-1}^{(i)}, \mathbf{r}_i, \mathbf{e}_{j+1}^{(i)}, \cdots, \mathbf{e}_p^{(i)}$ for the data sample vector $\mathbf{r}_i$. If any of these p recalculated unmixed LSEs

UME($\mathbf{r}_i$, $\mathbf{e}_2^{(i)}$, $\mathbf{e}_3^{(i)}$, ⋯, $\mathbf{e}_p^{(i)}$), ⋯, UME($\mathbf{e}_1^{(i)}$, ⋯, $\mathbf{e}_{p-1}^{(i)}$, $\mathbf{r}_i$) is greater than UME($\mathbf{e}_1^{(i)}$, $\mathbf{e}_2^{(i)}$, ⋯, $\mathbf{e}_p^{(i)}$), go to step 5. Otherwise, go to step 2.

5. Replacement Rule:

   The endmember, which is absent in the minimal unmixed LSE among UME($\mathbf{r}_i$, $\mathbf{e}_2^{(i)}$, $\mathbf{e}_3^{(i)}$, ⋯, $\mathbf{e}_p^{(i)}$), ⋯, UME($\mathbf{e}_1^{(i)}$, ⋯, $\mathbf{e}_{p-1}^{(i)}$, $\mathbf{r}_i$), will be replaced by the i^{th} sample vector $\mathbf{r}_i$. Assume that such an endmember is now denoted by $\mathbf{e}_j^{(i+1)}$. A new set of endmembers is then produced by letting $\mathbf{e}_j^{(i+1)} = \mathbf{r}_i$ and $\mathbf{e}_i^{(i+1)} = \mathbf{e}_i^{(i)}$ for $i \neq j$ and go to step 2.

5.3.2 Successive FCLS-EFA

Another sequential version, successive N-FINDR which was also derived for N-FINDR can be also used to derive successive FCLS-EEA as follows. Its difference from SQ FCLS-EFA is that the endmembers found by SC FCLS-EFA in previous iterations are fixed and remain unchanged while the SQ FCLS-EFA found endmembers must be re-calculated over and over again. As a result, SC FCLS-EFA produces less optimal set of endmembers but requires less computing time than SQ FCLS-EFA does.

SC FCLS-EFA

1. Initialization:

   Let p be the number of endmembers required to generate and $\left\{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \cdots, \mathbf{e}_p^{(0)}\right\}$ be a set of initial vectors randomly selected from the data. Let $\left\{\mathbf{r}_i\right\}_{i=1}^{N}$ be all data sample vectors.

2. Outer Loop (using j as a counter to keep track the j^{th} endmember $\mathbf{e}_j$)

   For $1 \le j \le p$, find $\mathbf{e}_j^{(*)}$ which will replace $\mathbf{e}_j^{(0)}$.

3. Inner Loop (using index i as a counter to keep track data sample vector $\mathbf{r}_i$)

   For $1 \le i \le N$, calculate UME($\mathbf{e}_1^{(*)}$, $\cdots$, $\mathbf{e}_{j-1}^{(*)}$, $\mathbf{r}_i$, $\mathbf{e}_{j+1}^{(0)}$, $\cdots \mathbf{e}_p^{(0)}$) defined by (5.4) for all $\{\mathbf{r}_i\}_{i=1}^{N}$, while fixing other endmembers $\mathbf{e}_i^{(*)}$ with $i < j$ and $\mathbf{e}_i^{(0)}$ with $i > j$. Find $\mathbf{e}_j^{(*)} = \arg\left\{\min_{\mathbf{r}_i} \text{UME}(\mathbf{e}_1^{(*)}, \cdots, \mathbf{e}_{j-1}^{(*)}, \mathbf{r}_i, \mathbf{e}_{j+1}^{(0)}, \cdots, \mathbf{e}_p^{(0)})\right\}$.

4. Stopping Rule:

   If $j \le p$, then $j \leftarrow j+1$ and go to step 2. Otherwise, the final set of $\{\mathbf{e}_1^{(*)}, \mathbf{e}_2^{(*)}, \cdots, \mathbf{e}_p^{(*)}\}$ is the desired p endmembers.

5.4 Randomness Issues Addressed by Fully Constrained Least Squares-Based Endmember Finding Algorithms

The SQ FCLS-EFA and SC FCLS-EFA developed in Section 5.3 suffer from uncertainty caused by random initial endmembers. This problem can be alleviated by similar developments for endmember initialization-driven N-FINDR, iterative N-FINDR and random NFINDR.

5.4.1 Initialization Driven FCLS-EFA

A simple way to eliminate random initial conditions is to custom-design an unsupervised target detection algorithm which can find a specific set of desired targets used as initial condition for FCLS-EFA. Such an algorithm is called endmember initialization algorithm (EIA) which ought to have a property that its

generated targets should be as close as possible in the sense of spectral similarity to final desired endmembers. For example, these algorithms can be ATGP, UNCLS, UFCLS, VCA even including those algorithms designed from finding endmembers, SGA, N-FINDR, etc. By virtue of these EIA-generated targets as initial condition the randomness of FCLS-EFA can be completely eliminated. By replacing random initial conditions implemented by FCLS-EFA with EIA-generated initial conditions we can derive an initialization-driven FCLS-EFA (ID-FCLS-EFA) as follows.

ID FCLS-EFA

1. Initialization:

   Assume that p is the number of endmembers to be generated.

2. Implement an EIA to generate an initial condition of p endmembers $\left\{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \cdots, \mathbf{e}_p^{(0)}\right\}$.

3. Apply FCLS-EFA (either SQ FCLS-EFA or SC FCLS-EFA) to find a final endmember sets.

5.4.2 Iterative FCLS-EFA

SQ FCLS-EFA and SC FCLS-EFA produce inconsistent endmember results resulting from randomly selected data sample vectors as their initial conditions. Interestingly, if we feed back the final endmember results produced by these two algorithms each time and use them as new initial conditions to re-run these algorithm. Such iterative process should be able to eliminate inconsistency caused by the uncertainty due to randomness. The following iterative FCLS-EFA (IFCLS-EFA) is derived from a similar idea used to derive IN-FINDR.

IFCLS-EFA

1. Initialization:

   Let $\{\mathbf{r}_i\}_{i=1}^{N}$ be all data sample vectors and $E^{(0)} = \{\mathbf{e}_1^{(0)}, \mathbf{e}_2^{(0)}, \cdots, \mathbf{e}_p^{(0)}\}$ be a set of initial vectors randomly generated from the data.

2. Apply SQ FCLS-EFA or SC FCLS-EFA using $E^{(0)}$ as an initial set of endmembers to produce final endmember set $E^{(1)}$ and let $k = 1$. (Note that the parameter k is used as a counter to keep track how many initial conditions have been updated).

3. At k^{th} iteration, apply SQ FCLS-EFA or SC FCLS-EFA using $E^{(k)}$ as an initial set of endmembers to generate p endmembers denoted by $E^{(k+1)} = \{\mathbf{e}_1^{(k+1)}, \mathbf{e}_2^{(k+1)}, \cdots, \mathbf{e}_p^{(k+1)}\}$.

4. Stopping Rule:

   If $E^{(k+1)} \neq E^{(k)}$, then $k \leftarrow k+1$ and go to step 4. Otherwise, the algorithm is terminated and the endmembers in $E^{(k)}$ is the desired set of endmembers.

5.4.3 Random FCLS

In IFCLS-EFA a feedback loop is implemented to keep feeding back endmembers generated by previous iterations to next iterations. The random FCLS-EFA (RFCLS-EFA) presented below takes a completely opposite approach by running FCLS-EFA as a random algorithm which considers each of FCLS-EFA using one set of randomly selected data sample vectors as initial condition as a single realization. In this case, we run FCLS-EFA as many times as we could where each time it uses a new set of random initial endmembers to produce a new realization. If an endmember is

significant, it should appear as many realizations as it can. Using this fact RFCLS-EFA should be able to converge to a set which contains final desired endmembers. Its detailed implementation is described in the following.

RFCLS-EFA

1. Initialization:

   Assume that p is the number of signatures to be generated. Let $n=1$ denote a counter to dictate number of runs required to implement FCLS-EFA (either SQ FCLS-EFA or SC FCLS-EFA).

2. Apply FCLS-EFA to generate p random signatures, denoted by $E^{(n)}=\left\{\mathbf{e}_1^{(n)}, \mathbf{e}_2^{(n)}, \cdots, \mathbf{e}_p^{(n)}\right\}$.

3. Find the unmixed error resulting from $E^{(n)}$, UME($\mathbf{e}_1^{(n)}, \cdots, \mathbf{e}_j^{(n)}, \cdots, \mathbf{e}_p^{(n)}$) specified by (5.4).

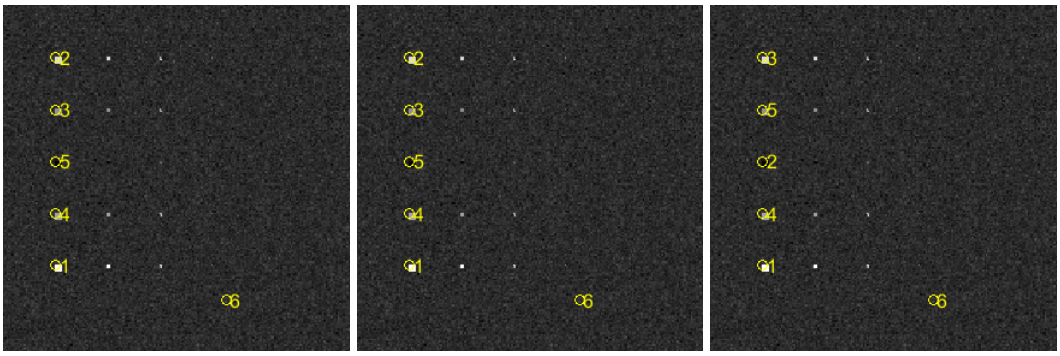
4. Find the variance of UME($\mathbf{e}_1^{(n)}, \cdots, \mathbf{e}_j^{(n)}, \cdots, \mathbf{e}_p^{(n)}$) for all realizations produced by $n\geq 1$, $\text{var}_{n\geq 1}\{\text{UME}(\mathbf{e}_1^{(n)}, \cdots, \mathbf{e}_j^{(n)}, \cdots, \mathbf{e}_p^{(n)})\}$.

5. If $\text{var}_{n\geq 1}\{\text{UME}(\mathbf{e}_1^{(n)}, \cdots, \mathbf{e}_j^{(n)}, \cdots, \mathbf{e}_p^{(n)})\}>\sigma^2$, let $n\leftarrow n+1$ and go to step 2. Otherwise the algorithm is terminated.

5.5 Experiments

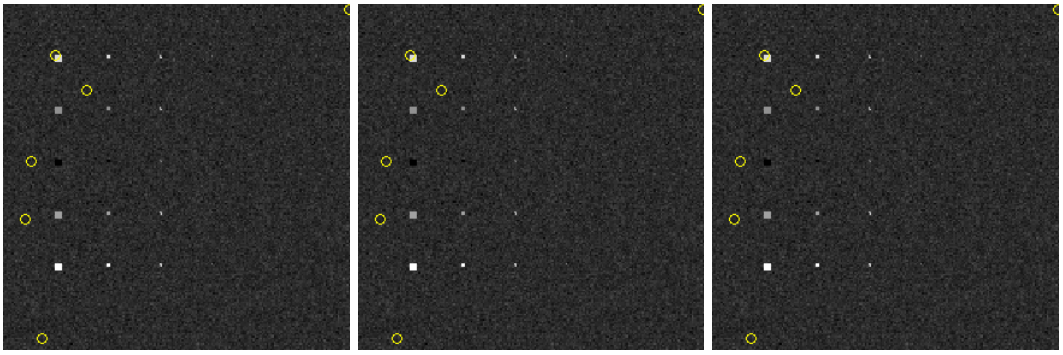
5.5.1 Synthetic Image Experiments

The same set of synthetic images, including TI and TE, are used for FCLS-EFA. The detailed introduction of TI and TE can be found in Section 3.5.1.

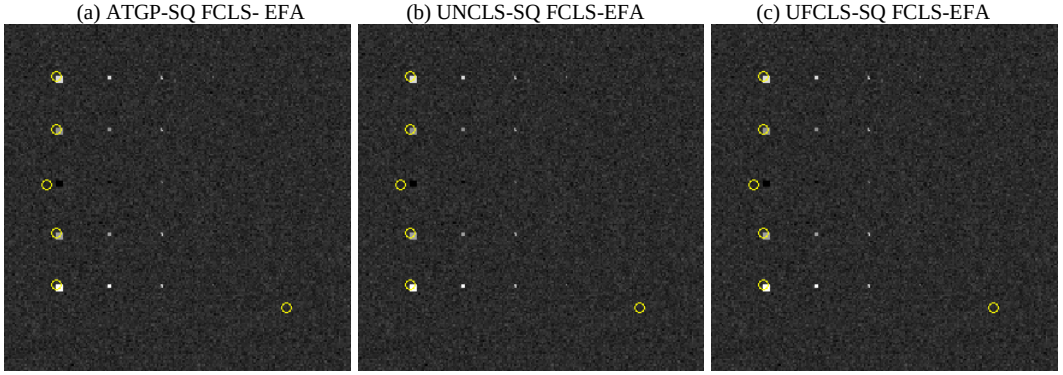


(a) ATGP (b) UNCLS (c) UFCLS

Figure 5.1 6 endmembers found for TI by ATGP, UNCLS and UFCLS

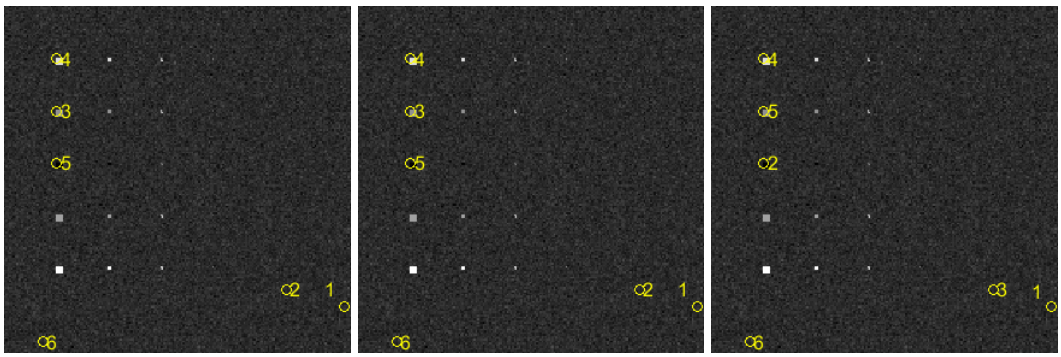


(a) ATGP-SQ FCLS- EFA (b) UNCLS-SQ FCLS-EFA (c) UFCLS-SQ FCLS-EFA

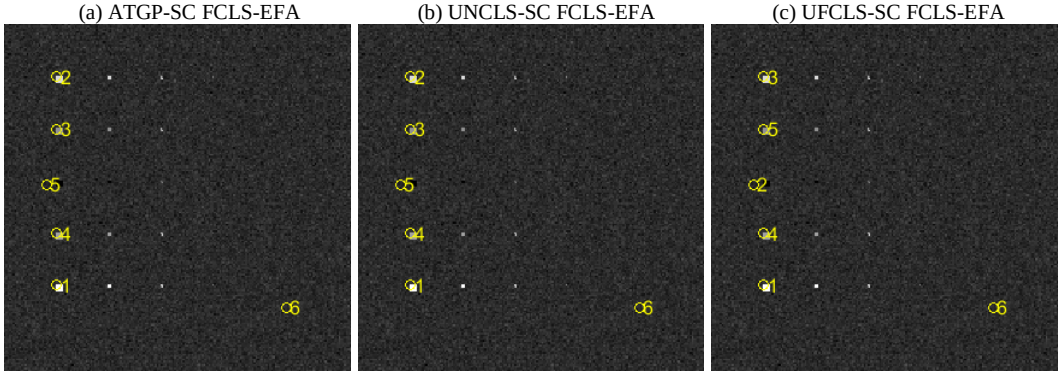


(d) ATGP-SQ N-FINDR (e) UNCLS-SQ N-FINDR (f) UFCLS-SQ N-FINDR

Figure 5.2 6 endmembers found for TI by ATGP-SQ FCLS-EFA, UNCLS-SQ FCLS-EFA, UFCLS-SQ FCLS-EFA, ATGP-SQ N-FINDR, UNCLS-SQ N-FINDR and UFCLS-SQ N-FINDR

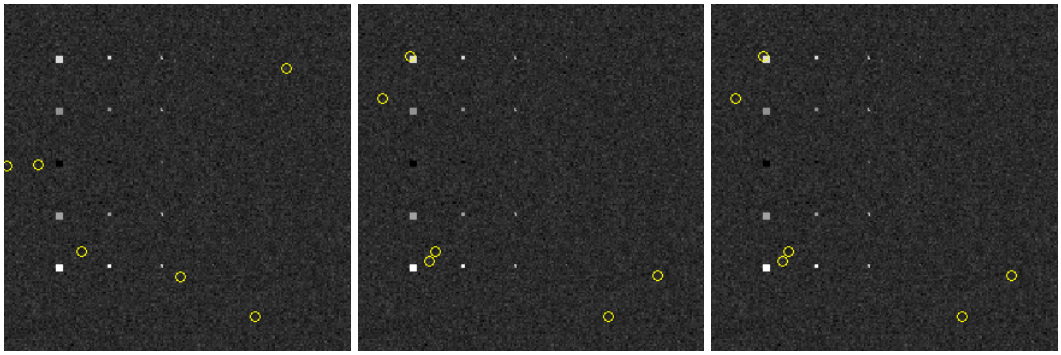


(a) ATGP-SC FCLS-EFA (b) UNCLS-SC FCLS-EFA (c) UFCLS-SC FCLS-EFA



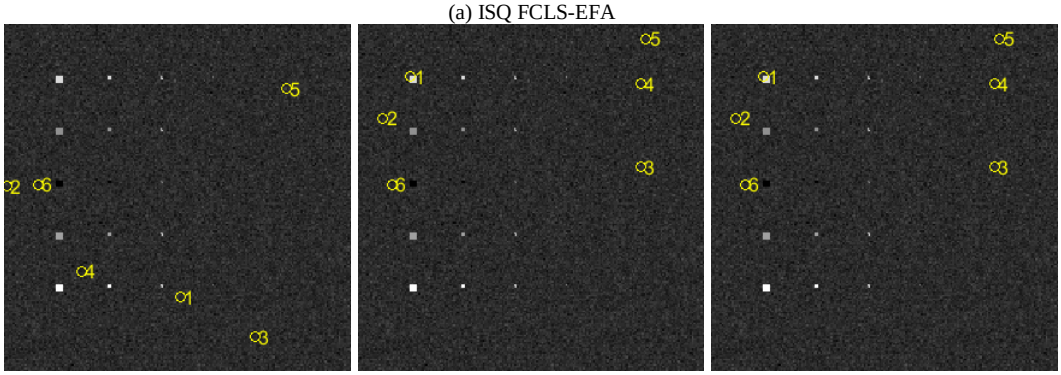
(d) ATGP-SC N-FINDR (e) UNCLS-SC N-FINDR (f) UFCLS-SC N-FINDR

Figure 5.3 6 endmembers found for TI by ATGP-SC FCLS-EFA, UNCLS-SC FCLS-EFA, UFCLS-SC FCLS-EFA, ATGP-SC N-FINDR, UNCLS-SC N-FINDR and UFCLS-SC N-FINDR



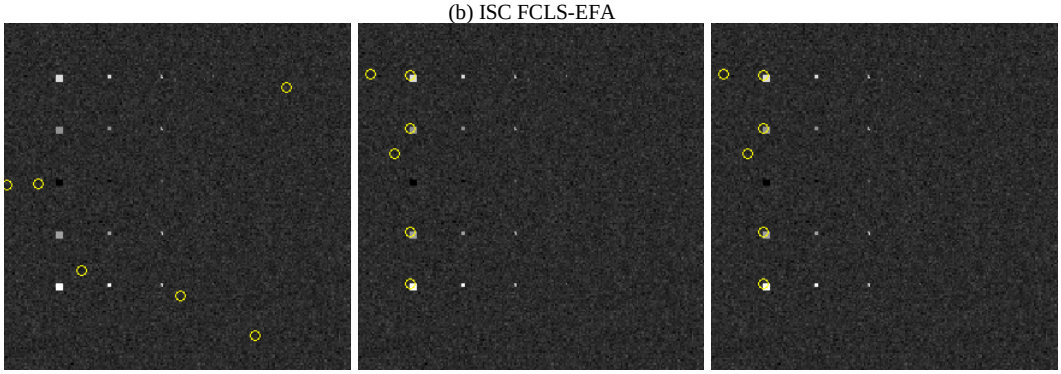
k= 0 *k*= 1 *k*=2

(a) ISQ FCLS-EFA



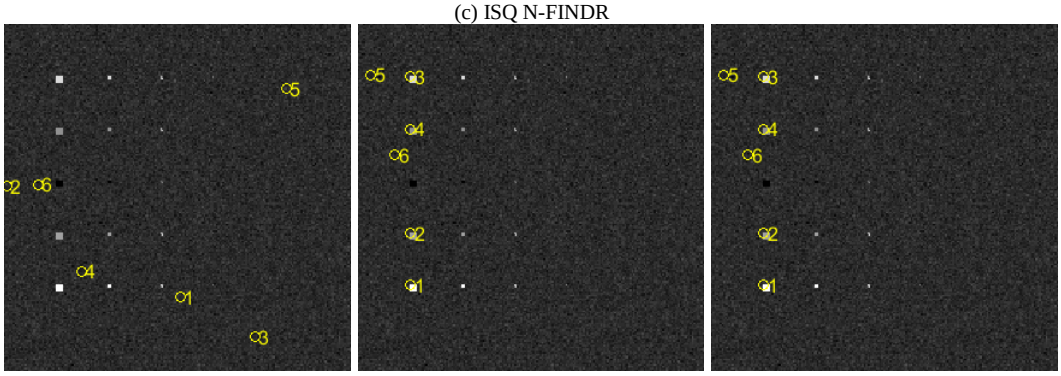
k= 0 *k*= 1 *k*=2

(b) ISC FCLS-EFA



k= 0 *k*= 1 *k*=2

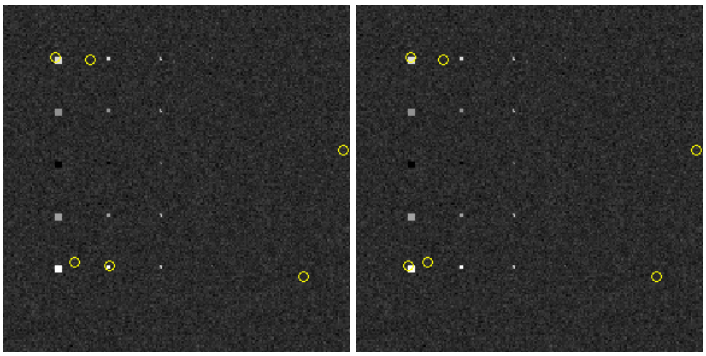
(c) ISQ N-FINDR



k= 0 *k*= 1 *k*=2

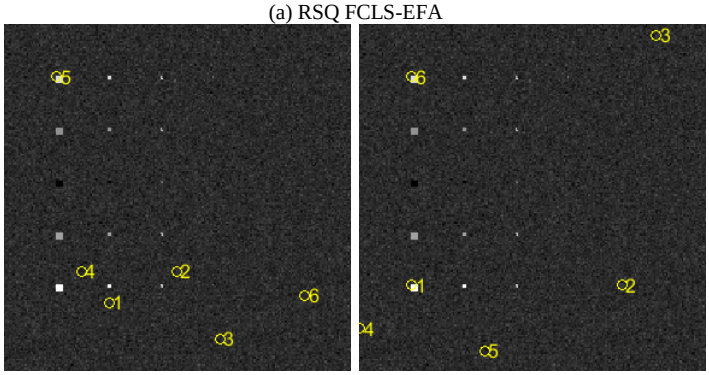
(d) ISC N-FINDR

Figure 5.4 6 endmembers found for TI by ISQ FCLS-EFA, ISC FCLS-EFA, ISQ N-FINDR and ISC N-FINDR



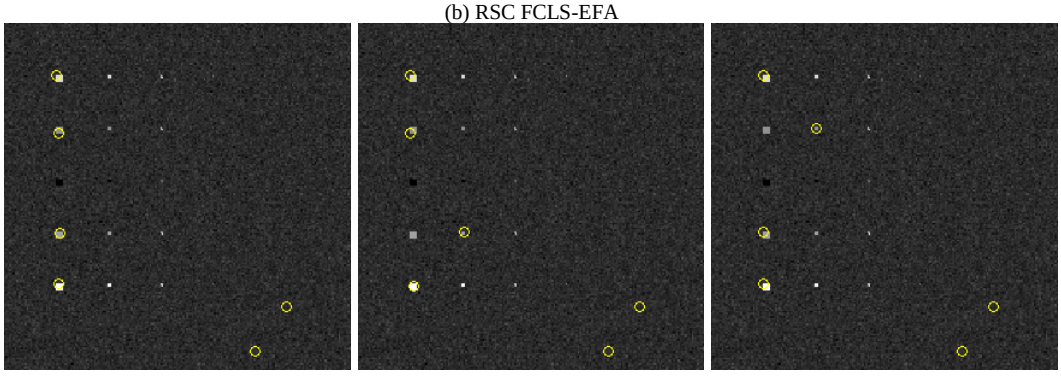
n = 1 *n* = 2

(a) RSQ FCLS-EFA



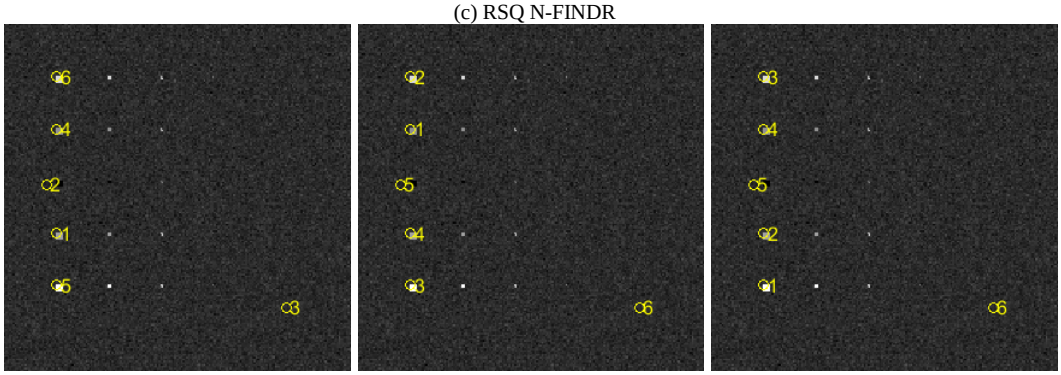
n = 1 *n* = 2

(b) RSC FCLS-EFA



n = 1 *n* = 2 *n* = 3

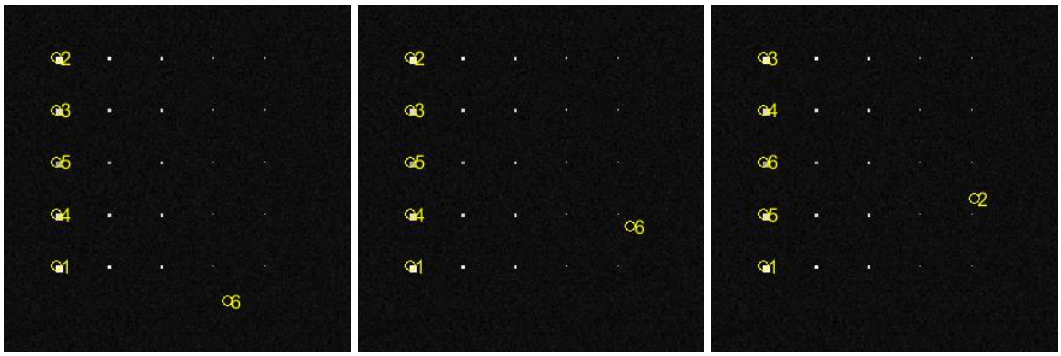
(c) RSQ N-FINDR



n = 1 *n* = 2 *n* = 3

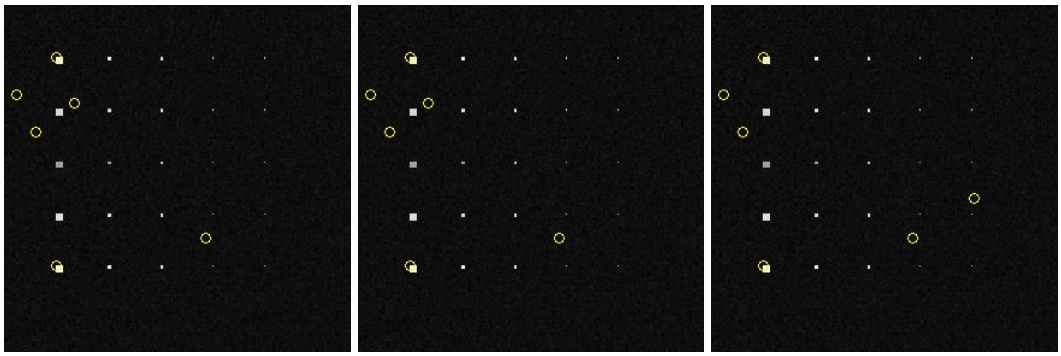
(d) RSC N-FINDR

Figure 5.5 6 endmembers found for TI by RSQ FCLS-EFA, RSC FCLS-EFA, RSQ N-FINDR and RSC N-FINDR

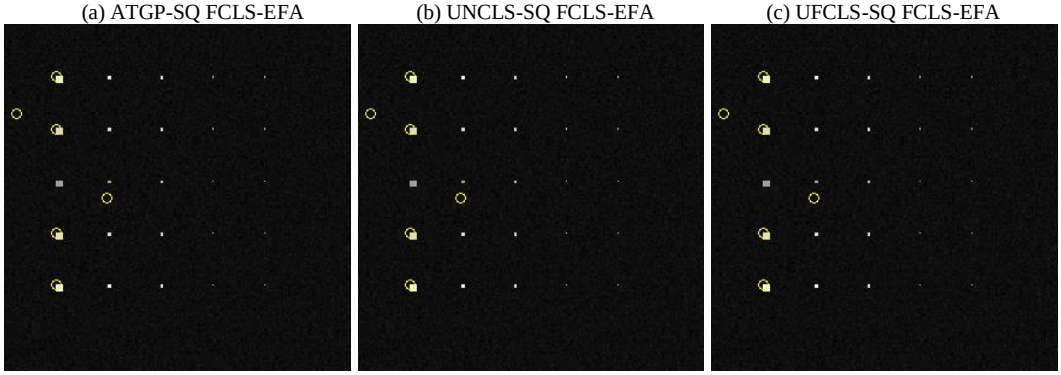


(a) ATGP (b) UNCLS (c) UFCLS

Figure 5.6 6 endmembers found for TE by ATGP, UNCLS and UFCLS

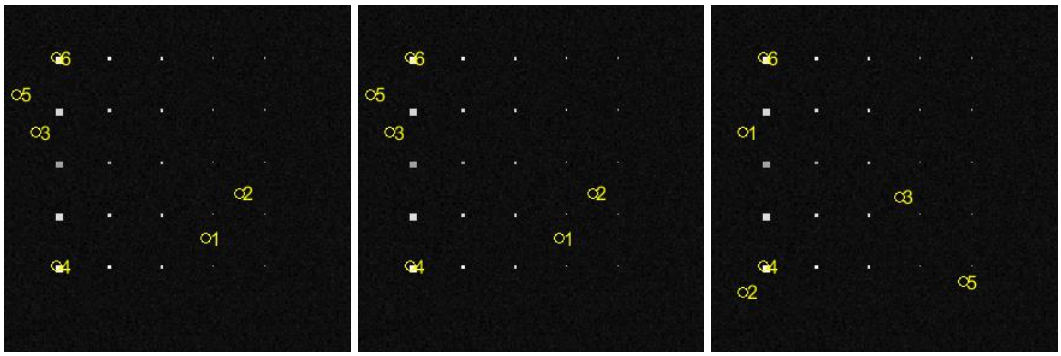


(a) ATGP-SQ FCLS-EFA (b) UNCLS-SQ FCLS-EFA (c) UFCLS-SQ FCLS-EFA

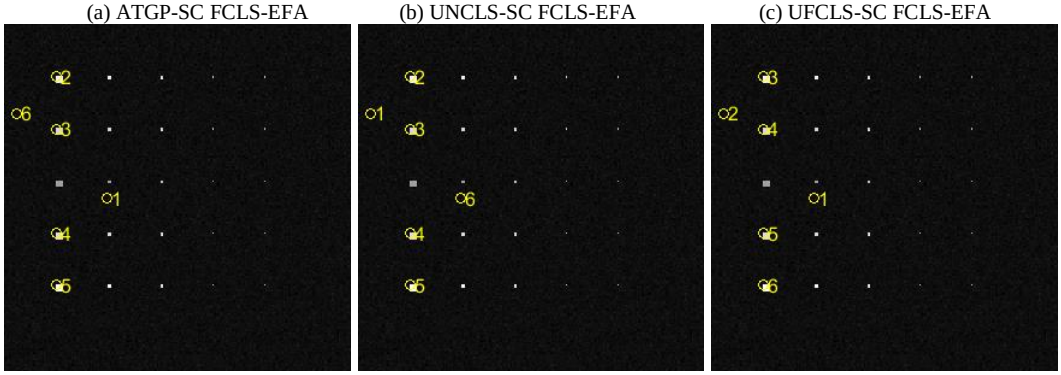


(d) ATGP-SQ N-FINDR (e) UNCLS-SQ N-FINDR (f) UFCLS-SQ N-FINDR

Figure 5.7 6 endmembers found for TE by ATGP-SQ FCLS-EFA, UNCLS-SQ FCLS-EFA, UFCLS-SQ FCLS-EFA, ATGP-SQ N-FINDR, UNCLS-SQ N-FINDR and UFCLS-SQ N-FINDR

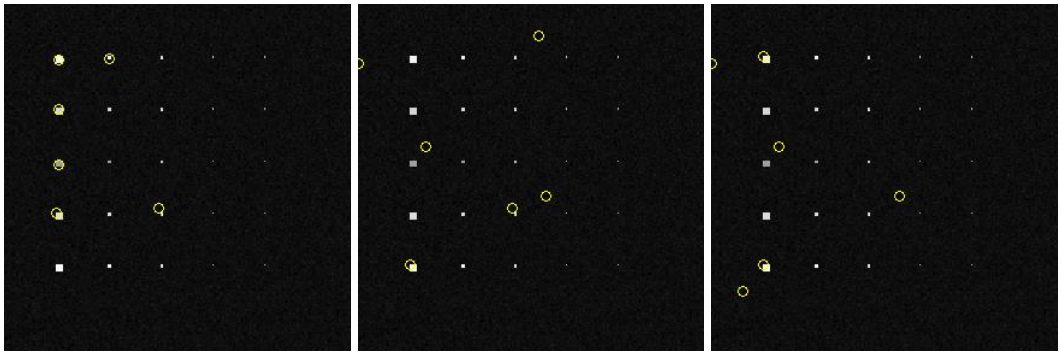


(a) ATGP-SC FCLS-EFA (b) UNCLS-SC FCLS-EFA (c) UFCLS-SC FCLS-EFA

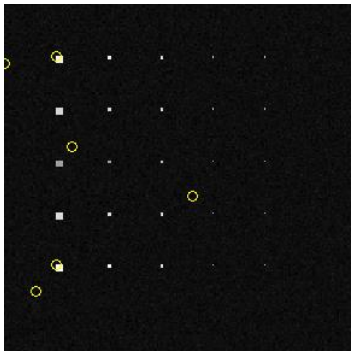


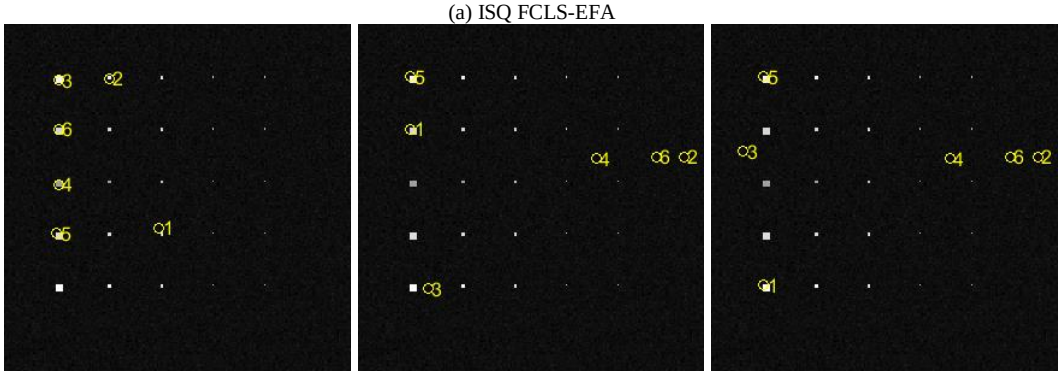
(d) ATGP-SC N-FINDR (e) UNCLS-SC N-FINDR (f) UFCLS-SC N-FINDR

Figure 5.8 6 endmembers found for TE for by ATGP-SC FCLS-EFA, UNCLS-SC FCLS-EFA, UFCLS-SC FCLS-EFA, ATGP-SC N-FINDR, UNCLS-SC N-FINDR and UFCLS-SC N-FINDR



k= 0 *k*= 1 *k*=2

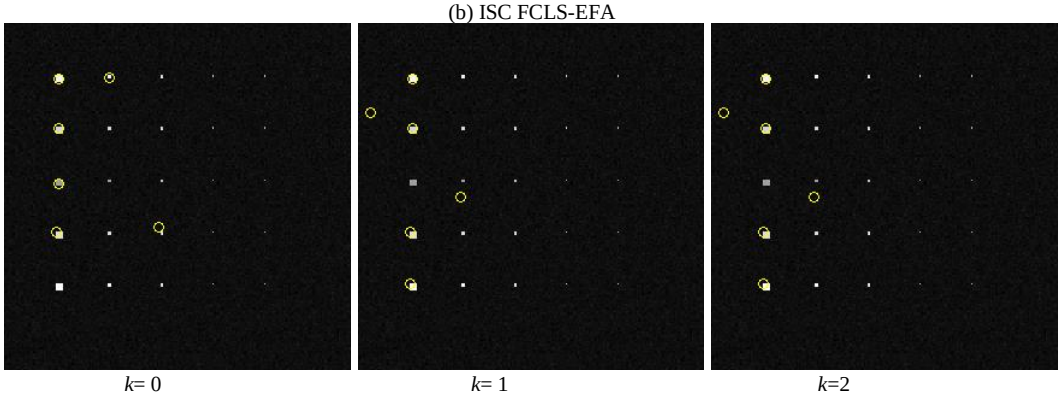




k= 0 *k*= 1 *k*=2

k=3

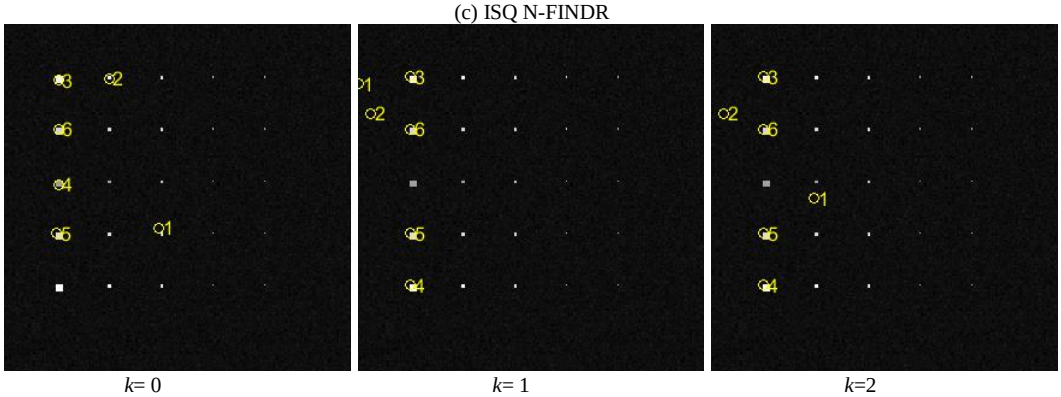
(a) ISQ FCLS-EFA



k= 0 *k*= 1 *k*=2

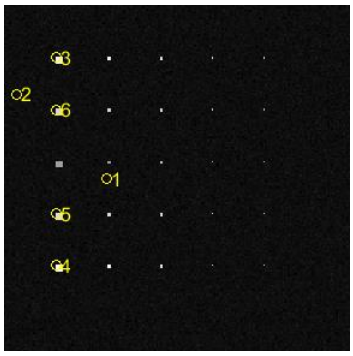
k=3

(b) ISC FCLS-EFA



k= 0 *k*= 1 *k*=2

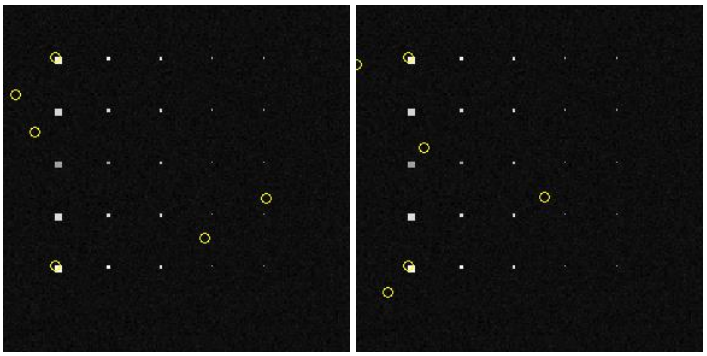
(c) ISQ N-FINDR



k=3

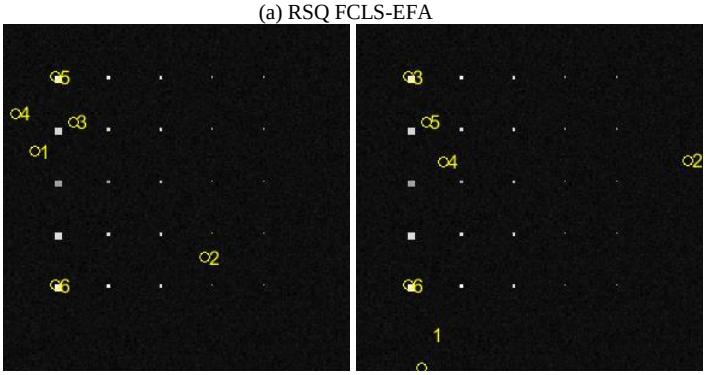
(d) ISC N-FINDR

Figure 5.9 6 endmembers found for TE by ISQ FCLS-EFA, ISC FCLS-EFA, ISQ N-FINDR and ISC N-FINDR



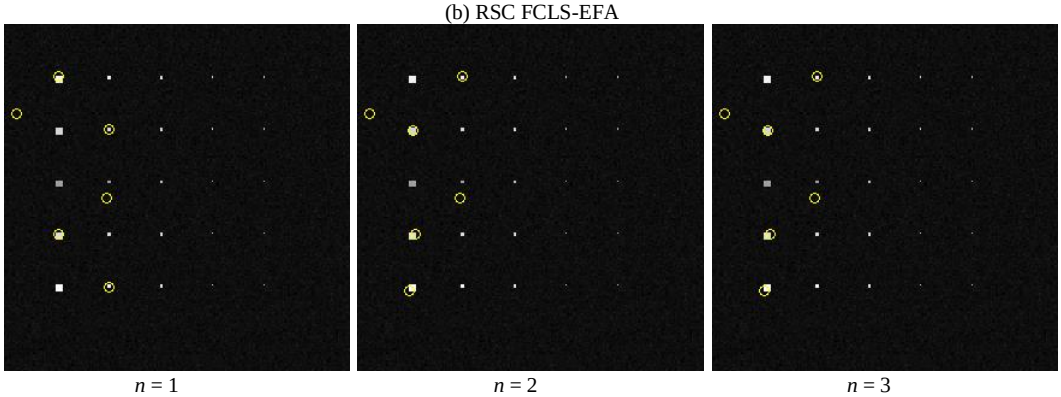
n = 1 *n* = 2

(a) RSQ FCLS-EFA



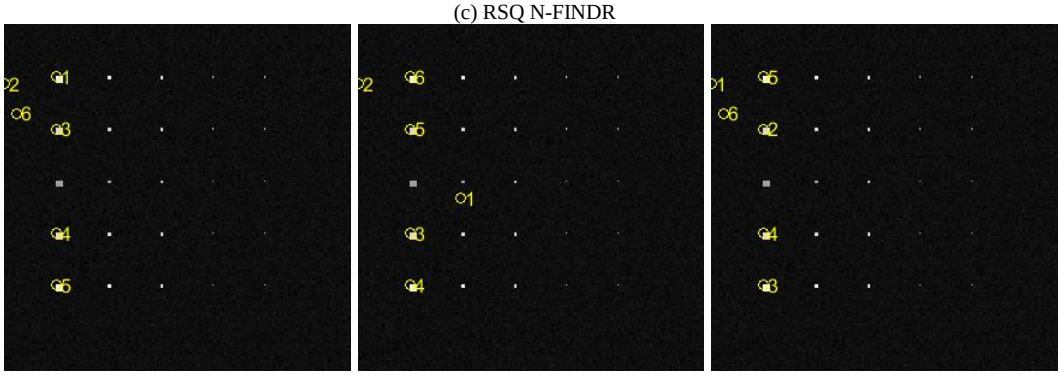
n = 1 *n* = 2

(b) RSC FCLS-EFA



n = 1 *n* = 2 *n* = 3

(c) RSQ N-FINDR



n = 1 *n* = 2 *n* = 3

(d) RSC N-FINDR

Figure 5.10 6 endmembers found for TE by RSQ FCLS-EFA, RSC FCLS-EFA, RSQ N-FINDR and RSC N-FINDR

Table 5.1. Averaged Unmixed Errors for TI produced by various versions of FCLS-EFA and N-FINDR

FCLS-EFA (N-FINDR)	Averaged Unmixing Errors
ATGP-SQ FCLS-EFA (ATGP-SQ N-FINDR)	81.63 (91.59)
UNCLS-SQ FCLS-EFA (UNCLS-SQ N-FINDR)	81.63 (91.59)
UFCLS-SQ FCLS-EFA (UFCLS-SQ N-FINDR)	81.63 (91.59)
ATGP-SC FCLS-EFA (ATGP-SC N-FINDR)	81.63 (91.59)
UNCLS-SC FCLS-EFA UNCLS-SC N-FINDR)	83.01 (91.59)
UFCLS-SC FCLS-EFA (UFCLS-SC N-FINDR)	83.01 (91.59)
ISQ FCLS-EFA (ISQ N-FINDR)	80.95 (91.59)
ISC FCLS-EFA (ISC N-FINDR)	81.07 (91.59)
ATGP	88.57
UNCLS	88.57
UFCLS	88.57
RSQ FCLS-EFA (*n* = 1) (RSQ N-FINDR *n* = 3)	80.85 (89.93)
RSC FCLS-EFA (*n* = 1) (RSC N-FINDR *n* = 1)	81.17 (91.59)

Table 5.2. Averaged Unmixed Errors for TE produced by various versions of FCLS-EFA and N-FINDR

FCLS-EFA (N-FINDR)	Averaged Unmixing Errors
ATGP-SQ FCLS-EFA (ATGP-SQ N-FINDR)	81.20 (91.27)
UNCLS-SQ FCLS-EFA (UNCLS-SQ N-FINDR)	81.20 (91.27)
UFCLS-SQ FCLS-EFA (UFCLS-SQ N-FINDR)	81.25 (91.27)
ATGP-SC FCLS-EFA (ATGP-SC N-FINDR)	81.32 (91.27)
UNCLS-SC FCLS-EFA UNCLS-SC N-FINDR)	81.32 (91.27)
UFCLS-SC FCLS-EFA (UFCLS-SC N-FINDR)	81.07 (91.27)
ISQ FCLS-EFA (ISQ N-FINDR)	81.34 (91.27)
ISC FCLS-EFA (ISC N-FINDR)	81.48 (91.27)
ATGP	107.98
UNCLS	112.46
UFCLS	101.07
RSQ FCLS-EFA (*n* = 2) (RSQ N-FINDR *n* = 3)	81.34 (91.27)
RSC FCLS-EFA (*n* = 1) (RSC N-FINDR *n* = 1)	80.94 (89.34)

For both TI and TE data, results for various versions of FCLS-EFA and N-FINDR are shown in Fig. 1-10. There are three methods to address random issue. In Fig. 1-2 and Fig. 6-7, three different endmember initialization algorithms, ATGP, UFCLS and UNCLS, are first used to find initial conditions and FCLS-EFA refined the result by minimizing averaged unmixing error. In Fig. 4 and Fig. 9, iterative FCLS-EFA and N-FINDR are used to refine the result by applying a feedback manner. It took 2 or 3 iterations to converge. In Fig. 5 and Fig. 10, random FCLS-EFA and N-FINDR use different initial conditions and the algorithm stops when the variance is less than a predefined threshold. From these results, FCLS-EFA cannot find all five panel pixels while N-FINDR can find four. In the last two rows in Table 5.1, averaged unmixing errors for random FCLS-EFA and N-FINDR are shown. $n = 1$ indicates the smallest UME from the first run among all the realizations. However, averaged unmixing errors for FCLS-EFA are less than those for N-FINDR as shown in Table 5.1 and Table 5.2, respectively for TI and TE. This illustrates that we are not looking for pure pixels to achieve better unmixing performance but most representative pixels in the image.

5.5.2 Real Image Experiments

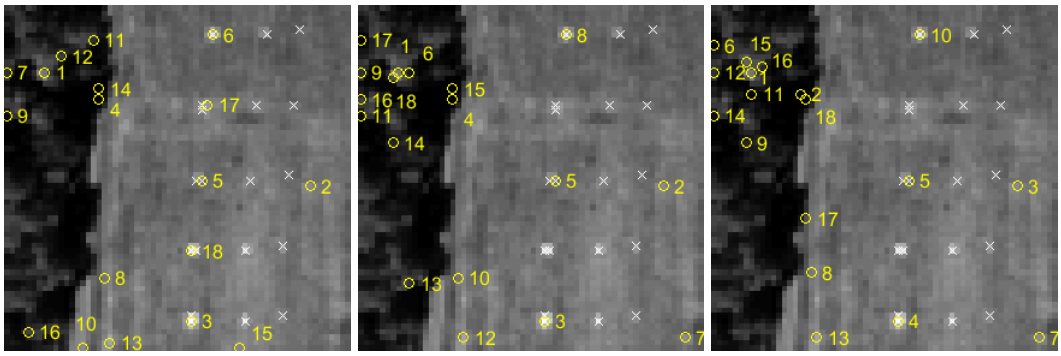
Real image, HYDICE, as introduced in Section 3.5.2, is used for testing effectiveness of algorithms developed in this chapter. According to [41, 42] the Virtual Dimensionality (VD) estimated for this scene for LSE was 18.

In order to address the use of random initial endmembers issue three approaches were conducted. The first approach is to use an EIA to generate a specific set of initial

endmembers to initialize SQ FCLS and SC FCLS. Fig. 12-13 shows that the results produced by SQ FCLS-EFA and SC FCLS-EFA along with SQ N-FINDR and SC N-FINDR with ATGP, UNCLS and UFCLS used as the EIA to generate a specific set of initial endmembers.

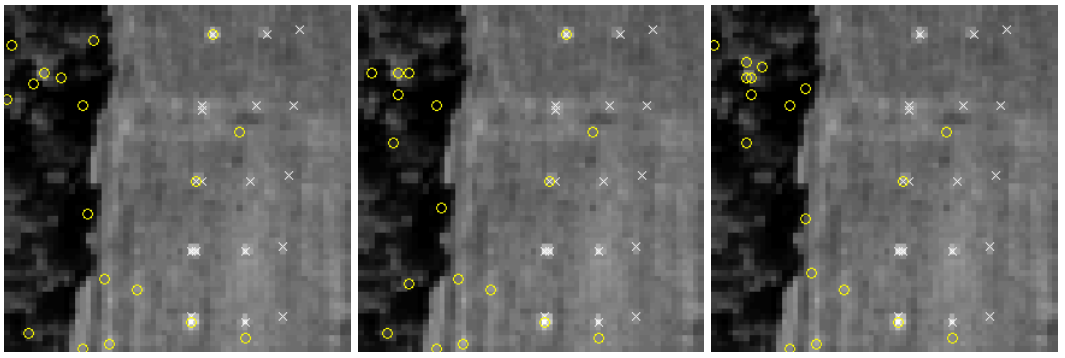
A second approach is to use a feedback approach which iteratively feed back the results produced by previous iterations as initial conditions for the next iteration. Figs. 14 also show results produced by ISQ FCLS and ISC FCLS along their counterparts ISQ N-FINDR and ISC N-FINDR respectively for comparison.

The third approach is random FCLS-EFA and N-FINDR. SC FCLS-EFA or SQ FCLS-EFA is applied multiple times by using different random initial conditions. The algorithm stops when the variance is less than a predefined threshold. In Fig. 15, results from different initial conditions are very similar. FCLS-EFA can only find 2 or 3 panel pixels while N-FINDR can find four. However, the averaged unmixing errors from FCLS-EFA are less than those from N-FINDR as shown in Table 5.3. As shown in Table 5.3, RSQ FCLS-EFA finds the third run which yields minimum unmixing error among three runs while the third run of RSQ N-FINDR finds minimum unmixing error 61.77. Similarly, the results for RSC FCLS-EFA and RSC N-FINDR are tabulated in the row of Table 5.3. The similar conclusion can be made with TI and TE data.

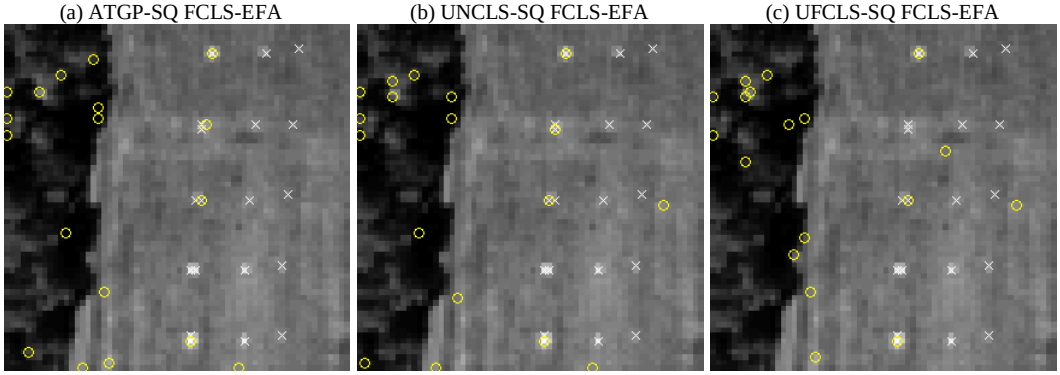


(a) ATGP (b) UNCLS (c) UFCLS

Figure 5.11 18 endmembers found for HYDICE by ATGP, UNCLS and UFCLS

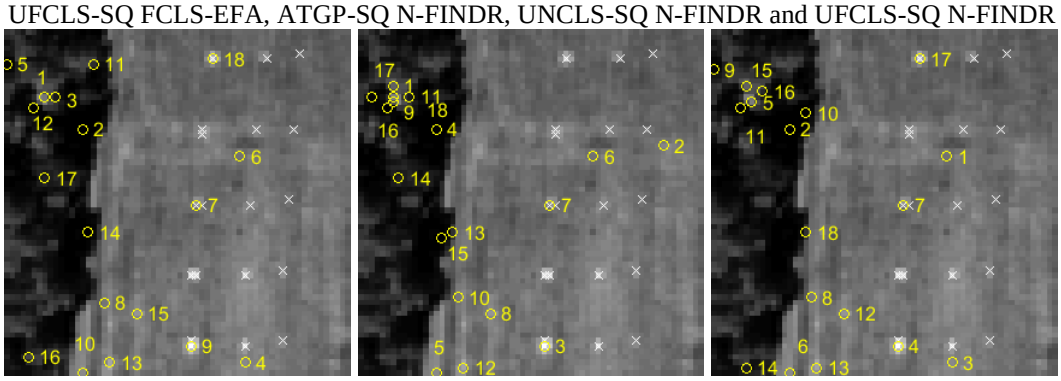


(a) ATGP-SQ FCLS-EFA (b) UNCLS-SQ FCLS-EFA (c) UFCLS-SQ FCLS-EFA

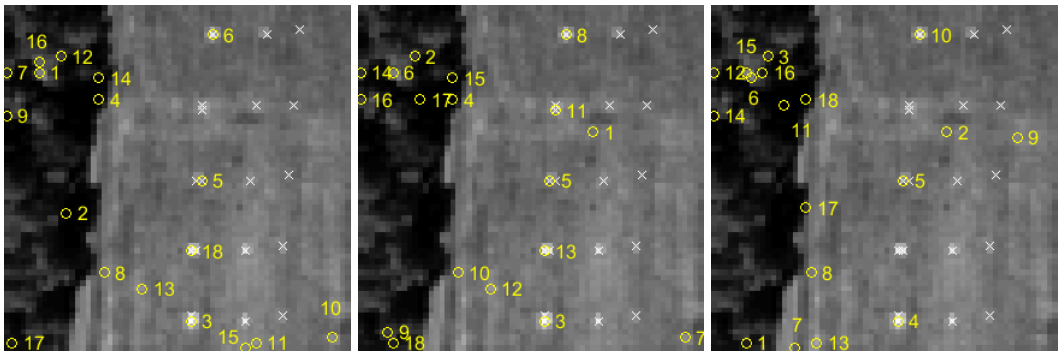


(d) ATGP-SQ N-FINDR (e) UNCLS-SQ N-FINDR (f) UFCLS-SQ N-FINDR

Figure 5.12 18 endmembers found for HYDICE by ATGP-SQ FCLS-EFA, UNCLS-SQ FCLS-EFA, UFCLS-SQ FCLS-EFA, ATGP-SQ N-FINDR, UNCLS-SQ N-FINDR and UFCLS-SQ N-FINDR

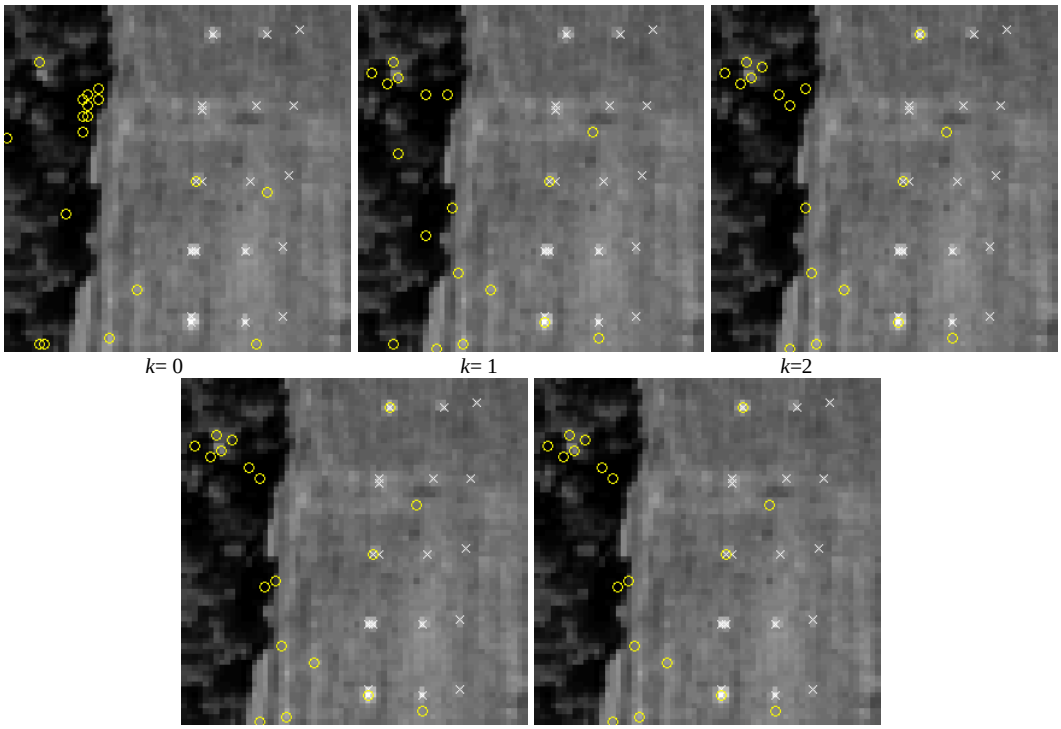


(a) ATGP-SC FCLS-EFA (b) UNCLS-SC FCLS-EFA (c) UFCLS-SC FCLS-EFA



(d) ATGP-SC N-FINDR (e) UNCLS-SC N-FINDR (f) UFCLS-SC N-FINDR

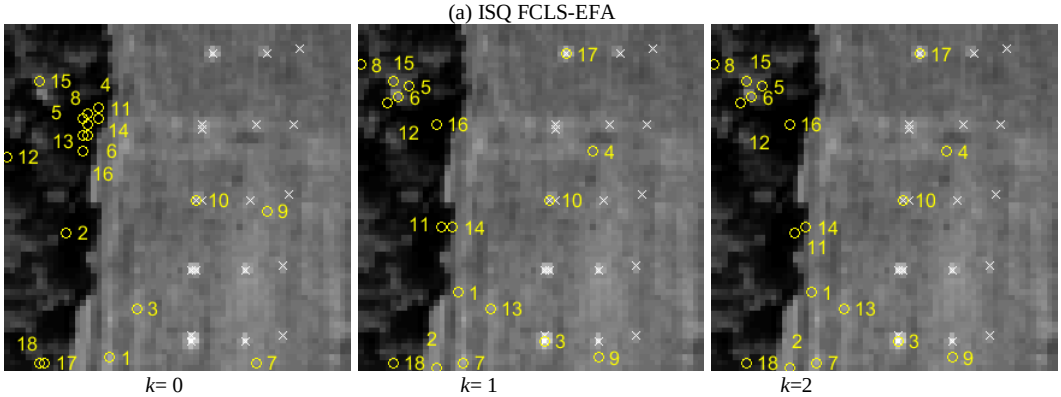
Figure 5.13 18 endmembers found for HYDICE by ATGP-SC FCLS-EFA, UNCLS-SC FCLS EFA and UFCLS-SC FCLS-EFA, ATGP-SC N-FINDR, UNCLS-SC N-FINDR and UFCLS-SC N-FINDR



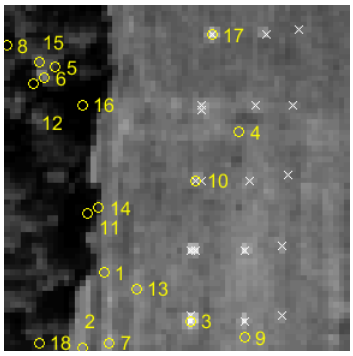
k= 0 *k*= 1 *k*=2

k=3 *k* =4

(a) ISQ FCLS-EFA

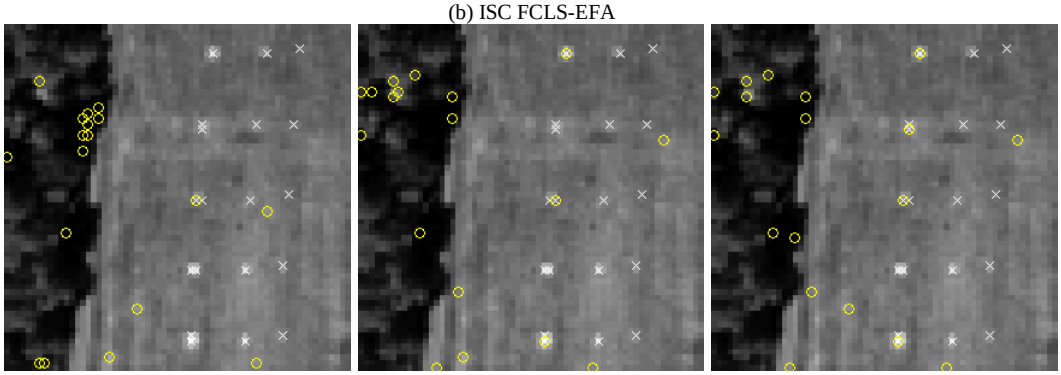


k= 0 *k*= 1 *k*=2

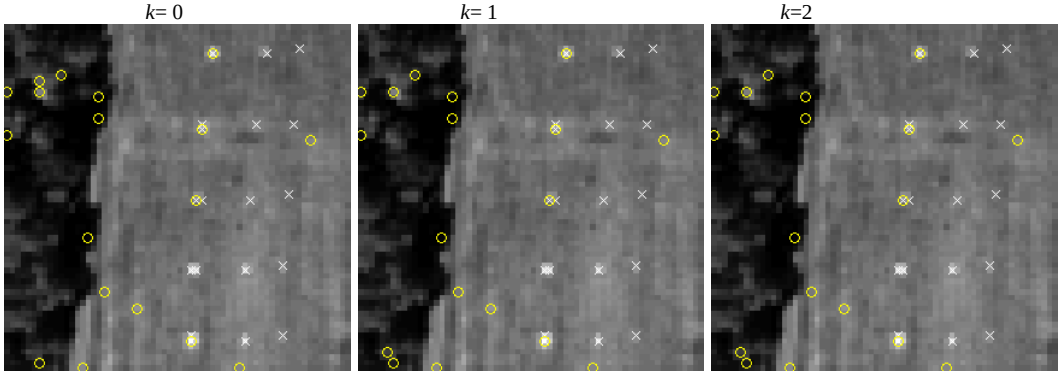


k=3

(b) ISC FCLS-EFA

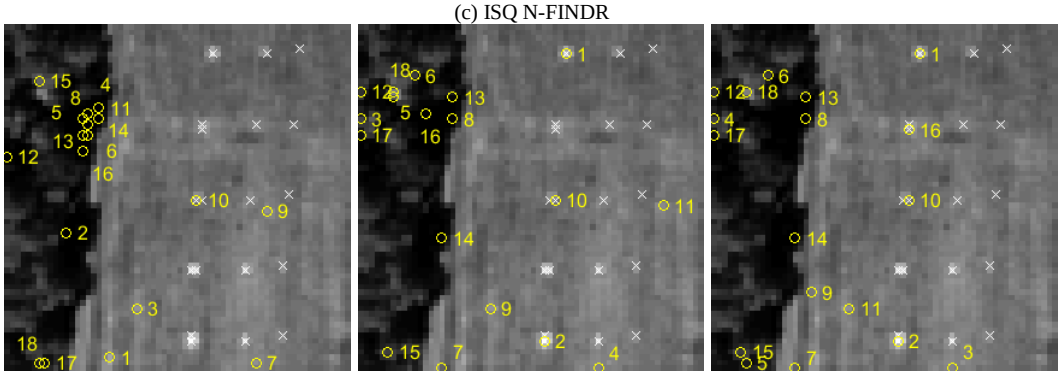


k= 0 *k*= 1 *k*=2

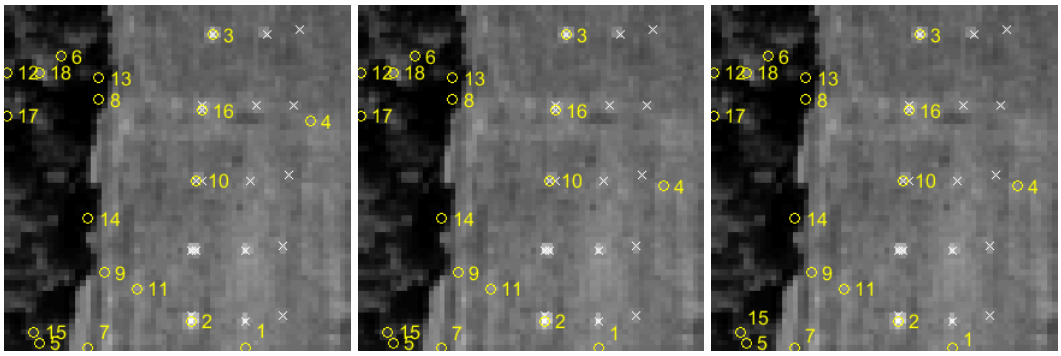


k=3 *k* =4 *k* =5

(c) ISQ N-FINDR



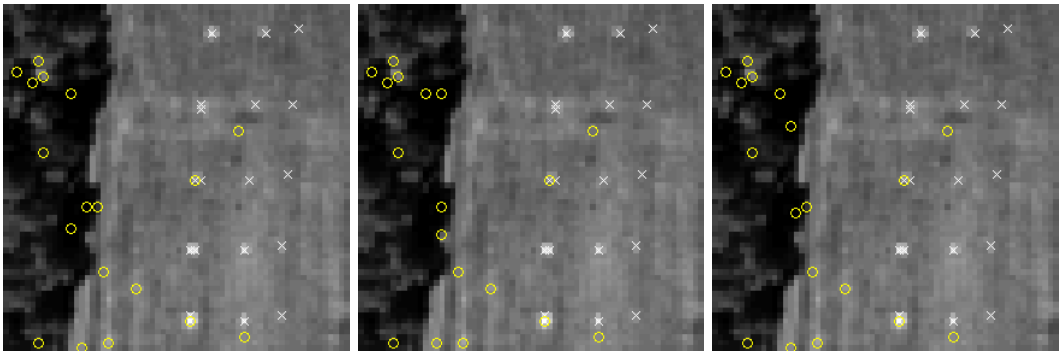
k= 0 *k*= 1 *k*=2



k=3 *k* =4 *k* =5

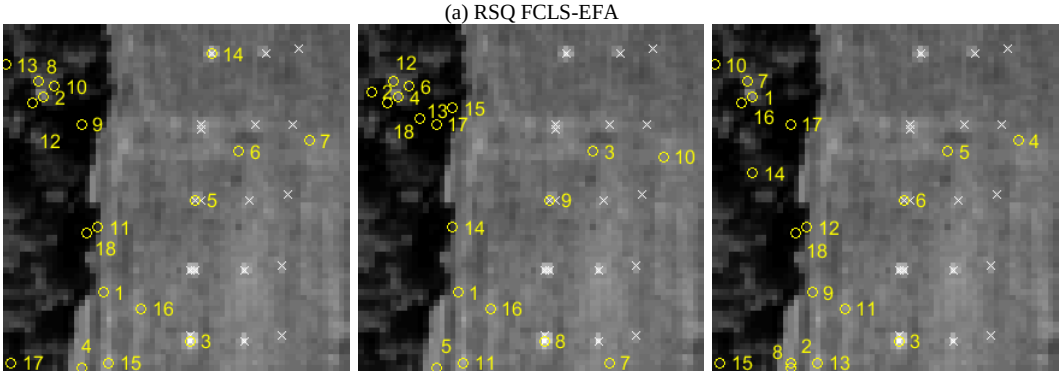
(d) ISC N-FINDR

Figure 5.14 18 endmembers found for HYDICE by ISQ FCLS-EFA, ISC FCLS-EFA, ISQ N-FINDR and ISC N-FINDR



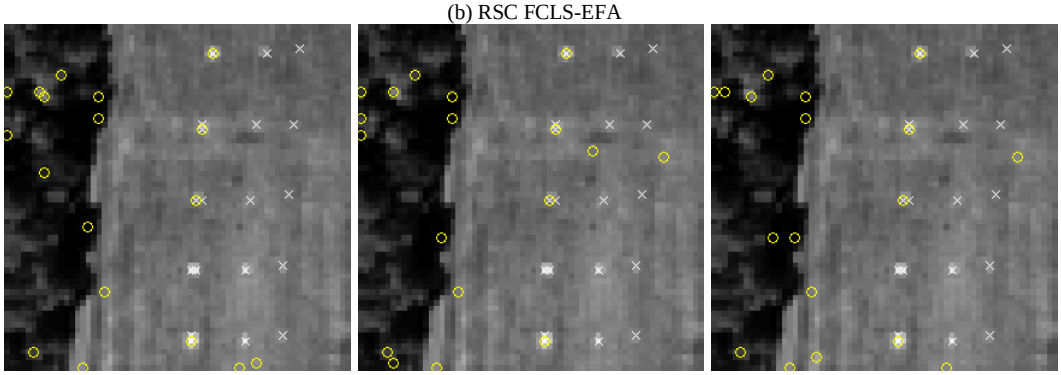
n = 1 *n* = 2 *n* = 3

(a) RSQ FCLS-EFA

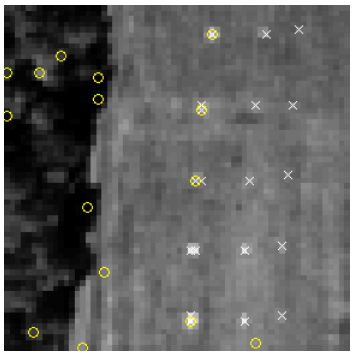


n = 1 *n* = 2 *n* = 3

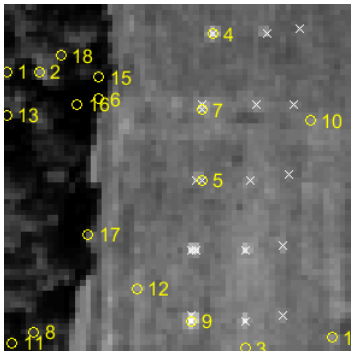
(b) RSC FCLS-EFA



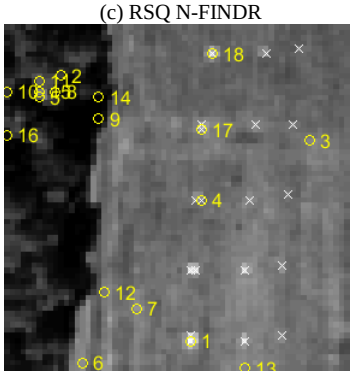
n = 1 *n* = 2 *n* = 3



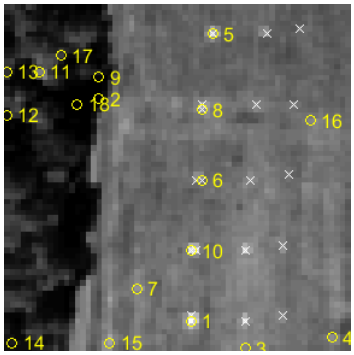
Intersection (15 endmembers)
(c) RSQ N-FINDR



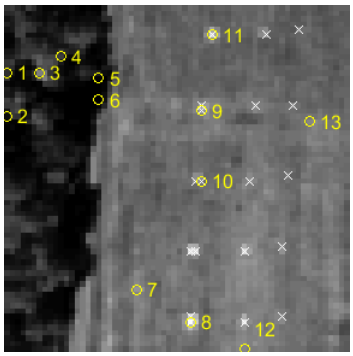
n = 1



n = 2



n = 3



Intersection (12 endmembers)
(d) RSC N-FINDR

Figure 5.15 18 endmembers found for HYDICE by RSQ FCLS-EFA, RSC FCLS-EFA, RSQ N-FINDR and RSC N-FINDR

Table 5.3 Averaged Unmixed Errors for HYDICE produced by various versions of FCLS-EFA and N-FINDR

FCLS-EFA (N-FINDR)	Averaged Unmixed Errors
ATGP-SQ FCLS-EFA (ATGP-SQ N-FINDR)	49.68 (62.67)
UNCLS-SQ FCLS-EFA (UNCLS-SQ N-FINDR)	50.74 (63.96)
UFCLS-SQ FCLS-EFA (UFCLS-SQ N-FINDR)	50.59 (60.90)
ATGP-SC FCLS-EFA (ATGP-SC N-FINDR)	49.98 (67.84)
UNCLS-SC FCLS-EFA UNCLS-SC N-FUNDR)	52.13 (67.43)
UFCLS-SC FCLS-EFA (UFCLS-SC N-FUNDR)	51.72 (65.23)
ISQ FCLS-EFA (ISQ N-FINDR)	51.81 (77.37)
ISC FCLS-EFA (ISC N-FINDR)	51.68 (74.09)
ATGP	69.74
UNCLS	66.87
UFCLS	61.67
RSQ FCLS-EFA (n = 3) (RSQ N-FINDR n = 3)	52.19 (61.77)
RSC FCLS-EFA (n = 1) (RSC N-FINDR n = 2)	51.91 (70.05)

5.6 Conclusion

In this chapter, average unmixing error is used as an optimization criterion to find endmembers in hyperspectral images. FCLS-based endmember finding algorithms (FCLS-EFA) are developed. In order to solve randomness issue, initialization driven FCLS version, iterative version and random version of algorithms are further developed. Both synthetic images and real images are used to test the effectiveness of the algorithms and the results are compared with the counterparts of N-FINDR. As a result, the found endmember by FCLS EFA are not necessarily pure signatures but rather those signatures can be used to perform better data unmixing. The experimental results demonstrate this fact and provide evidence that using endmembers produced by N-FINDR are not necessarily the best signatures to be used for data unmixing.

Chapter 6 Conclusion

6.1 Summary

In hyperspectral data processing, existence of endmembers cannot be guaranteed and endmembers may appear as various forms of true endmembers because of a number of reasons including environmental, atmospheric and temporal factors. In Chapter 2, a literature survey and review of endmember variability is conducted. A number of existing methods reported in the literature dealing with endmember variability are also summarized and compared in terms of advantages and disadvantages. The main idea of previous methods is to use a predefined threshold to find endmember classes. However, the disadvantages of doing so are as follows. 1. How to determine an appropriate predefined threshold. 2. How to apply thresholds to all the endmember classes since each endmember class should require a different threshold.

In order to deal with this issue, Algorithm for Finding Endmember Classes (AFEC) that does not require any prior knowledge or assumptions is first developed to find endmember classes in Chapter 3. In particular, Endmember Variability Algorithm (EVA) is designed as an iterative process, where an unsupervised endmember finding algorithm such as ATGP, UNCLS or UFCLS, is used to find initial endmembers which will be used by a follow-up AFEC to find corresponding endmember classes. The set of endmember classes are refined by an iterative process implemented in EVA.

In Chapter 4, a new concept is introduced to deal with endmember variability, referred to as endmember variability ratio (EVA)-based approach which is derived

from Fisher's ratio widely used in Fisher linear discriminant analysis. By taking advantage of EVR, several EVR-based endmember algorithms (EVR-EFA) can be also designed and developed.

In Chapter 5, a new optimization criterion, average unmixing error is used to find endmembers in hyperspectral images. Fully Constrained Least squares-based endmember finding algorithms (FCLS-EFA) are developed. In order to solve randomness issue, initialization driven FCLS version, iterative version and random version of algorithms are further developed. Both synthetic image and real image are used to test the effectiveness of the algorithms and the results are compared with the counterparts of N-FINDR. As a result, the endmembers found by FCLS-EFA are not necessarily pure signatures but rather those signatures can be used to perform better data unmixing. The experimental results demonstrate this fact and provide evidence that using endmembers produced by N-FINDR are not necessarily the best signatures to be used for data unmixing.

6.2 Future work

A future research direction is to extend the methods developed in this dissertation to more generic machine learning methods, which could be applied to more applications such as MRI image, vital signal data processing and as well as fingerprint.

Beside direct applications to other domain, there are other methods that could be used for endmember variability. For example, compressive sensing has received considerable interest in recent years. Sparsity and incoherence are two important conditions in compressive sensing. Sparsity requires endmembers to be sufficiently

sparse while incoherence requiring endmembers to be unrelated. One possible future work is to make use of compressive sensing technique to solve endmember variability issue.

REFERENCES

[1] C.-I Chang, *Hyperspectral Data Processing: Algorithm Design and Analysis*, John Wiley & Sons, New Jersey, 2013.

[2] C.-I Chang, Q. Du (2004), "Estimation of the number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Trans. Geosci. Remote Sens*. 42, no. 3, pp. 608-619.

[3] C.-I Chang (2013b), "Finding endmembers in hyperspectral imagery," SPIE Newsroom. DOI: 10.1117/2.1201304.004798.

[4] E. P. Frans and R. A. Schowengerdt (1997) "*Spatial-spectral unmixing using the sensor PSF*," *Proc. SPIE* 3118, Imaging Spectrometry III, 241 DOI:10.1117/12.283830.

[5] F. Garcia-Haro, S. Sommer, and T. Kemper, "A new tool for variable multiple endmember spectral mixture analysis (VMESMA)," Int. J. Remote Sensing, vol. 26, no. 10, pp. 2135–2162, May 2005.

[6] C. A. Bateson, G. P. Asner, and C. A. Wessman (2000), "Endmember bundles: A new approach to incorporating endmember variability into spectral mixture analysis," *IEEE Trans. Geosci. Remote Sensing*, vol. 38, no. 2, pp. 1083–1093.

[7] Y. Li, C. Gao, S.-Y Chen, C.-I Chang (2014), "Endmember variability resolved by pixel purity index in hyperspectral imagery", Proceedings of SPIE Vol. 9124, 91240I

[8] B. Somers, M. Zortea, A. Plaza and G.P. Asner (2012), “Automated extraction of image-based endmember bundles for improved spectral unmixing,” *IEEE J. Selected Tpics in Applied Earth Observation and Remote Sensing.*, vol. 5, no. 2, pp. 396-408, April 2012.

[9] C. Song 2005, “Spectral mixture analysis for subpixel vegetation fractions in the urban environment: How to incorporate endmember variability?” *Remote Sensing Environ*., vol. 95, no. 2, pp. 248–263.

[10] B. Somers, G. P. Asner, L. Tits, and P. Coppin (2011), “Endmember variability in spectral mixture analysis: A review,” *Remote Sensing Environ.*, vol. 115, no. 7, pp. 1603–1616.

[11] H. Ren and C.-I Chang (2003), "Automatic spectral target recognition in hyperspectral imagery," IEEE Trans. on Aerospace and Electronic Systems, vol. 39, no. 4, pp. 1232-1249.

[12] C.-I Chang, C. C. Wu, W. Liu and Y. C. Ouyang, "A New Growing Method for Simplex-Based Endmember Extraction Algorithm," in IEEE Transactions on Geoscience and Remote Sensing, vol. 44, no. 10, pp. 2804-2819, Oct. 2006.

[13] M.E. Winter (1999), “N-finder: an algorithm for fast autonomous spectral endmember determination in hyperspectral data,” *Image Spectrometry V, Proc. SPIE 3753*, pp. 266-277.

[14] D. Heinz, C. I. Chang and M. L. G. Althouse, "Fully constrained least-squares based linear unmixing hyperspectral image classification," *Geoscience and Remote Sensing Symposium, 1999. IGARSS '99 Proceedings. IEEE 1999 International*, Hamburg, 1999, pp. 1401-1403 vol.2.

[15] J. Adams, D. Sabol, V. Kapos, R. Filho, D. Roberts, M. Smith, and A. Gillespie, "Classification of multispectral images based on fractions of endmembers: Application to land-cover change in the Brazilian Amazon," Remote Sensing Environ., vol. 52, no. 2, pp. 137–154, May 1995.

[16] P. Dennison, K. Halligan, and D. Roberts, "A comparison of error metrics and constraints for multiple endmember spectral mixture analysis and spectral angle mapper," Remote Sensing Environ., vol. 93, pp. 359–367, Nov. 2004.

[17] B. Gao, M. Montes, C. Davis, and A. Goetz, "Atmospheric correction algorithms for hyperspectral remote sensing data of land and ocean," Remote Sensing Environ, vol. 113, Suppl. 1, pp. S17–S24, Sept. 2009.

[18] D. 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 Sensing Environ., vol. 65, pp. 267–279, Sept. 1998.

[19] J.-Ph. Combe, S. Le Mouelic, C. Sotin, A. Gendrin, J. F. Mustard, L. Le Deit, P. Launeau, J.-P. Bibring, B. Gondet, Y. Langevin, and P. Pinet, "Analysis of OMEGA/ Mars Express data hyperspectral data using a multiple-endmember

linear spectral unmixing model (MELSUM): Methodology and first results," Planet. Space Sci., vol. 56, pp. 951–978, Dec. 2008.

[20] G. Asner, M. Bustamante, and A. Townsend, "Scale dependence of biophysical structure in deforested areas bordering the Tapajs national forest, Central Amazon," Remote Sensing Environ., vol. 87, pp. 507–520, Mar. 2003.

[21] J. Jin, B. Wang, and L. Zhang, "A novel approach based on Fisher discriminant null space for decomposition of mixed pixels in hyperspectral imagery," IEEE Geosci. Remote Sensing Lett., vol. 7, no. 4, pp. 699–703, Oct. 2010.

[22] P. Dennison and D. Roberts, "Multiple endmember spectral mixture analysis using endmember average (MESMA)," Remote Sensing Environ., vol. 87, nos. 2–3, pp. 123–135, Oct. 2003.

[23] C. Bishop, Pattern Recognition and Machine Learning. New York: SpringerVerlag, 2006.

[24] F. Mianji and Y. Zhang, "SVM-based unmixing-to-classification conversion for hyperspectral abundance quantification," IEEE Trans. Geosci. Remote Sensing, vol. 49, no. 11, pp. 4318–4327, Nov. 2011.

[25] F. Bovolo, L. Bruzzone, and L. Carlin, "A novel technique for subpixel image classification based on support vector machine," IEEE Trans. Image Processing, vol. 19, no. 11, pp. 2983–2999, Nov. 2010.

[26] A. Castrodad, Z. Xing, J. Greer, E. Bosch, L. Carin, and G. Sapiro, “Learning discriminative sparse representations for modeling, source separation, and mapping of hyperspectral imagery,” IEEE Trans. Geosci. Remote Sensing, vol. 49, no. 11, pp. 4263–4281, Nov. 2011.

[27] A. Castrodad, Z. Xing, J. Greer, E. Bosch, L. Carin, and G. Sapiro, “Learning discriminative sparse representations for modeling, source separation, and mapping of hyperspectral imagery,” IEEE Trans. Geosci. Remote Sensing, vol. 49, no. 11, pp. 4263–4281, Nov. 2011.

[28] K. Canham, A. Schlamm, A. Ziemann, B. Basener, and D. Messinger, “Spatially adaptive hyperspectral unmixing,” IEEE Trans. Geosci. Remote Sensing, vol. 49, no. 11, pp. 4248–4262, Nov. 2011.

[29] M. Goenaga, M. Torres-Madronero, M. Velez-Reyes, S. Van Bloem, and J. Chinea, “Unmixing analysis of a time series of Hyperion images over the Gunica dry forest in Puerto Rico,” IEEE J. Select. Topics Appl. Earth Observ., vol. 6, no. 2, pp. 329–338, Apr. 2013.

[30] S. Moussaoui, D. Brie, A. Mohammad-Djafari, and C. Carteret, “Separation of non-negative mixture of non-negative sources using a Bayesian approach and MCMC sampling,” IEEE Trans. Signal Processing, vol. 54, no. 11, pp. 4133–4145, Nov. 2006.

[31] N. Dobigeon, S. Moussaoui, M. Coulon, J.-Y. Tourneret, and A. O. Hero, “Joint Bayesian endmember extraction and linear unmixing for hyperspectral

imagery," IEEE Trans. Signal Processing, vol. 57, no. 11, pp. 4355–4368, Nov. 2009.

[32] D. Stein, "Application of the normal compositional model to the analysis of hyperspectral imagery," in Proc. IEEE Workshop Advances in Techniques for Analysis of Remotely Sensed Data, Oct. 2003, pp. 44–51.

[33] O. Eches, N. Dobigeon, C. Mailhes, and J. Y. Tourneret, "Bayesian estimation of linear mixtures using the normal compositional model. Application to hyperspectral imagery," IEEE Trans. Image Processing, vol. 19, no. 6, pp. 1403–1413, June 2010.

[34] A. Zare, P. Gader, D. Dranishnikov, and T. Glenn, "Spectral unmixing using the beta compositional model," in Proc. IEEE Workshop. Hyperspectral Image and Signal Processing: Evolution in Remote Sensing, Gainesville, FL, June 2013.

[35] P. Bosdogianni, M. Petrou, and J. Kittler, "Mixture models with higher order moments," IEEE Trans. Geosci. Remote Sensing, vol. 35, no. 2, pp. 341–353, Mar. 1997.

[36] D. C. Heinz, C.-I Chang (2000), "Unsupervised fully constrained squares linear spectral mixture analysis method for multispectral imagery, " Geoscience and Remote Sensing Symposium, Proceedings IGARSS 2000 IEEE 2000 International, vol.4, no., pp.1681,1683 vol.4.

[37] C.-I Chang, A. Plaza (2006), “A fast iterative algorithm for implementation of pixel purity index,” Geoscience and Remote Sensing Letters, IEEE , vol.3, no.1, pp.63,67.

[38] C. Gao, S.-Y. Chen and C.-I Chang, “Fisher's ratio-based criterion for finding endmembers in hyperspectral imagery,” *Proc. SPIE* 9124, Satellite Data Compression, Communications, and Processing X, 91240K (May 22, 2014)

[39] W. Xiong, C.-I Chang, C.-C. Wu, K. Kalpakis, H. M. Chen (2011), “Fast Algorithms to Implement N-FINDR for Hyperspectral Endmember Extraction, ” *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of* , vol.4, no.3, pp.545,564.

[40] Chein-I Chang and C. M. Brumbley, "A Kalman filtering approach to multispectral image classification and detection of changes in signature abundance," in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 1, pp. 257-268, Jan 1999.

[41] C.-I Chang, X. Jiao, Y. Du and M.-L. Chang, “Unsupervised hyperspectral target analysis,” *EURASIP Journal on Advanced in Signal Processing*, Volume 2010

[42] J.C. Harsanyi and C.-I Chang, "Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach," *IEEE Trans. Geosci. Remote Sensing*, vol. 32, no. 4, pp. 779-785, July, 1994.