On the Generation of Training Samples for Neural Network-Based Mixed Pixel Classification

Javier Plaza[a], Chein-I Chang[b], Antonio Plaza[a,b], Rosa Pérez[a] and Pablo Martínez[a]

[a]Computer Science Department, University of Extremadura
Avda. de la Universidad s/n,10.071 Caceres, Spain;

[b]Remote Sensing Signal and Image Processing Laboratory
Department of Computer Science and Electrical Engineering
University of Maryland Baltimore County, Baltimore, MD 21250

ABSTRACT

One of great challenges in neural network-based analysis of remotely sensed imagery is to find an adequate pool of training samples without prior knowledge for the network so that that these unsupervised training samples can describe the data. A judicious selection of training data can be tremendously difficult due to the presence of subpixel targets and mixed pixels, particularly, when no prior knowledge is available. Surprisingly, the above issues have been largely overlooked in the past, where most of the efforts have been focused on exploring network architecture parameters such as the arrangement and number of neurons in the different layers. Very little has been done in regard to the selection of a set of good training samples for networks in mixed pixel classification. This paper revisits neural network-based mixed pixel classification from an aspect of training sample generation and further demonstrates that the selection of training samples can be more important than the choice of a specific network architecture. Since the training samples must be obtained directly from the data to be processed in an unsupervised fashion, four types of pixels: pure pixel, mixed pixel, anomalous pixel and homogeneous pixel are used to demonstrate this concept. A pure pixel is a pixel whose spectral signature is completely represented by a single material substance as opposed to a mixed pixel whose spectral signature is made up of more than one material substance. A homogeneous pixel is defined as a pixel whose spectral signature remains nearly constant subject to small variations within its surroundings. Therefore, a homogeneous pixel can be considered as an opposite of an anomalous pixel whose signature is spectrally distinct from the signatures of its neighboring pixels. In this paper, various scenarios are designed for experiments to substantiate the impact of using these four types of pixels as training samples for mixed pixel classification.

Keywords: Anomalous pixel. Homogeneous pixel. Neural networks. Mixed pixel. Mixed pixel classification. Pure pixel. Training samples.

1. INTRODUCTION

Hyperspectral imaging spectrometers collect data from airborne or spaceborne platforms[1]. Each of the pixels collected by hyperspectral imagers contains the resultant mixed spectrum from the reflected surface radiation of subpixel constituent materials within the pixel. Mixed pixels exist for several reasons. First, if the spatial resolution of the sensor is not high enough to separate different pure signature materials at a macroscopic level (endmembers)[2], these can jointly occupy a single pixel, and the resulting spectral measurement will be a composite of the individual spectra. Second, mixed pixels can also result when distinct materials are combined into a homogeneous mixture. This circumstance also occurs independent of the spatial resolution of the sensor. Spectral mixture analysis (SMA) involves the separation of a pixel spectrum into its component endmember spectra and the estimation of the abundance value for each endmember in the pixel[3]. The use of a linear spectral mixture model assumes that the collected spectra are linearly mixed. The definition of a linear (macroscopic) mixture is that endmember substances are sitting side-by-side within the field of view of the imager. Although the linear model has several advantages including ease of implementation and flexibility in different applications, there are many naturally occurring situations where a nonlinear mixture model may better describe the resultant mixed spectra for certain endmember distributions. In particular, nonlinear mixtures occur in situations where endmember components are randomly distributed throughout the field of view of the instrument[4].

Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XI,
edited by Sylvia S. Shen, Paul E. Lewis, Proceedings of SPIE Vol. 5806
(SPIE, Bellingham, WA, 2005) · 0277-786X/05/$15 · doi: 10.1117/12.604114

In recent years, neural networks have demonstrated great potential as a method to decompose mixed pixels due to the inherent capacity of neural architectures to approximate nonlinear functions[5]. Although many types of neural networks exist, for decomposition of mixed pixels in terms of nonlinear relationships mostly feed-forward networks such as the multi-layer perceptron (MLP) have been used. It has been shown in the literature that MLP-based models, when appropriately trained, generally outperform linear and nonlinear models such as regression trees or fuzzy classifiers[6]. Despite some encouraging results, exploitation of neural networks in mixed pixel classification and, specifically, in pixel unmixing, remains difficult. Specifically, one of the great challenges in design of neural network-based SMA techniques is to find an adequate pool of training samples so that that these training samples can accurately describe the data[7,8]. There is a need for algorithms able to generate training samples without prior knowledge.

Two major issues must be taken into account in the development of algorithms for generation of training samples. First, pixels in classic image processing are considered to be macroscopically pure compared to pixels in hyperspectral imaging, which are generally mixed. Under such circumstance, the pixel-level information present in hyperspectral scenes provides useful and crucial information that must be taken into account in the selection of training samples. A second issue is that classes in hyperspectral analysis are usually made up of mixed pixels or subpixel targets, which can be rare. However, a desired pool of training samples must include the information provided by such pixels. Subsequently, a judicious selection of training data can be tremendously difficult. In order to address the important issue of what pixel information can be extracted from hyperspectral image pixels, four types of pixels are defined and considered in this study[9]: pure, mixed, anomalous and homogeneous pixel. A pure pixel (endmember) is a pixel whose spectral signature is completely represented by a single material substance as opposed to a mixed pixel whose spectral signature is made up of more than one material substance. While pure/mixed pixels are defined by spectral properties, the spatial arrangement of pixels in the scene defines two new types of pixels. Specifically, a homogeneous pixel is defined as a pixel whose spectral signature remains nearly constant subject to small variations within its surroundings. Accordingly, a homogeneous pixel can be considered as an opposite of an anomalous pixel, whose signature is spectrally distinct from the signatures of its neighboring pixels. It is interesting to note that a pure pixel can be a homogenous pixel or anomalous pixel, so can a mixed pixel. Similarly, a homogeneous pixel and anomalous pixel can be also a pure pixel or a mixed pixel.

With the above definitions in mind, this paper offers a new look at the problem of training sample generation for neural network-based abundance estimation in hyperspectral imagery, and provides a set of intelligent algorithms which generate training samples in unsupervised fashion. Different sets of training samples, obtained by different algorithms, are used to train a MLP neural network in order to investigate the impact of the training sample generation process on the final classification results. The paper is organized as follows. Section 2 describes several unsupervised training sample selection algorithms. Section 3 describes a MLP neural network for pixel unmixing. Section 4 conducts experimental results using a database of true mixtures with absolute ground-truth, known as Mustard's data set, for performance analysis and quantitative comparison of SMA techniques based on training samples. Section 5 concludes with remarks.

2. UNSUPERVISED TRANING SAMPLE GENERATION ALGORITHMS

This section develops four unsupervised algorithms for generation of training samples from pure, mixed, homogeneous and anomalous pixels. Pure pixels are generated using a well-known N-FINDR algorithm. A new morphological erosion-based algorithm is developed to find the most highly mixed pixels in the data set. In order to generate training samples of homogeneous pixels, we use the automated morphological endmember extraction (AMEE) algorithm. Finally, the RX algorithm is used to generate training samples from anomalous pixels.

2.1. N-FINDR algorithm for extraction of training samples from pure pixels

The N-FINDR algorithm developed by Winter[10] is used in this work to generate training samples from pure pixels in the data. The algorithm assumes that, in N spectral dimensions, the N-dimensional volume formed by a simplex with vertices specified by purest pixels is always larger than that formed by any other combination of pixels. In this work, we use a modified version of N-FINDR, which assumes that the number of existing endmembers in the data, p, is known in advance. This value is estimated by taking advantage of the concept of virtual dimensionality (VD), introduced by Chang[1]. Specifically, the method used to determine the VD in this paper is the one developed by Harsanyi-Farrand-Chang[11], referred to as HFC method. Once the number of endmembers, p, has been estimated, our implementation of N-

FINDR is initialized by a simplex whose vertices are formed by a random set of pixels $\{\boldsymbol{t}_1^{(0)}, \boldsymbol{t}_2^{(0)}, \cdots, \boldsymbol{t}_p^{(0)}\}$ selected from the data. The algorithm then proceeds by finding $V(\boldsymbol{t}_1^{(0)}, \boldsymbol{t}_2^{(0)}, \cdots, \boldsymbol{t}_p^{(0)})$, the volume of the simplex defined by vertices $\boldsymbol{t}_1^{(0)}, \boldsymbol{t}_2^{(0)}, \cdots, \boldsymbol{t}_p^{(0)}$, denoted by $S(\boldsymbol{t}_1^{(0)}, \boldsymbol{t}_2^{(0)}, \cdots, \boldsymbol{t}_p^{(0)})$. Then, for each sample vector $\boldsymbol{r}$, it recalculates the volumes of p simplices, $S(\boldsymbol{r}, \boldsymbol{t}_2^{(0)}, \cdots, \boldsymbol{t}_p^{(0)})$, $S(\boldsymbol{t}_1^{(0)}, \boldsymbol{r}, \cdots, \boldsymbol{t}_p^{(0)})$,..., $S(\boldsymbol{t}_1^{(0)}, \boldsymbol{t}_2^{(0)}, \cdots, \boldsymbol{r})$, each of which is formed by replacing one pixel $\boldsymbol{t}_{\mathrm{j}}^{(0)}$ with the sample vector $\boldsymbol{r}$. If none of these p recalculated volumes is greater than $S(\boldsymbol{t}_1^{(0)}, \boldsymbol{t}_2^{(0)}, \cdots, \boldsymbol{t}_p^{(0)})$, then no endmember in $\boldsymbol{t}_1^{(0)}, \boldsymbol{t}_2^{(0)}, \cdots, \boldsymbol{t}_p^{(0)}$ is replaced. Otherwise, the endmember which is absent in the largest volume among the p simplices above is replaced by $\boldsymbol{r}$. Let such an endmember be denoted by $\boldsymbol{t}_{\mathrm{j}}^{(1)}$. A new set of endmembers can be produced by letting $\boldsymbol{t}_{\mathrm{j}}^{(1)} = \boldsymbol{r}$ and $\boldsymbol{t}_{\mathrm{i}}^{(1)} = \boldsymbol{t}_{\mathrm{i}}^{(0)}$ for $\mathrm{i} \neq \mathrm{j}$. The same process is repeated again for other sample vectors until all pixel vectors in the original data set are exhausted. In the end, a set of pixels denoted as $\{\boldsymbol{t}_{\mathrm{i}}^{(\mathrm{N\text{-}FINDR})}\}_{\mathrm{i}=1}^{p}$ is obtained.

2.2. Morphological erosion-based algorithm for extraction of training samples from mixed pixels

Multidimensional morphological operations can be very useful for the interpretation of mixed pixels[12]. These operations take into account both the spatial and the spectral properties of the image data. In order to incorporate the spatial information, let us denote by $\boldsymbol{r}(\mathrm{x}, \mathrm{y})$ the pixel vector at spatial coordinates (x, y). Similarly, let B be a kernel defined in the spatial domain of the image (the x - y plane). This kernel, usually called structuring element (SE) in mathematical morphology terminology, is translated over the image. The SE acts as a probe for extracting or suppressing specific structures of the image objects. Multidimensional morphological erosion at the pixel $\boldsymbol{r}(\mathrm{x}, \mathrm{y})$ can be defined as follows:

$$(\boldsymbol{r} \otimes B)(\mathrm{x}, \mathrm{y}) = \arg\_\mathrm{Min}_{(\mathrm{s},\mathrm{t}) \in B} \left\{ \sum_{(\mathrm{s},\mathrm{t})} \mathrm{Dist}(\boldsymbol{r}(\mathrm{x}, \mathrm{y}), \boldsymbol{r}(\mathrm{x}+\mathrm{s}, \mathrm{y}+\mathrm{t})) \right\} \tag{1}$$

where Dist is the SAM distance. Multidimensional erosion defined above extracts the most highly mixed pixel vector in the B-neighborhood. A morphological erosion index (MEI) can be defined for each pixel $\boldsymbol{r}(\mathrm{x}, \mathrm{y})$ by calculating the SAM distance between the pixel provided by the erosion operation and the data centroid $\boldsymbol{c}$, i.e. $\mathrm{MEI}(\mathrm{x}, \mathrm{y}) = \mathrm{Dist}((\boldsymbol{r} \otimes B)(\mathrm{x}, \mathrm{y}), \boldsymbol{c})$. Unsupervised generation of a number t of training samples can be accomplished by selecting the t pixel vectors with higher associated MEI score, $\{\boldsymbol{t}_{\mathrm{i}}^{(\mathrm{Erosion})}\}_{\mathrm{i}=1}^{t}$.

2.3. Automated morphological algorithm for extraction of training samples from homogeneous pixels

The automated morphological endmember extraction (AMEE) algorithm[12] has been recently shown to be a technique that most likely extracts homogeneous pixels in hyperspectral data[9]. It is based on the multidimensional dilation operation, which is the complementary of multidimensional erosion in equation (1) and defined as follows:

$$(\boldsymbol{r} \oplus B)(\mathrm{x}, \mathrm{y}) = \arg\_\mathrm{Max}_{(\mathrm{s},\mathrm{t}) \in \mathrm{K}} \left\{ \sum_{(\mathrm{s},\mathrm{t})} \mathrm{Dist}(\boldsymbol{r}(\mathrm{x}, \mathrm{y}), \boldsymbol{r}(\mathrm{x}-\mathrm{s}, \mathrm{y}-\mathrm{t})) \right\} \tag{2}$$

where Dist is the SAM distance. Opposite to multidimensional erosion, multidimensional dilation extracts the pixel vector that is most spectrally distinct in the B-neighborhood. A morphological dilation index (MDI) can be defined for each pixel $\boldsymbol{r}(\mathrm{x}, \mathrm{y})$ by calculating the SAM distance between the pixel provided by the erosion operation and the pixel provided by the dilation operation, i.e. $\mathrm{MDI}(\mathrm{x}, \mathrm{y}) = \mathrm{Dist}((\boldsymbol{r} \otimes B)(\mathrm{x}, \mathrm{y}), (\boldsymbol{r} \oplus B)(\mathrm{x}, \mathrm{y}))$. Homogeneous selection can be then accomplished by selecting the t pixel vectors with higher associated MDI score, $\{\boldsymbol{t}_{\mathrm{i}}^{(\mathrm{AMEE})}\}_{\mathrm{i}=1}^{t}$, after a spatial/spectral region growing process in order to prevent selection of several pixels that belong to the same homogeneous area.

2.4. RX algorithm for extraction of training samples from anomalous pixels

Finally, an algorithm referred to as Reed-Xiaoli (RX) algorithm[13] can be used for detecting anomalies in hyperspectral scenes. It has been widely used in signal processing applications. The filter implemented by the RX algorithm is referred to as RX filter (RXF), and can be specified for each pixel vector $\boldsymbol{r}$ as follows

$$\delta^{\mathrm{RXF}}(\boldsymbol{r}) = (\boldsymbol{r} - \boldsymbol{\mu})^{\mathrm{T}} \boldsymbol{K}^{-1} (\boldsymbol{r} - \boldsymbol{\mu}) \tag{3}$$

where $\boldsymbol{\mu}$ is the sample mean and $\boldsymbol{K}$ is the sample data covariance matrix. A set of training samples, $\left\{\boldsymbol{t}_{\mathrm{i}}^{(\mathrm{RX})}\right\}_{\mathrm{i}=1}^{t}$, can be extracted by selecting the t pixel vectors with higher value of δ^{RXF}.

3. MULTI-LAYER PERCEPTRON FOR MIXED PIXEL CLASSIFICATION

The neural network classifier considered in experiments is based on MLP, a standard multi-layer architecture that can successfully approximate virtually any function when trained correctly[14]. This implies that training is the most important step in MLP-based classification. The architecture of the MLP used in this work comprises a number of identical processing units organized in layers, with those units on one layer connected to those on the next layer by means of weighted connections[15]. Fig. 1 shows the topology of the simple, fully connected three-layer MLP used in this work for mixed pixel classification. As shown in Fig. 1, the neuron count at the input layer, p, equals the number of estimated endmember classes by the HFC method. The input patterns are the endmember fractional abundances, denoted by $\boldsymbol{\gamma}^{(\boldsymbol{r})} = \left(\gamma_1^{(\boldsymbol{r})}, \gamma_2^{(\boldsymbol{r})}, \cdots, \gamma_p^{(\boldsymbol{r})}\right)$, estimated for each sample vector $\boldsymbol{r}$ by fully constrained least squares spectral unmixing (FLCSU)[16] using the endmembers provided by N-FINDR method in section 2.1. These abundance fractions are subject to $\gamma_{\mathrm{i}}^{(r)} \geq 0$ for all $1 \leq \mathrm{i} \leq p$ (abundance non-negativity constraint) and $\sum_{\mathrm{i}=1}^{p} \gamma_{\mathrm{i}}^{(r)} = 1$ (abundance sum-to-one constraint). On other hand, the output layer has the same number of neurons as the input layer. Finally, a specific problem domain dictates how to design the number of hidden layers and the number of neurons in the hidden layer. Since this paper is devoted to investigation of training methods, finding optimal network parameters for the MLP is beyond its scope. Based on previous results in the literature and our own experimentation[17], in this work we have set the number of hidden neurons empirically to $2 \times p$.

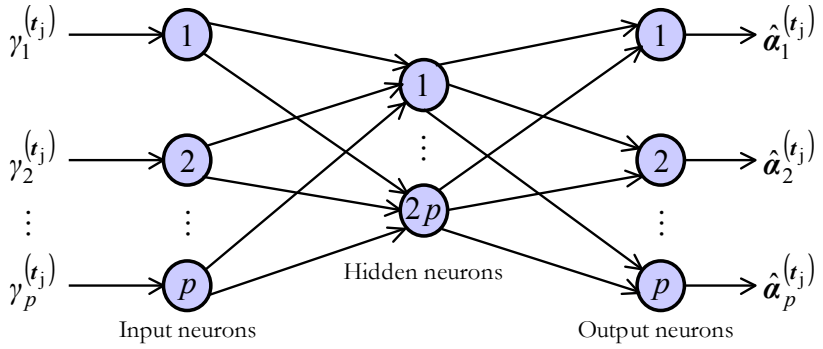


Fig. 1. Architecture of multi-layer perceptron (MLP) neural network for mixed pixel classification.

The usage of the MLP neural architecture in Fig. 1 involves two phases: training and classification. MLP models are typically trained using the error back-propagation (EBP) algorithm[14], a supervised learning-from-data technique of training. Let us denote by $\boldsymbol{\gamma}^{(\boldsymbol{t}_\mathrm{j})} = \left(\gamma_1^{(\boldsymbol{t}_\mathrm{j})}, \gamma_2^{(\boldsymbol{t}_\mathrm{j})}, \cdots, \gamma_p^{(\boldsymbol{t}_\mathrm{j})}\right)$ the vector of fractional abundances estimated by FCLSU for the j-th training sample, $\boldsymbol{t}_\mathrm{j}$, used in the EBP algorithm, and let $\boldsymbol{\alpha}^{(\boldsymbol{t}_\mathrm{j})} = \left(\alpha_1^{(\boldsymbol{t}_\mathrm{j})}, \alpha_2^{(\boldsymbol{t}_\mathrm{j})}, \cdots, \alpha_p^{(\boldsymbol{t}_\mathrm{j})}\right)$ be the desired vector of contributions, i.e. the true proportions vector for that pixel. In the training phase, a set of t labeled input-output training

vector pairs $\left\{\left(\boldsymbol{t}_{\mathrm{j}}, \boldsymbol{\alpha}^{(\boldsymbol{t}_{\mathrm{j}})}\right)\right\}_{\mathrm{j}=1}^{t}$ are presented to the network, which computes an error between the output vector $\hat{\boldsymbol{\alpha}}^{(\boldsymbol{t}_{\mathrm{j}})} = \left(\hat{\alpha}_1^{(\boldsymbol{t}_{\mathrm{j}})}, \hat{\alpha}_2^{(\boldsymbol{t}_{\mathrm{j}})}, \cdots, \hat{\alpha}_p^{(\boldsymbol{t}_{\mathrm{j}})}\right)$, that can be seen as an estimate of the endmember abundance fractions of $\boldsymbol{t}_{\mathrm{j}}$, and the vector of desired values for each output unit. This error is propagated successively back through the network and the matrix of weights $\boldsymbol{W}$ is progressively updated until the network approximates the desired output closely enough. An MSE-based measure of how much the network is deviating from the desired performance can be expressed as follows:

$$\mathrm{MSE}(\boldsymbol{W}) = \frac{1}{2}\sum_{\mathrm{j}=1}^{t}\sum_{\mathrm{i}=1}^{p}\left(\alpha_{\mathrm{i}}^{(\boldsymbol{t}_{\mathrm{j}})} - \hat{\alpha}_{\mathrm{i}}^{(\boldsymbol{t}_{\mathrm{j}})}\right)^2 \tag{4}$$

where t is the total number of training patterns, p is the number of endmember classes, $\alpha_{\mathrm{i}}^{(\boldsymbol{t}_{\mathrm{j}})}$ and $\hat{\alpha}_{\mathrm{i}}^{(\boldsymbol{t}_{\mathrm{j}})}$ represent the target and the network output for the j-th training sample, $\boldsymbol{t}_{\mathrm{j}}$, of the i-th class, $\boldsymbol{e}_{\mathrm{i}}$, respectively, where the training samples are generated by an unsupervised algorithm in section 2.

4. EXPERIMENTAL RESULTS

In this section, we conduct a quantitative and comparative analysis of linear and nonlinear SMA techniques using real mixture data with absolute ground-truth collected at Dr. John Mustard's laboratory, Brown University. Mustard's data sets consisted of 26 spectra collected using the RELAB spectrometer (a high resolution, bi-directional spectrometer), and included spectra from individual endmembers (Anorthosite, Enstatite, Magnetite and Olivine), along with both binary and ternary mixtures of these endmembers. Within the 26 considered data sets, there were 15 binary mixtures, seven ternary mixtures and four endmember spectra. For illustrative purposes, Fig. 2 shows the spectral signatures of the four mineral endmembers. In order to demonstrate the impact of training on the proposed neural network model, spectra of true nonlinear mixtures from Mustard's database were used in computer simulations to design various scenarios in order to verify the utility of the proposed model in solving realistic pixel unmixing problems.

4.1. Generation of a synthetic image made up of true mixture data

The reflectance spectra of pure mineral spectra in Fig. 2, along with available nonlinear mixtures of the above endmembers, were used to create a synthetic hyperspectral scene as follows. First, a background image with size of 20×20 pixels was generated by pixels made up of ternary mixtures of Anorthosite/Enstatite/Olivine, using the abundance fractions described in Table 1. Next, three sets of panels $\left\{\boldsymbol{p}_{\mathrm{ij}}^{\mathrm{k}}\right\}_{\mathrm{i=1,j=1}}^{2,2}$, $\left\{\widetilde{\boldsymbol{p}}_{\mathrm{ij}}^{\mathrm{k}}\right\}_{\mathrm{i=1,j=1}}^{2,2}$ and $\left\{\overline{\boldsymbol{p}}_{\mathrm{ij}}^{\mathrm{k}}\right\}_{\mathrm{i=1,j=1}}^{2,2}$, $\mathrm{k}=1,2,3$, each of which made up of three 2×2 panels, were created by using pure spectra and true nonlinear mixtures with abundance fractions in Table 1. The notations of $\boldsymbol{p}$, $\widetilde{\boldsymbol{p}}$ and $\hat{\boldsymbol{p}}$ denote three sets of the pixels in the 2×2 panels in rows 1, 2, 3, respectively, while the superscript "*k*" denotes the *k*-th 2×2 panel in each of the three sets. For example, the first set of the three 2×2 panels $\left\{\boldsymbol{p}_{\mathrm{ij}}^{\mathrm{k}}\right\}_{\mathrm{i=1,j=1}}^{2,2}$ was simulated as follows. The four pixels in the first 2×2 panel, $\left\{\boldsymbol{p}_{\mathrm{ij}}^{1}\right\}_{\mathrm{i=1,j=1}}^{2,2}$ were all simulated by 100% pure Enstatite signature. For the second 2×2 panel, the top left pixel $\boldsymbol{p}_{11}^{2}$ was simulated by a true mixture of (90%-Enstatite/10%-Olivine); the top right pixel $\boldsymbol{p}_{12}^{2}$ was simulated by (10%-Enstatite/90%-Olivine), while the two bottom left and right pixels $\boldsymbol{p}_{21}^{2}, \boldsymbol{p}_{22}^{2}$ were simulated by true ternary mixtures using the fractional abundances in Table 1. For the third 2×2 panel, its top left pixel, $\boldsymbol{p}_{11}^{3}$ was simulated by 100% pure Enstatite and the other three (i.e., top right, bottom left and bottom right pixels), $\boldsymbol{p}_{12}^{3}, \boldsymbol{p}_{21}^{3}, \boldsymbol{p}_{22}^{3}$, were simulated by true binary mixtures of (75%-Enstatite/25%-Olivine), (50%-Enstatite/50%-Olivine), and (25%-Enstatite/75%-Olivine) respectively. As shown in Table 1, the 2nd and 3rd sets of the three 2×2 panels, $\left\{\widetilde{\boldsymbol{p}}_{\mathrm{ij}}^{\mathrm{k}}\right\}_{\mathrm{i=1,j=1}}^{2,2}$ and $\left\{\overline{\boldsymbol{p}}_{\mathrm{ij}}^{\mathrm{k}}\right\}_{\mathrm{i=1,j=1}}^{2,2}$ were simulated by using Anorthosite and Magnetite instead of Enstatite, respectively, where the bottom pixels of the second 2×2 panel, i.e. $\widetilde{\boldsymbol{p}}_{21}^{2}, \widetilde{\boldsymbol{p}}_{22}^{2}$ and $\overline{\boldsymbol{p}}_{21}^{2}, \overline{\boldsymbol{p}}_{22}^{2}$ were again simulated by using true ternary mixtures.

	Anorthosite	Enstatite	Magnetite	Olivine
$\{\boldsymbol{p}_{ij}^1\}_{i=1,j=1}^{2,2}, \boldsymbol{p}_{11}^3$	-	100	-	-
$\boldsymbol{p}_{11}^2$	-	90	-	10
$\boldsymbol{p}_{12}^2$	-	10	-	90
$\boldsymbol{p}_{21}^2$	16.16	16.24	-	67.70
$\boldsymbol{p}_{22}^2$	16.13	67.85	-	16.02
$\boldsymbol{p}_{12}^3$	-	75	-	25
$\boldsymbol{p}_{21}^3$	-	50	-	50
$\boldsymbol{p}_{22}^3$	-	25	-	75
$\{\widetilde{\boldsymbol{p}}_{ij}^1\}_{i=1,j=1}^{2,2}, \widetilde{\boldsymbol{p}}_{11}^3$	100	-	-	-
$\widetilde{\boldsymbol{p}}_{11}^2$	90	-	-	10
$\widetilde{\boldsymbol{p}}_{12}^2$	10	-	-	90
$\widetilde{\boldsymbol{p}}_{21}^2$	67.81	15.99	-	16.20
$\widetilde{\boldsymbol{p}}_{22}^2$	16.05	41.83	-	41.12
$\widetilde{\boldsymbol{p}}_{12}^3$	75	-	-	25
$\widetilde{\boldsymbol{p}}_{21}^3$	50	-	-	50
$\widetilde{\boldsymbol{p}}_{22}^3$	25	-	-	75
$\{\overline{\boldsymbol{p}}_{ij}^1\}_{i=1,j=1}^{2,2}, \overline{\boldsymbol{p}}_{11}^3$	-	-	100	-
$\overline{\boldsymbol{p}}_{11}^2$	-	-	90	10
$\overline{\boldsymbol{p}}_{12}^2$	-	-	10	90
$\overline{\boldsymbol{p}}_{21}^2$	41.83	41.77	-	16.40
$\overline{\boldsymbol{p}}_{22}^2$	41.92	16.11	-	41.97
$\overline{\boldsymbol{p}}_{12}^3$	-	-	75	25
$\overline{\boldsymbol{p}}_{21}^3$	-	-	50	50
$\overline{\boldsymbol{p}}_{22}^3$	-	-	25	75
Background	33.61	33.03	-	33.36

Table 1. Abundance fractions in percentage of three panel sets, each of which has three 2×2 panels, simulated by pure signatures and true nonlinear mixtures of Anorthosite, Enstatite, Magnetite and Olivine.

The three sets of three 2×2 panels above, illustrated in Fig. 3(a), were then implanted in order, i.e. 1st row: Enstatite $\{\boldsymbol{p}_{ij}^k\}_{i=1,j=1}^{2,2}$; 2nd row: Anorthosite $\{\widetilde{\boldsymbol{p}}_{ij}^k\}_{i=1,j=1}^{2,2}$; 3rd row: Magnetite $\{\overline{\boldsymbol{p}}_{ij}^k\}_{i=1,j=1}^{2,2}$ in the background image at the center to generate a synthetic image shown in Fig. 3(b). It should be noted that all the background pixels in the synthetic scene were considered to be homogenous pixels as well as mixed pixels. Similarly, the pure pixels in the first column in Fig. 3(b) are also homogeneous. On other hand, the pure pixels at the top left of the 2×2 panels in the third column in Fig. 3(b), $\boldsymbol{p}_{11}^3, \widetilde{\boldsymbol{p}}_{11}^3, \overline{\boldsymbol{p}}_{11}^3$ are also anomalous pixels since their signatures are spectrally distinct from the spectral signatures of their neighboring pixels. All other panel pixels are mixed and anomalous with various mixtures of different spectral signatures, where $\boldsymbol{p}_{21}^2, \boldsymbol{p}_{22}^2, \widetilde{\boldsymbol{p}}_{21}^2, \widetilde{\boldsymbol{p}}_{22}^2, \overline{\boldsymbol{p}}_{21}^2, \overline{\boldsymbol{p}}_{22}^2$ are made up of ternary mixtures and the other pixels of binary mixtures.

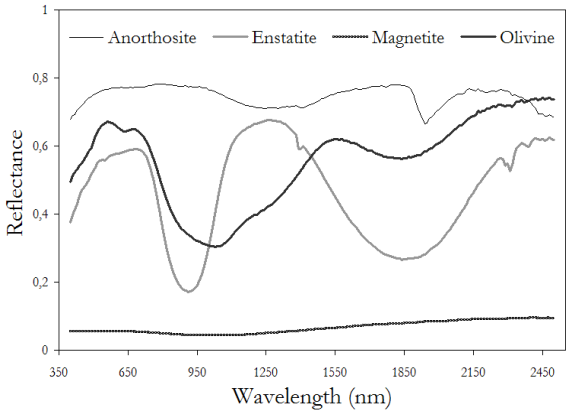
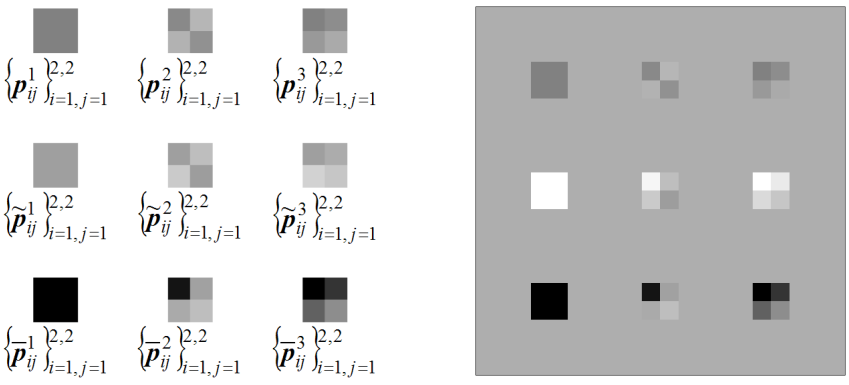


Fig. 2. Pure mineral spectra in Mustard's data.

4.2. Unsupervised generation of training samples

According to the synthetic image described above, there were 25 distinct signatures, of which there were three pure signatures, Enstatite, Anorthosite, Magnetite, fifteen binary mixtures, and seven ternary mixtures. Therefore, in this synthetic image, $p = 3$. Using this value as the desired number of endmembers, the N-FINDR algorithm found the pure training samples $\{\boldsymbol{t}_i^{(\text{N-FINDR})}\}_{i=1}^{3}$ shown in Fig. 4(a). The morphological erosion-based algorithm produced a set of 22 distinct mixed training samples $\{\boldsymbol{t}_i^{(\text{Erosion})}\}_{i=1}^{22}$ [see Fig. 4(b)]. Interestingly, this algorithm first exhausted all ternary mixtures before selection of binary mixtures. As can be noted in Fig. 4(b), the morphological erosion algorithm always extracted first the most highly mixed available samples (e.g. binary mixtures made up of 50% of one material were extracted before the other binary mixtures), thus showing a performance that is completely opposite to N-FINDR algorithm in Fig. 4(a), which extracts the purest available samples. On other hand, Fig. 4(c) shows the set of four training samples $\{\boldsymbol{t}_i^{(\text{AMEE})}\}_{i=1}^{4}$ produced by AMEE algorithm. The AMEE-selected pixels represent three pure panel pixels and one background pixel, all of which were considered as homogeneous pixels due to the region growing process included in AMEE, where homogeneous pixels of the same type were not extracted once the first pixel of its type is extracted. In contrast, the performance of RX algorithm was completely opposite. As shown in Fig. 4(d), the RX algorithm selected all mixed panel training samples plus three pure pixels, $\boldsymbol{p}_{11}^{3}$, $\tilde{\boldsymbol{p}}_{11}^{3}$, $\bar{\boldsymbol{p}}_{11}^{3}$ located at the upper left corner of the third panel in each of the three rows. All the extracted pixels $\{\boldsymbol{t}_i^{(\text{RX})}\}_{i=1}^{24}$ represented spectrally distinct signatures in their surroundings and were considered as anomalous pixels. The RX algorithm did not extract any background pixel because they were all considered as homogenous pixels.



(a) (b)

Fig. 3. Three sets of 2×2 simulated panels made up of true mixtures from Mustard's data. (b) Synthetic image.

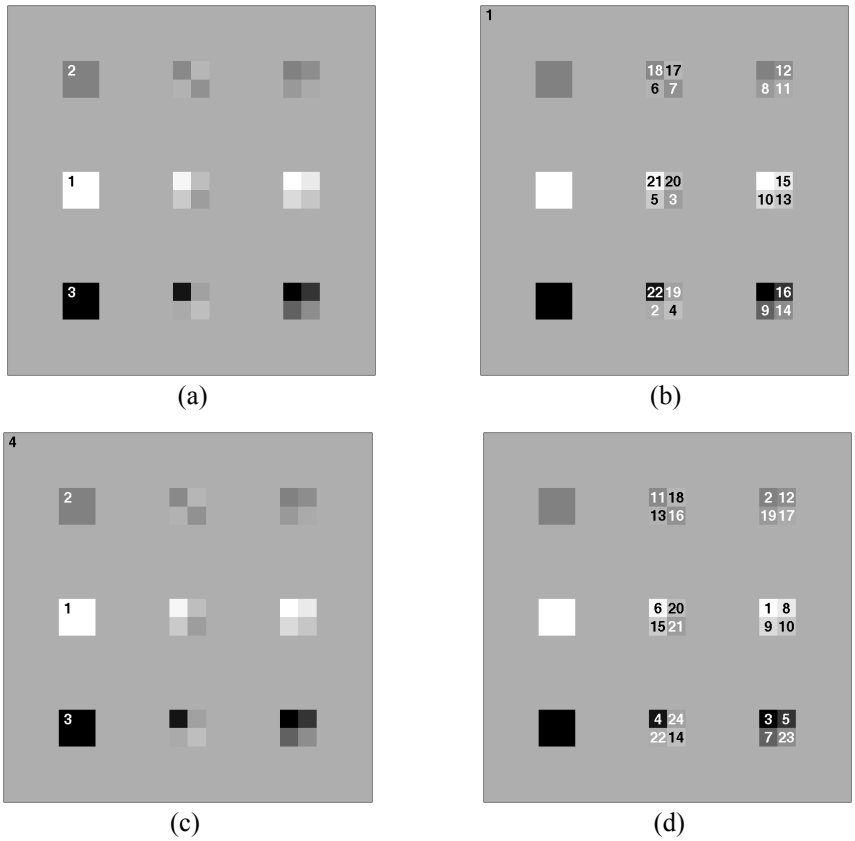


Fig. 4. (a) Pure pixels extracted by N-FINDR algorithm; (b) Mixed pixels extracted by morphological erosion-based algorithm; (c) Homogeneous pixels extracted by AMEE algorithm; (d) Anomalous pixels selected by RX algorithm.

4.3. Fractional abundance estimation results

The training samples above were used to train our neural network model, where Table 2 shows the individual and average RMSE scores in fractional abundance estimation using FCLSU and the proposed MLP classifier trained with different combinations of pure and mixed training samples, provided by N-FINDR and morphological erosion algorithms, respectively. Each column in the table represents a training-testing experiment, where pixels labeled as "Training" were used to train the MLP and the other pixels were used for testing. The score associated to each unlabeled pixel represents the individual RMSE in fractional abundance estimation at that pixel. As can be noted from Table 2, our general approach was to use selected training samples sequentially, as generated by the tested algorithms. For instance, the in case of N-FINDR algorithm only a training-testing scenario in which the three extracted pixels in Fig. 4(a) were used for training and the other pixels were used for testing. Interestingly, this training-testing case produced exactly the same results as FCLSU, which indicated that training the MLP with endmember pixels cannot provide more information than that obtained by a standard linear mixture model. On other hand, the first training sample selected by the morphological erosion algorithm was a background pixel (ternary mixture). The second column in Table 2 shows that, when this sample was incorporated to the previous set, the abundance estimation results in all ternary mixtures improved with respect to the case where only three endmembers were used for training. Similarly, the second training sample selected by morphological erosion corresponds to the pixel $\bar{\boldsymbol{p}}_{21}^{2}$, another ternary mixture. The incorporation of this sample to the training set resulted in the best abundance estimation results in the pixels made up of ternary mixtures, while the estimation at some other pixels made up of binary mixtures such as $\bar{\boldsymbol{p}}_{12}^{2}$, $\bar{\boldsymbol{p}}_{21}^{3}$ or $\bar{\boldsymbol{p}}_{22}^{3}$ was also improved. In order to see if the incorporation of additional pixels made up of ternary mixtures can improve the estimation results in the pixels made up of binary mixtures, we also tried a training-testing scenario in which all ternary mixtures, i.e. the first seven pixels in Fig. 4(b), were incorporated to the training set. Interestingly, although this case improved the estimation of many binary mixtures, other binary mixtures were better by FCLSU. This seems to indicate that binary mixtures are mostly linear in nature, while training samples made up of ternary mixtures are required to improve abundance

estimation in complex mixtures. In this regard, the background pixel extracted in first place by the morphological erosion algorithm demonstrated to be a very effective and useful training sample. Finally, it should be noted that the incorporation of additional training samples selected by the morphological erosion algorithm resulted in a moderate reduction of average RMSE scores. This result indicated that ternary mixtures contain the most significant information for complex mixture analysis.

	FCLSU	**N-FINDR**	**Morphological Erosion**		
$\{\boldsymbol{p}_{ij}^1\}_{i=1,j=1}^{2,2}, \boldsymbol{p}_{11}^3$	0.000	Training	Training	Training	Training
$\boldsymbol{p}_{11}^2$ (binary)	**0.008**	**0.008**	0.046	0.043	0.039
$\boldsymbol{p}_{12}^2$ (binary)	**0.014**	**0.014**	0.019	0.024	0.022
$\boldsymbol{p}_{21}^2$ (ternary)	0.066	0.066	**0.019**	0.028	Training
$\boldsymbol{p}_{22}^2$ (ternary)	0.080	0.080	0.025	**0.020**	Training
$\boldsymbol{p}_{12}^3$ (binary)	**0.016**	**0.016**	0.094	0.133	0.071
$\boldsymbol{p}_{21}^3$ (binary)	0.024	0.024	0.149	0.118	**0.021**
$\boldsymbol{p}_{22}^3$ (binary)	0.022	0.022	0.119	0.091	**0.019**
$\{\widetilde{\boldsymbol{p}}_{ij}^1\}_{i=1,j=1}^{2,2}, \widetilde{\boldsymbol{p}}_{11}^3$	0.000	Training	Training	Training	Training
$\widetilde{\boldsymbol{p}}_{11}^2$ (binary)	0.047	0.047	0.097	0.096	**0.020**
$\widetilde{\boldsymbol{p}}_{12}^2$ (binary)	0.010	0.010	0.036	0.045	**0.009**
$\widetilde{\boldsymbol{p}}_{21}^2$ (ternary)	0.158	0.158	0.156	**0.061**	Training
$\widetilde{\boldsymbol{p}}_{22}^2$ (ternary)	0.082	0.082	0.066	**0.043**	Training
$\widetilde{\boldsymbol{p}}_{12}^3$ (binary)	**0.073**	**0.073**	0.168	0.159	0.165
$\widetilde{\boldsymbol{p}}_{21}^3$ (binary)	0.089	0.089	0.178	0.160	**0.036**
$\widetilde{\boldsymbol{p}}_{22}^3$ (binary)	0.028	0.028	0.088	0.097	**0.011**
$\{\overline{\boldsymbol{p}}_{ij}^1\}_{i=1,j=1}^{2,2}, \overline{\boldsymbol{p}}_{11}^3$	0.000	Training	Training	Training	Training
$\overline{\boldsymbol{p}}_{11}^2$ (binary)	0.130	0.130	**0.054**	0.110	0.130
$\overline{\boldsymbol{p}}_{12}^2$ (binary)	0.201	0.201	0.085	**0.081**	0.164
$\overline{\boldsymbol{p}}_{21}^2$ (ternary)	0.141	0.141	**0.063**	Training	Training
$\overline{\boldsymbol{p}}_{22}^2$ (ternary)	0.127	0.127	0.094	**0.022**	Training
$\overline{\boldsymbol{p}}_{12}^3$ (binary)	0.112	0.112	**0.091**	0.112	0.132
$\overline{\boldsymbol{p}}_{21}^3$ (binary)	0.200	0.200	0.198	**0.161**	0.164
$\overline{\boldsymbol{p}}_{22}^3$ (binary)	0.258	0.258	0.237	**0.090**	0.250
Background	0.120	0.120	Training	Training	Training
Number of training samples	-	3	4	5	9
Average RMSE	0.112	0.112	0.089	0.084	**0.081**

Table 2. Individual and average RMSE in fractional abundance estimation obtained by FCLSU and MLP using different training-testing scenarios (in each column), where different sets of pure/mixed training samples were generated by N-FINDR and morphological erosion algorithm, respectively. The best estimation results for each panel are shown in bold typeface.

Results in Table 3 shows that using the four AMEE-generated training samples produces acceptable estimation results in terms of both individual and average RMSE scores. This is because a highly mixed sample, i.e. the fourth pixel extracted in Fig. 4(c) was included in the training set along with the first three endmember pixels. However, the second column in Table 3 shows that using the first four training samples generated by the RX algorithm in Fig. 4(d), which consisted of the same three pure samples plus a fourth sample made up of (90%-Magnetite/10%-Olivine) could not improve the results found by using only four AMEE-selected pixels as training samples. Interestingly, the incorporation of ten and eleven training samples extracted by RX cannot improve the results found using only 4 AMEE-generated samples.

	AMEE	**RX algorithm**		
$\{\boldsymbol{p}_{ij}^1\}_{i=1,j=1}^{2,2}, \boldsymbol{p}_{11}^3$	Training	Training	Training	Training
$\boldsymbol{p}_{11}^2$ (binary)	**0.046**	0.068	0.093	Training
$\boldsymbol{p}_{12}^2$ (binary)	**0.019**	0.024	0.039	0.079
$\boldsymbol{p}_{21}^2$ (ternary)	**0.019**	0.066	0.155	0.158
$\boldsymbol{p}_{22}^2$ (ternary)	**0.025**	0.116	0.222	0.107
$\boldsymbol{p}_{12}^3$ (binary)	**0.094**	0.098	0.230	0.103
$\boldsymbol{p}_{21}^3$ (binary)	**0.149**	0.151	0.295	0.163
$\boldsymbol{p}_{22}^3$ (binary)	**0.119**	0.148	0.127	0.122
$\{\widetilde{\boldsymbol{p}}_{ij}^1\}_{i=1,j=1}^{2,2}, \widetilde{\boldsymbol{p}}_{11}^3$	Training	Training	Training	Training
$\widetilde{\boldsymbol{p}}_{11}^2$ (binary)	0.097	0.217	Training	Training
$\widetilde{\boldsymbol{p}}_{12}^2$ (binary)	0.036	0.051	0.014	**0.020**
$\widetilde{\boldsymbol{p}}_{21}^2$ (ternary)	**0.156**	0.251	0.539	0.400
$\widetilde{\boldsymbol{p}}_{22}^2$ (ternary)	**0.066**	0.158	0.337	0.202
$\widetilde{\boldsymbol{p}}_{12}^3$ (binary)	**0.168**	0.261	Training	Training
$\widetilde{\boldsymbol{p}}_{21}^3$ (binary)	**0.178**	0.222	Training	Training
$\widetilde{\boldsymbol{p}}_{22}^3$ (binary)	**0.088**	0.122	Training	Training
$\{\overline{\boldsymbol{p}}_{ij}^1\}_{i=1,j=1}^{2,2}, \overline{\boldsymbol{p}}_{11}^3$	Training	Training	Training	Training
$\overline{\boldsymbol{p}}_{11}^2$ (binary)	0.054	Training	Training	Training
$\overline{\boldsymbol{p}}_{12}^2$ (binary)	0.085	0.041	**0.039**	0.076
$\overline{\boldsymbol{p}}_{21}^2$ (ternary)	**0.063**	0.211	0.451	0.301
$\overline{\boldsymbol{p}}_{22}^2$ (ternary)	**0.094**	0.209	0.349	0.268
$\overline{\boldsymbol{p}}_{12}^3$ (binary)	**0.091**	0.096	Training	Training
$\overline{\boldsymbol{p}}_{21}^3$ (binary)	**0.198**	0.229	Training	Training
$\overline{\boldsymbol{p}}_{22}^3$ (binary)	**0.237**	0.300	0.300	0.249
Background	Training	**0.156**	0.396	0.270
Number of training samples	4	4	10	11
Average RMSE	**0.089**	0.152	0.239	0.164

Table 3. Individual and average RMSE in fractional abundance estimation obtained by MLP using different training-testing scenarios (in each column), where different sets of homogeneous/anomalous training samples were generated by AMEE and RX algorithm, respectively. The best estimation results for each panel are shown in bold typeface.

Summarizing, results in Tables 3 and 4 reveal that intelligent training is required for neural network-based mixed pixel classification. In experiments, the spectral properties of training samples were found to be more important than their spatial properties. Specifically, the issue of whether a training sample corresponds to an anomalous pixel or not did not seem to play an important role. However, the degree of spectral purity of training samples was indeed a very important feature. In particular, it was found in experiments that training sets which contain the most highly mixed pixels can be very effective in fractional abundance estimation. Subsequently, there is a need for algorithms able to extract the most highly mixed pixels in the data, a behavior that is completely opposite to available endmember extraction algorithms designed to find the purest pixels available. One of the main contributions of this work has been the development of a new morphological erosion-based algorithm that seeks for the most highly mixed samples in the data set. This algorithm, when used in combination with N-FINDR algorithm for pure sample generation, produced the best training set in experiments.

To conclude this section, we must also point out that several other configurations for the generation of the synthetic scene above were tested. Specifically, the background was also simulated by a pure pixel made up of Olivine, and also by binary mixtures in different proportions. In all tested cases, the samples selected by AMEE and RX algorithm did not change because the spatial properties of the scene were not modified. However, spectrally guided techniques such as the morphological erosion algorithm always exhausted the most highly mixed pixels first. In experiments, the background image was also corrupted with Gaussian noise with signal-to-noise (SNR) ratio of 20:1. In this case, we did not have prior knowledge about how many endmembers were present in the image. In order to determine this number, we used the HFC method to estimate the VD based on various false alarm probabilities. According to our experiments, an appropriate estimate for the VD was 3, which resulted in essentially the same results for training sample generation algorithms as those addressed in Fig. 4. This also led to a very similar trend in abundance estimation errors. The same experiments were also conducted for different SNRs of 30:1 and 10:1, where the trend in abundance estimation (not included here) was also very similar.

CONCLUSIONS

Finding training samples without prior knowledge in hyperspectral data exploitation is a very challenging issue. In particular, the high spectral resolution provided by modern hyperspectral sensors allows one to extract and uncover many unknown substances, such as subpixel targets that cannot be identified *a priori*. Also, the complexity of mixing within pixels in hyperspectral imagery leads to the fact that such pixels are no longer considered pure, and traditional image processing techniques may not be applicable. Since hyperspectral imagery contains crucial information in subpixel targets and mixed pixels, a desired pool of training samples must include such information. This paper has explored the issues above by looking into what type of pixel information can be extracted from training samples in the context of neural network-based mixed pixel classification. Instead of relying on a single algorithm to generate training samples as it is the case in traditional image processing, we have investigated several unsupervised algorithms designed in accordance with the types of training samples that can be extracted from the viewpoint of pixel-level information. In order to effectively do so, four types of pixels were considered: pure pixel, mixed pixel, anomalous pixel and homogeneous pixel. Since these four types of pixels provide different levels of information, different sets of samples made up of these pixels were used for training. An extensive experimental study was conducted using Mustard's data set of mineral spectra, which contains true mixture data with known ground-cover proportions. Our experimental study revealed that a small number of intelligently selected training samples might be used to classify a hyperspectral data set as accurately as a larger training set derived in a conventional manner. The results illustrate the potential to direct training sample generation strategies to target the most useful training samples to allow efficient and accurate mixed pixel classification.

ACKNOWLEDGEMENT

The authors acknowledge Prof. John Mustard for providing the true mixture data used in experimental results. J. Plaza would like to acknowledge support received from Junta de Extremadura (local government) to carry out a research visit at Remote Sensing Signal and Image Processing Laboratory (RSSIPL). A. Plaza would also like to thank for support received from the Spanish Ministry of Education and Science (PR2003-0360 Fellowship).

REFERENCES

1. C.-I Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*, Kluwer Academic/Plenum Publishers, 2003.
2. R.A. Schwogengerdt, *Remote Sensing: Models and Methods for Image Processing*, 2nd. Ed., Academic Press, 1997, p. 447.
3. A. Plaza, P. Martinez, R. Perez and J. Plaza, "A quantitative and comparative analysis of endmember extraction algorithms from hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, pp. 650-663, 2004.
4. K. J. Guilfoyle, M. L. Althouse and C.-I Chang, "A quantitative and comparative analysis of linear and nonlinear mixture models using radial basis function neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, pp. 2314-2318, 2001.
5. W. Liu and E. Y. Wu, "Comparison of non-linear mixture models: sub-pixel classification," *Remote Sensing of Environment*, vol. 94, pp. 145-154, 2005.
6. A. Baraldi, E. Binaghi, P. Blonda, P. A. Brivio and A. Rampini, "Comparison of the multi-layer perceptron with neuro-fuzzy techniques in the estimation of cover class mixture in remotely sensed data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, pp. 994-1005, 2001.
7. G. M. Foody, M. B. McCulloch and W. B. Yates, "The effect of training set size and composition on artificial neural network classification," *International Journal of Remote Sensing*, vol. 16, pp. 1707-1723, 1995.
8. G. M. Foody and A. Mathur, "Toward intelligent training of supervised image classifications: directing training data acquisition for SVM classification," *Remote Sensing of Environment*, vol. 93, pp. 107-117, 2004.
9. C.-I Chang and A. Plaza, "What pixel information can be extracted from hyperspectral images?," *IEEE Transactions on Geoscience and Remote Sensing* (submitted, under review).
10. M.E. Winter, "N-FINDR: an algorithm for fast autonomous spectral endmember determination in hyperspectral data," *Imaging Spectrometry V*, Proc. SPIE 3753, pp. 266-277, 1999.
11. C.-I Chang and Q. Du, "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 42, no. 3, pp. 608-619, March 2004.
12. A. Plaza, P. Martinez, R. Perez and J. Plaza, "Spatial/spectral endmember extraction by multidimensional morphological operations," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 40, no 9, pp. 2025-2041, September 2002.
13. I. S. Reed and X. Yu, "Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution," *IEEE Trans. on Acoustic, Speech and Signal Process.*, vol. 38, no. 10, pp. 1760-1770, Oct. 1990.
14. C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.
15. J. Plaza, P. Martinez, R. Perez and A. Plaza, "Nonlinear neural network mixture models for fractional abundance estimation in AVIRIS hyperspectral images," in Proc. *XIII NASA/Jet Propulsion Laboratory Airborne Earth Science Workshop*, Pasadena, 2004.
16. D. Heinz and C.-I Chang, "Fully constrained least squares linear mixture analysis for material quantification in hyperspectral imagery," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 39, no. 3, pp. 529-545, March 2001.
17. J. Plaza, P. Martinez, A. Plaza and R. Perez, "Nonlinear neural network-based mixture model for estimating the concentration of nitrogen salts in turbid inland waters using hyperspectral imagery," *Chemical and Biological Standoff Detection II*, Proc. SPIE 5584, pp. 165-173, 2004.