Multispectral and Hyperspectral Image Analysis with Convex Cones

Agustin Ifarraguerri, *Student Member, IEEE,* and Chein-I Chang, *Senior Member, IEEE*

Abstract—A new approach to multispectral and hyperspectral image analysis is presented. This method, called convex cone analysis (CCA), is based on the fact that some physical quantities such as radiance are nonnegative. The vectors formed by discrete radiance spectra are linear combinations of nonnegative components, and they lie inside a nonnegative, convex region. The object of CCA is to find the boundary points of this region, which can be used as endmember spectra for unmixing or as target vectors for classification. To implement this concept, we find the eigenvectors of the sample spectral correlation matrix of the image. Given the number of endmembers or classes, we select as many eigenvectors corresponding to the largest eigenvalues. These eigenvectors are used as a basis to form linear combinations that have only nonnegative elements, and thus they lie inside a convex cone. The vertices of the convex cone will be those points whose spectral vector contains as many zero elements as the number of eigenvectors minus one. Accordingly, a mixed pixel can be decomposed by identifying the vertices that were used to form its spectrum. An algorithm for finding the convex cone boundaries is presented, and applications to unsupervised unmixing and classification are demonstrated with simulated data as well as experimental data from the hyperspectral digital imagery collection experiment (HYDICE).

Index Terms—Classification, convex cone analysis, hyperspectral digital imagery collection experiment (HYDICE), hyperspectral image, multispectral image, unmixing.

I. INTRODUCTION

SPECTRAL unmixing and object classification are two important goals of remote imaging spectrometry [1], [2]. Unmixing is used for geological and ecological research as well as for environmental monitoring. Classification is useful for automatic target recognition [3] and other machine vision applications. Unmixing and classification methods generally require a reference, i.e., a set of models or a library of spectra that is used to derive abundances for pixel unmixing, or a measure of distance or fit for classification. When no such reference is available, the analyst must use unsupervised methods to accomplish his/her goal. Principal components analysis (PCA) [4] and related transforms, such as maximum noise fraction (MNF) [5], are the tool of choice of many analysts as a first step. Spectral features are visualized in their spatial context by creating projection images. These images are arranged in order of decreasing variance or signal-to-noise ratio (SNR), and mutually orthogonal. A visual analysis often provides clues that make it possible to obtain estimates of the endmember or target spectra. The greatest drawback of this approach is that it usually requires a highly trained expert to study large amounts of data. In a situation in which large scenes must be analyzed quickly as a matter of routine, this approach may not be effective.

This paper introduces an approach that can lead to the automated, unsupervised analysis of multispectral or hyperspectral image scenes. The concept presented here is geometric in nature. It was introduced by Mavrovouniotis *et al.* [6], [7] and applied to the classification of biological samples using time-resolved pyrolysis-mass spectrometry. The idea of convex cone analysis (CCA) is based on the observation that some physical quantities, such as radiance (in the sense of photon counts) or mass spectra, are strictly nonnegative. The vectors formed by such spectra thus lie inside a convex region that contains the nonnegative spectra. The objective of CCA is to find the boundaries of this region as defined by its vertices. These corner spectra can then be used as endmember spectra for unmixing or target spectra for pixel classification.

CCA can be viewed as a form of factor analysis [4]. It can be applied to any type of multivariate data provided that it makes sense to impose a nonnegativity constraint. Given the number of components (i.e., targets for classification or endmembers for unmixing) in a scene c, we find the boundaries of the region that contains all positive linear combinations of the first c eigenvectors of the spectral correlation matrix. The region is convex [6] with a vertex at the origin, forming a cone, as shown in Fig. 1, with $c = 3$ (hence, the name). A slice perpendicular to the axis of the cone forms a convex polygon whose boundaries are formed by a number of corners (vertices) and edges. If the corner points and their associated spectra are interpreted as extreme components, every point inside the cone represents a linear combination of these components. This idea is closely linked to self modeling curve resolution (SMCR) [8]. SMCR was introduced by W. H. Lawton and E. A. Silvestre in 1971 to resolve two-component spectral mixtures without prior knowledge of the component spectra, using the concept described above. The "feasible regions," where pure component spectra must reside, are determined by geometric considerations. In [9], Hamilton and Gremperline reviewed the topic of SMCR as it applies to chemical mixture analysis. They described the progress and research directions since the original work, including extensions to three components and techniques for estimating the component spectra. A paper by
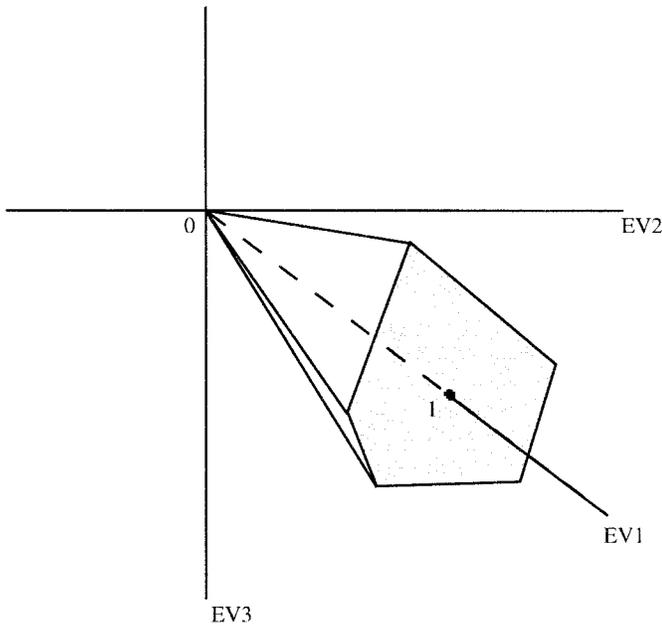
Fig. 1. Three-dimensional convex cone. Eigenvector (EV) 1–3 are the axes defined by the eigenvectors of the spectral correlation matrix. The shaded region is a slice of the positive subspace.

Borgen and Kowalski [10], also referenced in [8], proposed an SMCR method very similar to the one presented here, but with a focus on determining the feasible regions geometrically. In contrast to [10], our convex cone is determined algebraically, with geometric representations used only for visualization. Another similar concept was proposed by Craig [11] for application to remotely sensed data. He introduced two linear, nonorthogonal transforms: a dark-point-fixed (DPF) transform, and a fixed-point-free (FPF) transform that map the data onto a positive subspace defined by the smallest simplex that will contain all (rank-reduced) data points. The DPF transform is based on absolute coordinates (centered at the origin), while the FPF transform is barycentric (centered at the data). He referred to such transforms as minimum-volume transforms (MVT) and showed how they could be obtained using linear programming methods. The vertices of the simplex were then used as endmembers for unmixing. More recently, Boardman [12], [13] successfully demonstrated this method on AVIRIS scenes containing mixed pixels.

The paper is organized as follows. The application of CCA to imaging spectrometry and a new algorithm to compute the convex cone are described in Section II. Section III illustrates the method with examples and simulation results. Examples with experimental data are presented in Section IV, and the paper ends with some concluding remarks in Section V.

II. CONVEX CONE ANALYSIS

In this paper, we detail the concept of CCA with application to hyperspectral image analysis. To perform CCA, it is desirable to work with normalized pixel spectra (scaled to make the norm constant among all pixels) to remove the information related to the total radiance, while retaining the spectral pattern. The importance of normalization in CCA was discussed in [6] and [7].

A. Derivation of the Convex Cone

Given an $n \times m$ image with b bands, the image cube is rearranged into an $nm \times b$ matrix $\mathbf{S}$, where each row is a pixel spectrum and each column contains all of the pixels corresponding to a spectral band. The ordering of the pixels is not important. A $b \times b$ sample correlation matrix $\mathbf{C}$ is then obtained from the normalized spectral matrix $\mathbf{S}$

$$\mathbf{C} = \mathbf{S}^T \mathbf{S}. \tag{1}$$

The next step is to find the eigenvectors of $\mathbf{C}$ by diagonalizing $\mathbf{C}$ using the singular value decomposition (SVD) [14] for a square symmetric matrix

$$\mathbf{C} = \mathbf{PDP}^T \tag{2}$$

where $\mathbf{P}$ is the orthonormal matrix made up of the eigenvectors of $\mathbf{C}$ and $\mathbf{D}$ is a diagonal matrix containing the corresponding eigenvalues. The matrices $\mathbf{D}$ and $\mathbf{P}$ are of the same dimension as $\mathbf{C}$, and they are arranged such that the eigenvectors forming the columns of $\mathbf{P}$ are associated with the eigenvalues in the same column position in $\mathbf{D}$. By convention, the eigenvalues are arranged in decreasing order. We note that, in the original convex cone work, the matrix $\mathbf{P}$ was obtained by a direct application of the SVD to $\mathbf{S}$. We have chosen to work with the correlation matrix $\mathbf{C}$ because it is more computationally efficient given the typical number of data points produced by imaging spectrometry. This transformation, known as the Karthunen–Loève Transform, is very similar to PCA, with the important difference that the sample correlation matrix is used, rather than the covariance matrix.

To proceed from this point, we need to know *a priori* the number of components of interest c to model. This number must be no greater than the intrinsic dimensionality of $\mathbf{C}$ (which is equivalent to the number of nonzero eigenvalues) plus one. If $\mathbf{S}$ is full column rank prior to normalization, c can be no greater than b since normalization reduces the rank of $\mathbf{C}$ by one [15]. In practice, however, the "true" dimensionality of $\mathbf{C}$ is much smaller than the number of bands for hyperspectral data.

Given c, we select the eigenvectors corresponding to the c largest eigenvalues and look for the boundaries of the convex cone, where the linear combinations of these eigenvectors produce vectors that are strictly nonnegative

$$\mathbf{x} = \mathbf{p}_1 + a_1 \mathbf{p}_2 + \cdots + a_{c-1} \mathbf{p}_c \geq \mathbf{0}. \tag{3}$$

The $\mathbf{p}_i$ are the eigenvectors corresponding to the c largest eigenvalues and $\mathbf{0}$ is the zero vector. Since $\mathbf{x}$ is a convex linear combination, the coefficients of the $\mathbf{p}_i$ can be multiplied by a scaling factor so as to make the coefficient of $\mathbf{p}_1$ unity, giving $c - 1$ free parameters a_i. The $\mathbf{p}_1$ eigenvector, which corresponds to the largest eigenvalue, is aligned along the direction of the cone axis. For a matrix with nonnegative elements, such as a matrix of radiance or reflectance measurements, this eigenvector contains nonnegative elements only. To see this, consider that all b-dimensional vectors

that make the rows of the matrix $\mathbf{S}$ are located in the positive hyperquadrant, so that the first eigenvector will pass through the origin and point in the direction of the data, which contains only nonnegative values. Also, because the eigenvectors are mutually orthogonal, the remaining ones will necessarily have negative elements. Consequently, we can find sets of coefficients $\{a_i\}_{i=1}^{c-1}$ that produce a linear combination which contains $c-1$ elements of $\mathbf{x}$ that are exactly zero, with all of the other elements nonnegative. These points represent the corners of the convex cone.

B. Computing the Convex Cone

A method for the computation of the convex polygon was developed by Mavrovouniotis et al. for the cases of two–four components [16]. Unfortunately, no formal algorithm is provided in [16], and each case was treated differently. For the case of two components, the cone was obtained by simply ratioing the elements of the second and first eigenvectors and finding the two extreme values. In the case of three and four components, linear programming was used to find the coefficients $\{a_i\}_{i=1}^{c-1}$ by maximizing the norm of $\mathbf{a}$, subject to the constraint of (3). The optimization was repeated until all possible corners were found. No prescription was provided to extend this method beyond $c = 4$. We develop in this paper a new, simpler algorithm that works with arbitrary c at the expense of some computational complexity.

To derive the algorithm, we rewrite (3) as

$$\mathbf{x} = [\mathbf{p}_1 \cdots \mathbf{p}_c] \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_{c-1} \end{bmatrix} = \mathbf{Pa} \geq \mathbf{0} \qquad (4)$$

where $\mathbf{p}_i$ are b-dimensional column vectors. For $b > c$, $\mathbf{Pa} = \mathbf{0}$ is an overdetermined system of linear equations. If we view the elements of $\mathbf{P}$ as coefficients and those of $\mathbf{a}$ as variables, there are b equations of the form

$$p_{j1} + a_1 p_{j2} + \cdots + a_{c-1} p_{jc} = 0, \qquad \text{for } j = 1, \cdots, b \quad (5)$$

which define $(c-1)$-dimensional hyperplanes in c-dimensional space. Exact solutions can be found by taking $(c-1)$-tuples among the b equations. These solutions will produce linear combinations of the eigenvectors that have at least $c-1$ zeros. The boundary of the convex cone is the set of all solution vectors $\mathbf{a}$ that satisfy (3), or equivalently

$$\min(\mathbf{x}) = 0 \qquad (6)$$

where the minimum is taken over all $x_i \in \mathbf{x}$, $i = 1, \cdots, b$. In other words, a boundary occurs when at least one of the vector elements in the linear combination of eigenvectors is zero while the other elements are nonnegative. In practice, a small tolerance ε for negative numbers is introduced to allow for numerical errors. If a solution vector $\mathbf{a}$ from a $(c-1)$-tuple of (5) also satisfies (6), then $\mathbf{a}$ will be one of the corners of the convex cone. Fig. 2 shows a flowchart for the convex cone algorithm in the case where $c > 1$ (in the trivial case of $c = 1$, the solution is simply $\mathbf{p}_1$). It takes as input the first c eigenvectors $\mathbf{p}_1, \cdots, \mathbf{p}_c$. The output is a set of corner
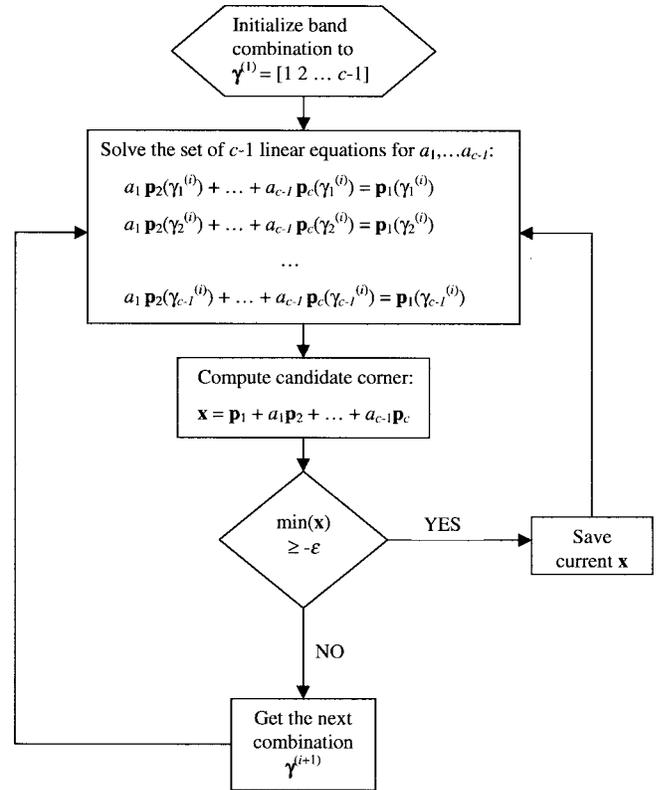

Fig. 2. Flowchart for convex cone algorithm. For $c > 1$, the algorithm searches through all possible combinations of $c - 1$ bands to find all sets of coefficients $a_1, \cdots, a_{c-1}$ that satisfy the constraint in (3). $\boldsymbol{\gamma}^{(i)}$ represents some combination of the bands.

spectra $\mathbf{x}_j$ that define the cone. The algorithm searches through all possible combinations of $c - 1$ bands to find all sets of coefficients $a_1, \cdots, a_{c-1}$ that satisfy (6). The vector $\boldsymbol{\gamma}(i)$ is used to represent the ith combination of $c - 1$ bands.

For b equations and $c - 1$ coefficients, there are $\binom{b}{c-1}$ possible solutions for $\mathbf{a}$, one for each $(c - 1)$-tuple of the rows of $\mathbf{P}$. For $c = 2$, there are b possible solutions for $\mathbf{a}$. As c increases, the number of solutions to be sought increases at $O(b^c)$. For a 100-band hyperspectral image and five components, there are nearly 4 million possible solutions. Clearly this algorithm is not computationally efficient for large b, but it is still practical for many situations given current computer capabilities. Since the solution is obtained by testing a large number of independent cases, a multiprocessor implementation can reduce computation time almost proportionally to the number of processors. In addition, there are preprocessing steps that can reduce the computation time if necessary. One is to reduce b by band selection or combination procedures. Recent work by Shen [17] indicates that it is feasible to greatly reduce the number of bands in hyperspectral data without significantly affecting object classification performance. This result has been confirmed by the authors and their colleagues [18]. We can also divide large scenes into sections and process each one separately, based on the premise that smaller scenes have less components. This divide-and-conquer approach is often used in automatic target recognition applications [3] to reduce the amount of computation.

C. Exploiting the Cone

After we have found the convex cone, we can use the spectra corresponding to the corners as endmember spectra for unmixing or target vectors for classification. There is no lack of methods in the literature for either. We will apply basic approaches here, which by no means represent the state of the art in classification or unmixing. For classification, the inner product of the normalized data with the normalized corners (i.e., spectral angle) provides a score that can be interpreted as a membership index. However, we have found that a spectral matched filter [19] provides better classification performance. In this case, we can compute the matched filters as follows:

$$\mathbf{m} = \mathbf{x}^T \mathbf{P}_{1:c} \mathbf{D}_{1:c}^{-1} \mathbf{P}_{1:c}^T \qquad (7)$$

where $\mathbf{x}$ is the corner spectrum and $\mathbf{P}_{1:c} \mathbf{D}_{1:c}^{-1} \mathbf{P}_{1:c}^T$ is the rank-reduced inverse of the correlation matrix $\mathbf{C}$. The inverse is computed using only the eigenvectors corresponding to the c largest eigenvalues for numerical stability. This filtering operation is equivalent to a spectral angle operation in which both the data and corner spectrum have been rank-reduced and sphered by the transformation $\mathbf{D}_{1:c}^{-(1/2)} \mathbf{P}_{1:c}^T$. It results in scores that are high for pixels which are similar to $\mathbf{x}$ and low for dissimilar pixels. The scores for each corner are then scaled so that the largest becomes one and the smallest becomes zero. This is consistent with the idea that, in the case of classification, for each target spectrum, there is at least one pixel that corresponds to it and at least one that does not. Finally, each pixel can be assigned to the (object) class with the maximum score.

If the application is unmixing, the standard approach is to use least-squares regression with the intercept forced through the origin. The operator in this case is

$$\mathbf{U} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \qquad (8)$$

where $\mathbf{X}$ is a $b \times c$ matrix whose columns can contain either the chosen corners of the convex cone, or representative "pure pixel" spectra obtained through a prior classification step (the "pure pixel" approach is also useful for subsequent endmember identification). The unmixing operator $\mathbf{U}$ is a $c \times b$ matrix that produces an estimated abundance vector when applied to a pixel spectrum. Because the true spectra of the endmembers is unknown, this method cannot give quantitative results without additional information. The estimated abundances will be proportional to the true abundances with an unknown proportionality constant. To obtain estimates of the true abundances, we would need additional information, such as the average abundances for the scene.

D. Corner Spectra

The method presented here makes no prior assumptions on the shape of the convex cone. In the case of two components, the positive subspace is confined to a line, so two corners are always produced that correspond, respectively, to the extreme positive and negative values of a_1. With three or more components, the positive subspace is a plane of two or more dimensions (hyperplane), so the number of vertices or corners can be greater than the number of components.

For example, if the data contain three distinct components, the two-dimensional (2-D) slice of the cone perpendicular to the cone axis will form a triangle. However, if the data have less than three components or the component spectra are highly collinear (i.e., have a spectral angle near one, where the spectral angle is the cosine of the angle between the vectors formed by the spectra), additional corners will be produced, forming an irregular convex polygon. Since we assume no prior information on the shape of the data space, it is not possible to predict the number of corners that the cone will have.

If more corners than components are produced, some must be eliminated by a suitable method. If $c = 3$, visual inspection of a slice of the cone works quite well [6], [7], but for larger numbers of components, it is necessary to use an automated method. For the unmixing application, there are variable selection techniques from regression analysis [20]. We have successfully selected corners by performing unmixing with all possible combinations and keeping the one that produces the maximum number of positive abundances. If several combinations produce all positive abundances, one is selected arbitrarily.

In the case of classification, feature selection techniques are available [21]. Our approach in this case is to compute matched filter scores for all corners and select the combination that minimizes some mutual correlation index, such as the condition number of the correlation coefficient matrix of each combination. In cases in which very large numbers of corners are produced (>20), this method can be computationally cumbersome. Fortunately, many of the corners are highly collinear, so we can reduce their number by checking for collinearity. These suggestions have worked for the authors in practice, but it is too soon to judge whether they are the best approach, as we are still experimenting with them.

III. IMPLEMENTATION OF CCA WITH SIMULATED DATA

For simplicity, we will only consider scenes with two and three components in this section. Using only a small number of components makes it possible to visualize the convex cone with 2-D plots. Scenes with three components can be visualized by projecting the data on to the second and third eigenvectors of the spectral correlation matrix to produce a scatter plot of the projection scores. Scenes with more components are more difficult to visualize and require additional automation in their processing (i.e., corner selection). Since the purpose of this paper is to illustrate idea of CCA, they are not treated here.

A. Two Simple Examples

Two simple ten-band multispectral scenes were composed to illustrate the implementation of CCA. The choice of ten spectral bands is a purely artificial one. In general, the number of bands required is larger than or equal to the number of components. The first image consists of a uniform background with an associated reflectance spectrum and a square "object" with a different spectrum. No noise is added. The object is a 33 × 33 pixel square centered on a 64 × 64 pixel
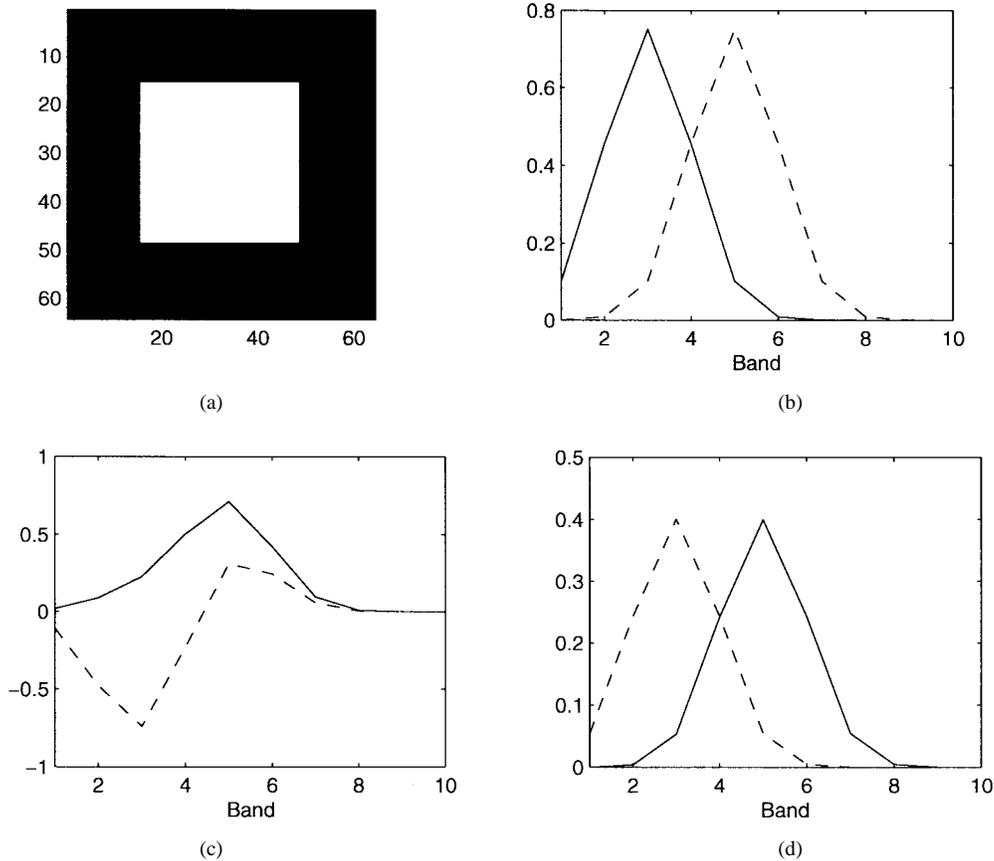
Fig. 3. Illustration of CCA for image segmentation: (a) original image, (b) background (dashed line) and object (solid line) spectra, (c) first (solid line) and second (dashed line) eigenvectors of the spectral correlation matrix, and (d) spectra corresponding to the convex cone corners.

image [Fig. 3(a)]. Fig. 3(b) shows the discrete spectra of the object and background. These spectra were generated using a Gaussian function with unit width constant and centered at bands 3 and 5 for the object and background, respectively. Normalization is not necessary in this case since all spectra have the same total intensity, so we can compute the SVD of the spectral correlation matrix directly. The eigenvectors corresponding to the two nonzero eigenvalues are plotted in Fig. 3(c). Note that one has all nonnegative elements and the other has both positive and negative elements. As pointed out earlier, this is a consequence of the orthogonality constraint. We now apply our algorithm to the eigenvectors and obtain a 2-D convex cone. The spectra corresponding to the corners are plotted in Fig. 3(d). The original spectra are recovered perfectly, but their identities have been lost; that is, the one-to-one correspondence can not be determined.

The second image uses the same spectra as the first, but in this case, each pixel is a mixture of the object and background with a small amount of added noise. The simulated image was created based on a simple linear model

$$\mathbf{r} = \mathbf{M}\alpha + n \qquad (9)$$

where r is a vector containing the simulated discrete spectrum, $\mathbf{M}$ is the matrix of endmember spectra, α is the abundance vector, and $\mathbf{n}$ is Gaussian random noise. The noise includes contributions from ambient (clutter) and instrumental sources. The true abundances were generated randomly and add to one.

Fig. 4(a) and (b) show the eigenvectors and the corner spectra. Again, the original spectra are recovered. Fig. 4(c) shows the scatter plot of the data and the two corners projected on to the eigenvector space. The corners appear toward the ends of the data cloud, at the locations where the pure pixels would show. Two lines passing through the origin and each corner are included to show the boundaries of the 2-D convex cone. For illustration purposes, the one-to-one correspondence between the endmember and the corner, which is normally not known, was determined in this example from the corner spectra in Fig. 4(b) so that we can visualize the performance by plotting the estimated versus the true abundances [Fig. 4(d)]. Because of the small amount of noise and low collinearity of the spectra, there is quantitative agreement between the estimated and true abundances.

B. Classification Simulations

Two parameters were modified to study the behavior of CCA: the SNR and the collinearity of the component spectra as measured by their spectral angle. For simplicity, we have assumed uniform illumination throughout the scene. Although this assumption is artificial, we make it here because the pixel spectra are typically normalized prior to computing the covariance matrix, so illumination information is discarded in any case.

For the simulations, we will define the SNR for each band as the ratio of the 50% reflectance signal level to the standard
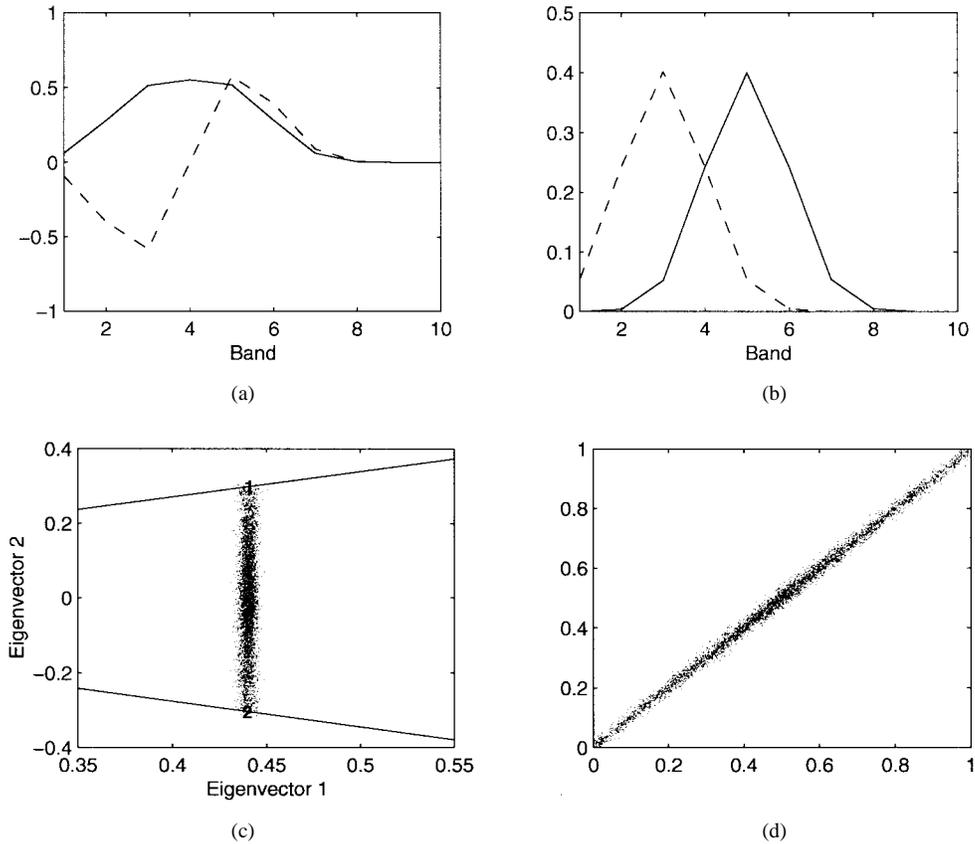
Fig. 4. Illustration of CCA for unmixing: (a) first (solid line) and second (dashed line) eigenvectors of the spectral correlation matrix, (b) spectra corresponding to the convex cone corners, (c) projections of the pixel spectra as well as the corners of the convex cone on to the eigenvector space, and (d) scatter plot of the estimated versus true abundances for the first endmember.

deviation of the noise [22]. This results in noise standard deviation that is roughly proportional to the average signal, a phenomenon that is often observed in radiometric data. The additive random noise was simulated by using numbers with a standard normal distribution obtained from a pseudorandom number generator. The simulated data were obtained by

$$\mathbf{r}_i = \left(\frac{\text{SNR}}{2} + \mathbf{n}_i \right) \cdot (\mathbf{M}\alpha_i) \qquad (10)$$

for each pixel $\mathbf{r}_i$. The simulated images were created by setting one component of the α vector to one and the rest to zero to indicate class membership. Scaling the signal by 50% of the SNR is equivalent to reducing the noise standard deviation by the inverse factor (2/SNR), so that the simulated data meets the SNR definition. The vector terms in the parentheses are multiplied element by element. To change the collinearity of the single-peak spectra, we keep the width of the peaks constant and change their relative location. As the peaks get closer, the spectral angle approaches one and we should see an SNR-dependent degradation in performance.

Two simulations were conducted for the two and three-class images, respectively. The single-object (two-class) image is the same as the example in section A. The two-object (three-class) image contains two 24 × 24 pixel objects in a 64 × 64 image with a uniform background. The objects are located in the top left and bottom right corners. For each image, SNR values of 5, 10, 20, and 40 were used. These

numbers are representative of what is typically found in real sensors, such as HYDICE (see Section IV). The peak for the background spectrum was kept centered at band 5, and the peak of the object spectra were moved toward it to increase the amount of overlap. For the image with one object, the peak was placed at 3.5, 4, 4.5, and 4.8, giving spectral angles of 0.57, 0.78, 0.94, and 0.99. For the two-object image, the second spectrum was located at 6.5, 6, 5.5, and 5.2 so that both objects have the same spectral angle with the background. In the cases with low SNR, the simulated data contained a small proportion of negative values. This situation is not unusual for radiometrically calibrated spectra. We deal with them here by simply setting them to zero. This approach can be used if the number of negative values is relatively small. This issue will be discussed further in Section IV.

We will examine the results of the segmentation of a single object using an SNR of ten and a spectral angle of 0.94. This situation is typical of experimental data. Fig. 5 shows (a) the original spectra of the object and background, (b) the eigenvalues of the covariance matrix, and (c) the corner spectra. Fig. 5(d) shows the projections of the pixel spectra as well as the corners of the convex cone on to the eigenvector space. The cone in this case is 2-D and has a vertex at the origin, with edges passing through each corner, as shown. Note that some of the pixels appear to lie outside the cone. Since the convex cone is obtained from a low rank subspace, the signal associated with the unused eigenvectors can cause
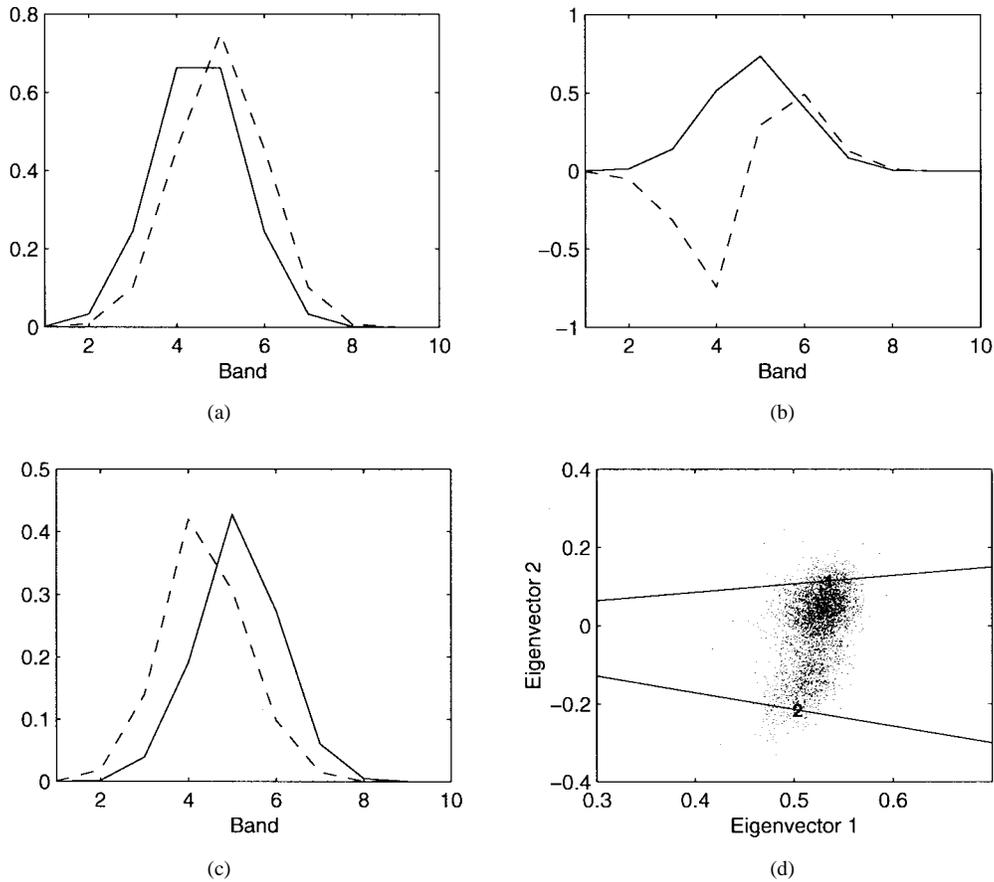
(a)

(b)

(c)

(d)

Fig. 5. Results of segmentation of noisy two-class image: (a) original spectra of the object (solid line) and background (dashed line), (b) first (solid line) and second (dashed line) eigenvectors of the spectral correlation matrix, (c) corner spectra, and (d) projections of the pixel spectra as well as the corners of the convex cone on to the eigenvector space.

some of the pixels to project outside the cone boundaries. Using the spectra from the corners, we create two matched filters and their corresponding output images [Fig. 6(a) and (b)]. Even though the output is noisy, we can easily see the object. The matched filter classifier works quite well in this case [Fig. 6(c)] in spite of the significant collinearity of the spectra. The segmentation produces the object, corrupted only by some "salt and pepper" noise. This noise can be nearly eliminated (at the expense of some image degradation) by using a 3 × 3 median filter [23]. Fig. 6(d) shows the filtered image.

To study a three-class case, we use the same SNR and a spectral angle of 0.78 (Fig. 7). We can visualize the result in two dimensions by projecting the data and the corners on to the second and third eigenvectors, as shown in Fig. 7(d). This is in essence a slice through the cone. We could visualize the full cone by plotting the first eigenvector in the third dimension. The corners would then become the edges of the three-dimensional (3-D) cone, with a vertex at the origin. In this case, there are quite a few pixels that lie outside the cone, and there are more than three corners (seven, in this case), but they are arranged in the shape of a triangle in the eigenvector space. We need to select the appropriate corners to proceed with the analysis. For this illustration, we choose corners 1, 2, and 7, which are closest to the data cloud. The resulting matched filtered output images are shown in Fig. 8(b)–(d).

They clearly correspond to the background and objects. The segmented image is shown in Fig. 8(e) and (f). Again a 3 × 3 median filter is used to clean the image.

Tables I and II show the ten-run average classification error rates for each combination of SNR and spectral angle. In the case of the two-class image, we have perfect classification down to an SNR of ten for well-separated spectra and good separation at five. As the spectral angle increases, the classification error rates increase. For highly similar spectra, the performance is good down to an SNR of 20, but degrades quickly thereafter. The results are similar with the three-class image, but there is faster degradation in performance with increasing spectral angle. For a spectral angle of 0.99, the performance is poor at all SNR values used.

C. Unmixing Simulations

Scenes with two and three endmembers and random abundance distributions were generated. The same SNR and spectral angle values were used as with the classification simulations. Fig. 9 shows the results of the CCA of a tree endmember scene with SNR of 20 and spectral angle of 0.78. Nine corners are produced in this case [Fig. 9(c)], again forming a triangle in the eigenvector space. We choose corners 1, 2, and 9 to use as endmembers and estimate the abundances. Even though these are not the vertices of the triangles, they
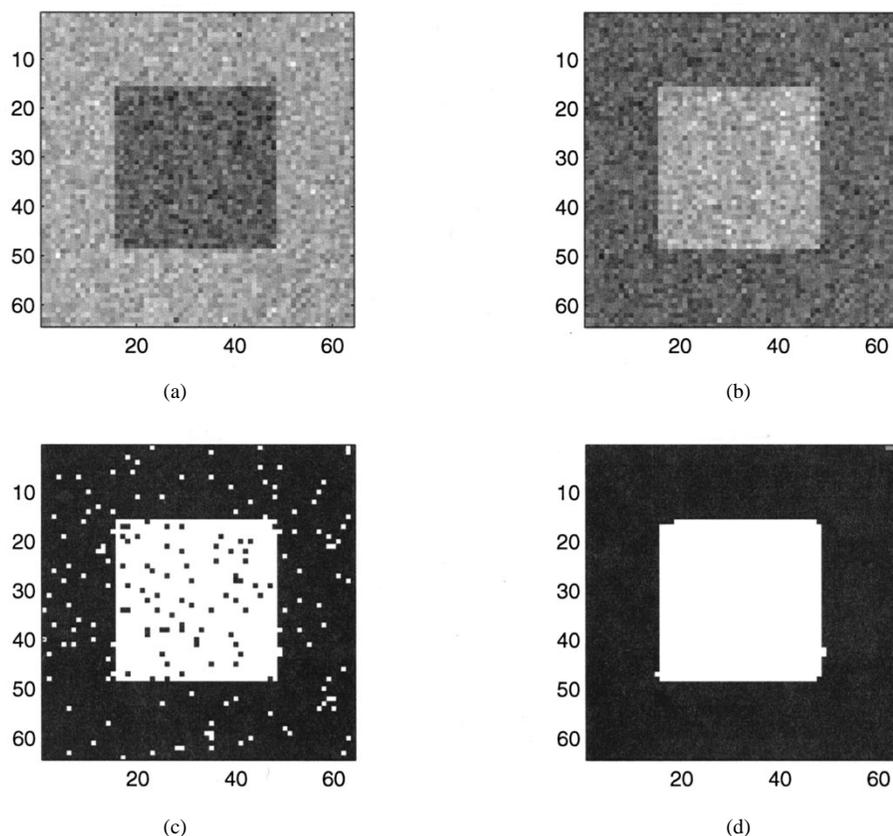
Fig. 6. Matched filter output and segmented images for two-class image: (a) corner 1, (b) corner 2, (c) segmented image, and (d) segmented image after median filtering.

are closer to the data, so the abundance estimates derived from these corners will be closer to the true abundances than estimates derived from the corners 2, 5, and 6. The resulting rms error in fractional abundance is 0.09. Fig. 9(d) shows a scatter plot of the true versus estimated fractional abundance of one of the endmembers. Again, the one-to-one correspondence was determined by comparing the chosen corners with the true spectra. Note also that the slope is less than unity. When the corners of the convex cone are further from the data than the true endmember spectra, the abundance will be underestimated. Conversely, if the chosen corners were closer to the data than the true endmember spectra, the abundances would be overestimated. This effect is inherent to the unsupervised nature of the method, and it is worsened by increased endmember collinearity or decreased SNR. The ten-run average rms errors for all simulation runs are presented in Tables III and IV. As in the case of classification, the rms error increases with reduced SNR or increased collinearity. Unlike the case of classification, there is no significant difference in performance between two and three endmembers.

IV. APPLICATION OF CCA TO EXPERIMENTAL DATA

We present in this section an application of CCA to experimental data. The purpose of these illustrations is to demonstrate some of the nuances and pitfalls associated with using real sensor data. We describe some additional preprocessing steps required to prepare the images for CCA. Of course, in

general, the preprocessing will depend on the specifics of the data under study.

The HYDICE sensor [24], [25] was designed to investigate the utility of hyperspectral imaging technology for military and civil applications. The sensor is operated from an aircraft in a pushbroom fashion, oriented at nadir, with 320 spatial pixels per line and 210 spectral bands created by dispersive optics. The spectral range extends from 400 to 2500 nm, with nominal wavelength spacing of 10 nm. Ground sampling distance ranges from 0.75 to 4 m, depending on aircraft altitude. The data were delivered with the radiometric calibration already done, so it is in units of $W/m^2/Sr/\mu m$. Spectral calibration, nonuniformity correction, and bad pixel compensation were also done prior to delivery.

To demonstrate the CCA method, we have selected 64 × 64 pixel portions from two HYDICE images (Fig. 10). The images contain the same scene but were collected at two different altitudes. The first (low altitude) contains two vehicles of the same type on a grass field with trees on the left edge. The second (high altitude) includes the objects from the first plus another two vehicles and an object. The task is to segment the objects or features for subsequent mapping and identification.

A. Preprocessing

Since we derive the target spectra from the data, it is not necessary to convert from radiance to apparent reflectance.
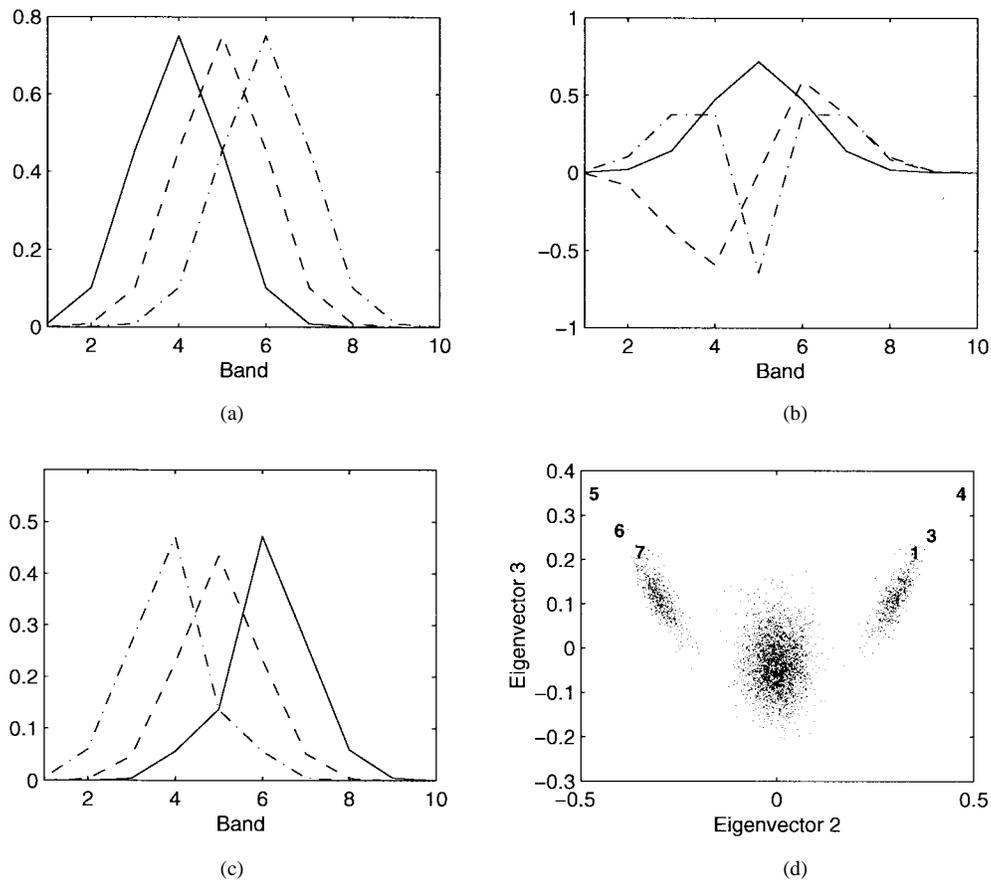
Fig. 7. Results of segmentation of a noisy three-class image: (a) original spectra of the objects (solid and dashdot lines) and background (dashed line), (b) first (solid line), second (dashed line) and third (dashdot line) eigenvectors of the spectral correlation matrix, (c) spectra corresponding to corners 1 (solid), 2 (dashed), and 7 (dashdot), and (d) projections of the pixel spectra as well as the corners of the convex cone on to the eigenvector space.

Instead, the spectra are normalized and a correction is done to deal with negative radiance values.

1) Normalization: The selected images contain significant illumination nonuniformity due to the shadows produced by the trees and vehicles, so careful normalization is critical. We will normalize each spectrum to unity. Prior to normalization, we must subtract the additive portion produced by a combination of atmospheric scattering (radiance not reflected by the surface) and any errors from the spectral calibration process (which can produce negative radiance values in low SNR bands). If normalization is done with this offset, a significant amount of unnecessary error will be introduced, which will reduce the class separability. To correct this, we use an approach similar to (but simpler than) the regression intersection method [26]. Prior to normalization, the offset is calculated for every band image using least-squares regression and subtracted from each pixel spectrum. The regressor in this case is the sum of all bands in the spectrum. Since the offset correction affects the sum, the regression is repeated until the baseline becomes acceptably small. It was found that three iterations are sufficient for this data.

2) Negative Values: A problem that must be corrected prior to the application of CCA is the presence of negative values in the data, which are caused by instrumental noise and calibration errors in the low SNR bands and the effects of our normalization process. Recall that a basic assumption is

that radiance values are strictly nonnegative. Having negative values in the first eigenvector may cause a significant portion of the data to fall outside the convex cone, and may even cause the algorithm to fail due to lack of a solution. To solve this problem, we could simply set any negative values to zero, but this will introduce additional variance. Adding a constant to every spectrum to make everything nonnegative is another possibility, but this requires keeping track of additional numbers and in practice increases the number of vertices in the convex cone. We have found empirically that the best approach is to determine the number of negative pixel values for every band image. If the number of negative values is greater than one per 1000 samples, we ignore the band completely. This somewhat arbitrary cutoff corresponds to approximately 3.1 standard deviations away from the mean (i.e., it is equivalent to deleting bands whose mean is less than 3.1 standard deviations from zero). The remaining negative values are then set to zero. This approach minimizes the amount of induced variance but protects from deleting useful bands because of an outlier and potentially reducing the amount of spectral information. After the negative values have been eliminated, the data are renormalized.

B. Results

Sample class spectra obtained from the first image are shown in Fig. 11(a). Their mutual spectral angles are 0.80 for
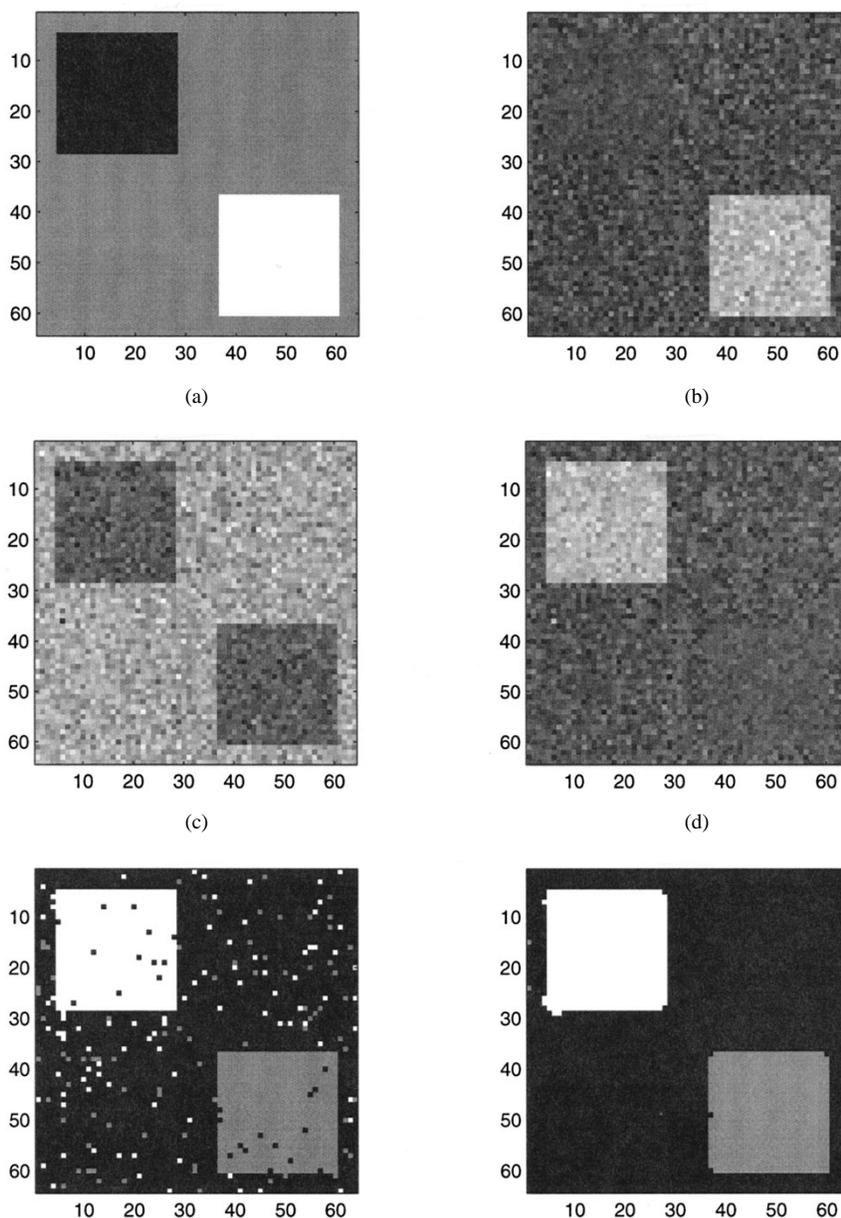
Fig. 8. Results of three-class segmentation of noisy image: (a) original scene, (b) matched filter output of corner 1, (c) matched filter output of corner 2, (d) matched filter output of corner 7, (e) segmented image, and (f) segmented image after median filtering.

TABLE I
ERROR RATES FOR TWO-CLASS CLASSIFICATION SIMULATION
AT FOUR SNR VALUES AND FOUR SPECTRAL ANGLES

SNR	Spectral Angle			
	0.5698	0.7786	0.9394	0.9901
5	0.0146	0.0719	0.2827	0.4407
10	0.0000	0.0003	0.0426	0.3672
20	0.0000	0.0000	0.0001	0.0724
40	0.0000	0.0000	0.0000	0.0009

TABLE II
ERROR RATES FOR THREE-CLASS CLASSIFICATION SIMULATION
AT FOUR SNR VALUES AND FOUR SPECTRAL ANGLES

SNR	Spectral Angle			
	0.5698	0.7786	0.9394	0.9901
5	0.2102	0.3552	0.4453	0.4590
10	0.0002	0.0762	0.3446	0.4578
20	0.0000	0.0000	0.2635	0.4336
40	0.0000	0.0000	0.0305	0.4214

the vehicles and trees, 0.90 for the vehicles and background, and 0.96 for the trees and background. The convex cone was computed for three components. In this case, there are a total of 11 corner spectra produced. We have chosen corners 1, 2, and 5 as targets for computation of matched filters based on the shape of the data space and the convex cone projection

in Fig. 11(b). In the eigenvector space, these corners form a simplex that contains all pixels. However, this choice of corners is not the only possible one. Corners that are close in this eigenvector space tend to produce similar matched filter scores. For instance, choosing corner 6 instead of 5 would make little or no difference to the final segmentation

(a)　　　　　　　　　　　　　　　　　　　　　(b)
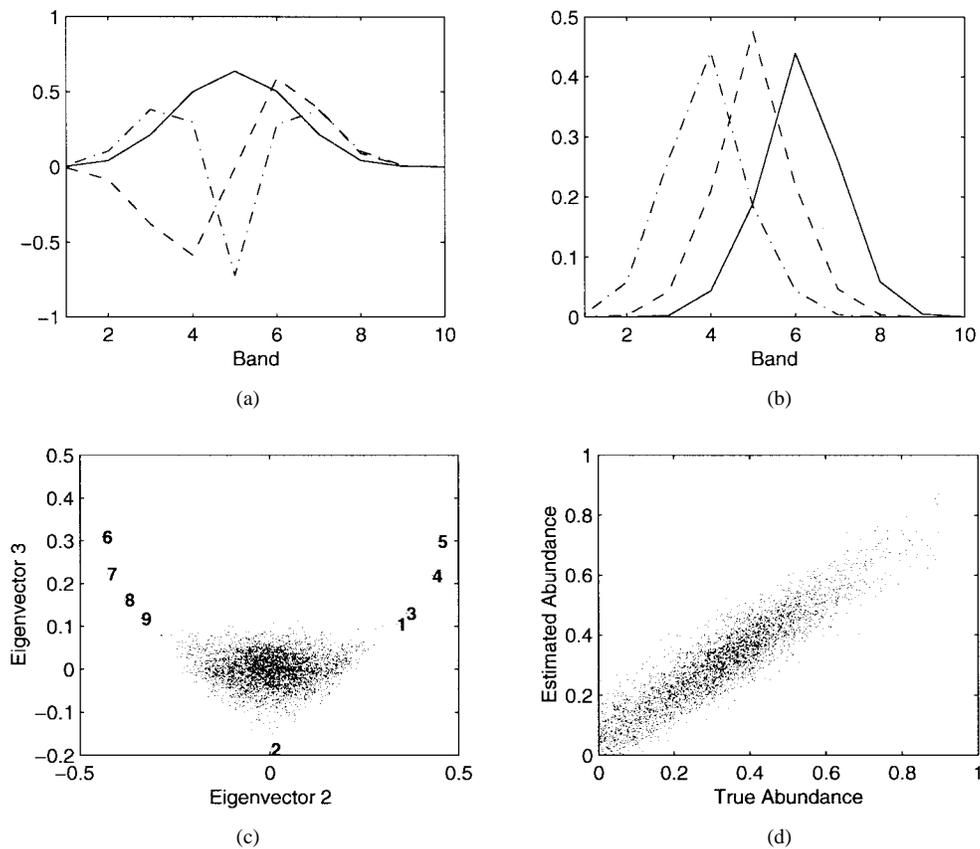
(c)　　　　　　　　　　　　　　　　　　　　　(d)

Fig. 9. Results of unmixing of a noisy three-endmember image: (a) first (solid line), second (dashed line), and third (dashdot line) eigenvectors, (b) spectra for corners 1 (solid), 2 (dashed), and 9 (dashdot), (c) projections of the pixel spectra as well as the corners of the convex hull on to the eigenvector space, and (d) scatter plot of the estimated versus true abundances for the first endmember.
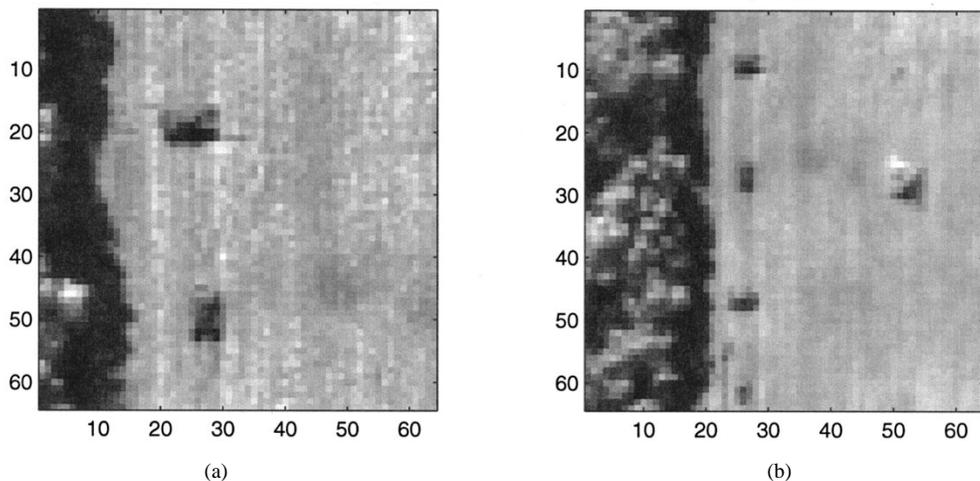

(a)　　　　　　　　　　　　　　　　　　　　　(b)

Fig. 10. HYDICE scenes selected for study: (a) low altitude and (b) high altitude.

results. The selected corner spectra are shown in Fig. 11(c) for comparison with the class spectra. In this case, the corner spectra are not generally similar to the class spectra. We can see in Fig. 11(b) that the corners are far apart from the data in eigenvector space, which can account for the lack of similarity. However, since we are interested in classification, there is no need for the corner spectra to match the class spectra as long as the results are demonstrably correct. If class spectra are needed for further processing (e.g., object identification), they can be easily obtained by finding the highest scoring pixels for each class. Fig. 11(d) shows the spectra corresponding to the highest scoring pixels in our test image. They closely match the class spectra in Fig. 11(a).

The matched-filtered score images for the chosen corners are shown in Fig. 12. We can see that image 12(a) correlates closely with the background 12(b) with the trees, and 12(c)
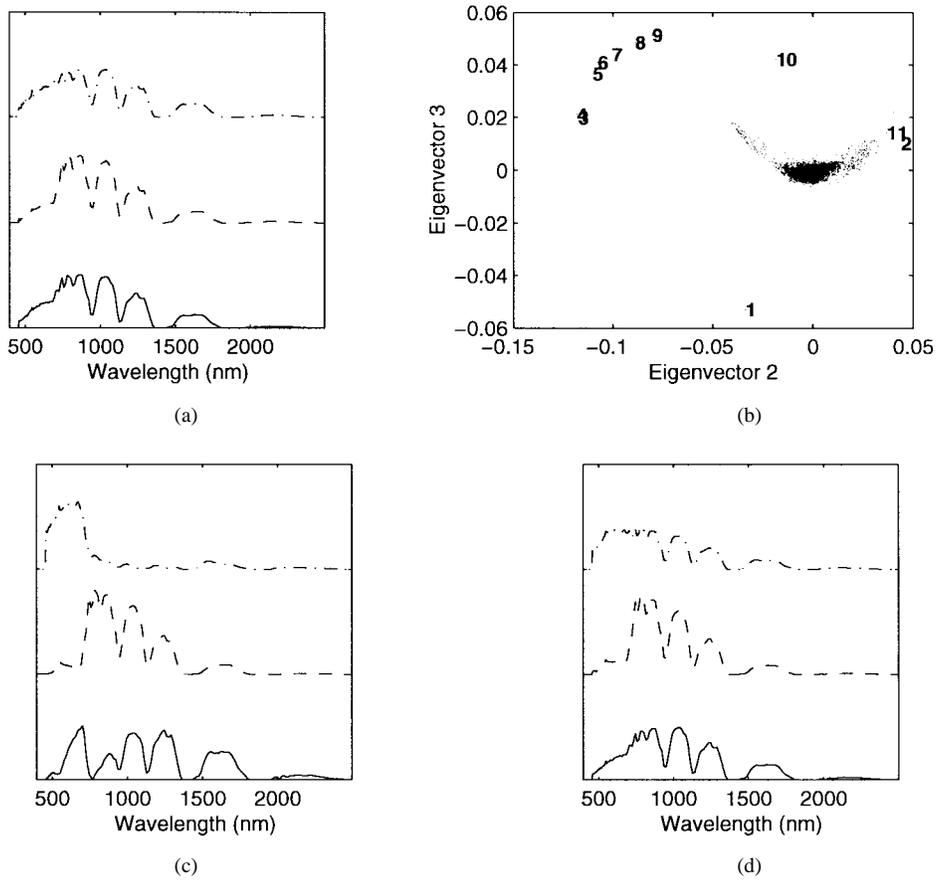
Fig. 11.   CCA of a three-class HYDICE scene: (a) sample spectra of the vehicles (solid), trees (dashed), and background (dashdot), (b) projections of the pixel spectra as well as the corners of the convex cone on to the eigenvector space, (c) spectra for corners 1 (solid), 2 (dashed), and 5 (dashdot), and (d) spectra associated with highest-scoring pixels for corners 1 (solid), 2 (dashed), and 5 (dashdot).
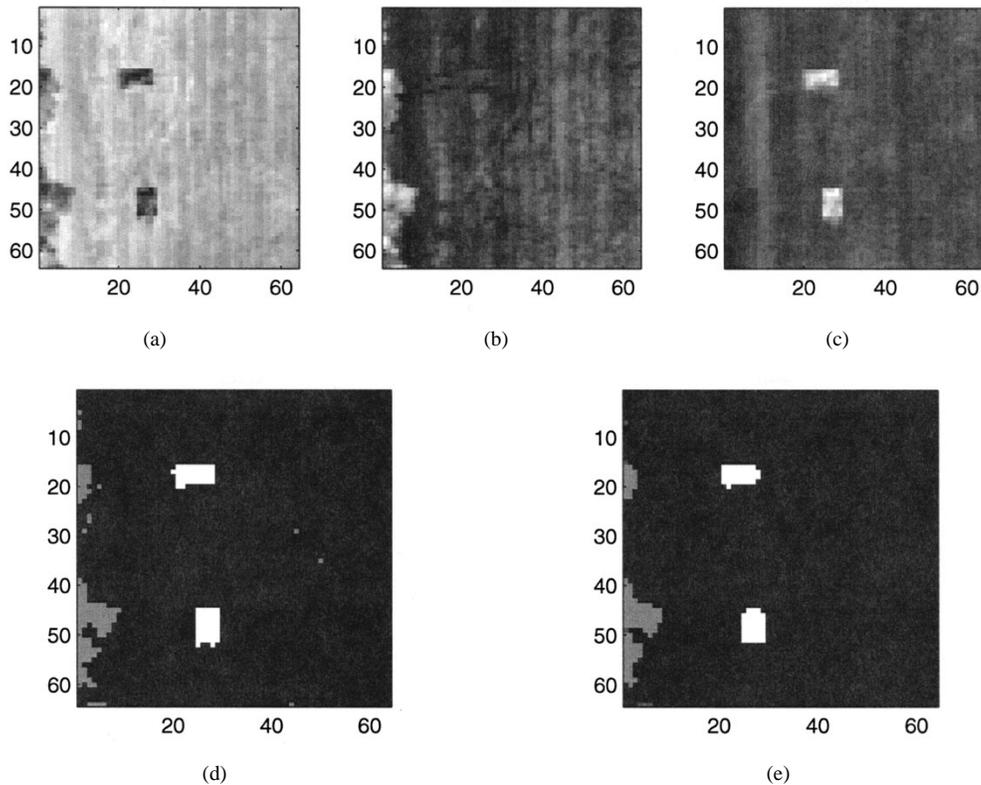

Fig. 12.   Results of three-class segmentation of HYDICE image: (a) score image corresponding to corner 1, (b) score image corresponding to corner 2, (c) score image corresponding to corner 5, (d) segmented image, and (e) segmented image after 3×3 median filtering.
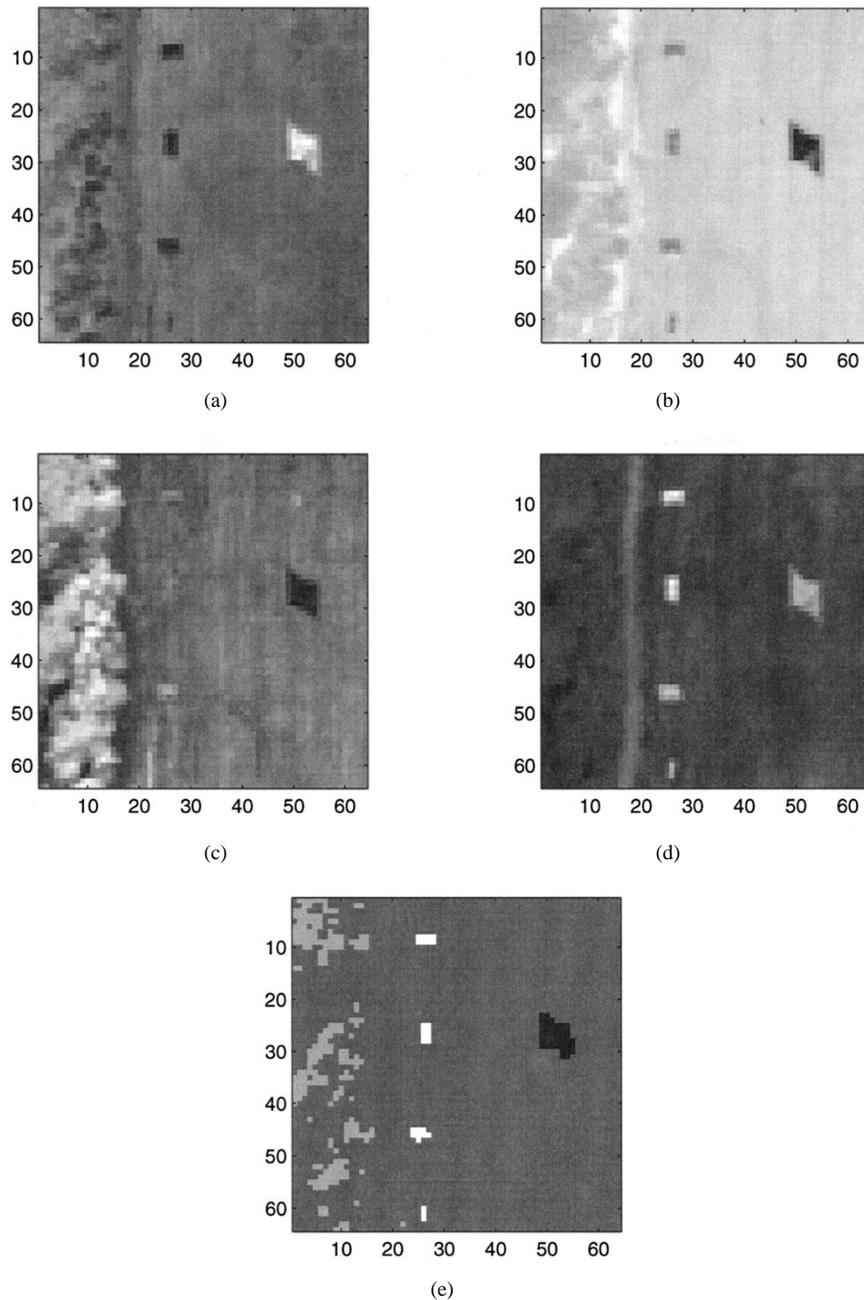
Fig. 13.   Results of 4-class segmentation of HYDICE image: (a) score image corresponding to corner 1, (b) score image corresponding to corner 3, (c) score image corresponding to corner 4, (d) score image corresponding to corner 19, and (e) segmented image.

TABLE III
RMS ERRORS FOR TWO-ENDMEMBER UNMIXING SIMULATION
AT FOUR SNR VALUES AND FOUR SPECTRAL ANGLES

SNR	Spectral Angle			
	0.5698	0.7786	0.9394	0.9901
5	0.1642	0.2259	0.2642	0.2768
10	0.0824	0.1309	0.2137	0.2440
20	0.0415	0.0662	0.1379	0.2420
40	0.0210	0.0353	0.0890	0.1879

TABLE IV
RMS ERRORS FOR THREE-ENDMEMBER UNMIXING SIMULATION
AT FOUR SNR VALUES AND FOUR SPECTRAL ANGLES

SNR	Spectral Angle			
	0.5698	0.7786	0.9394	0.9901
5	0.1422	0.1703	0.2000	0.2157
10	0.0782	0.1302	0.1656	0.1906
20	0.0474	0.0960	0.1448	0.1767
40	0.0289	0.0572	0.1444	0.1626

with the vehicles. The final segmentation is obtained as before by selecting the largest score for each pixel and is again enhanced by 3×3 median filtering. We can compare the segmented images in Fig. 12(d) and (e) with the broadband (total radiance) image in Fig. 10(a). The method segments the objects in spite of the strong illumination nonuniformity and the mismatch between the corner spectra and the true spectra of the objects in the scene.

To segment the second image, we have assumed four classes: grass, trees, vehicles, and object. We show the matched-filtered score and segmented images in Fig. 13. Again they correlate strongly with the object classes. In this case, we have chosen corners 1, 3, 4, and 19 out of a total of 29 by the automated method mentioned earlier, rather than attempting to make the selection graphically. No median filtering is done in this case. Because some of the objects in the scene occupy only a few pixels, even a 3×3 filter would have annihilated them.

V. CONCLUSIONS

The use of CCA for multispectral and hyperspectral image analysis is a new application. The idea of using endmembers derived from the data for classification and unmixing with remote imaging spectrometry has been considered before [11]–[13], but there have been few methods for deriving the component spectra. In contrast, the literature abounds with methods that rely on external information sources, such as spectral libraries [21], [22]. Unfortunately, such information is often not available in practice, so data analysts have resorted to clustering methods to estimate the component spectra. CCA offers an alternative approach to solve this problem. Unlike clustering, or the MVT, it does not require initialization or iterations. All optimizations are done by searching, so there are no problems with local minima. In addition, it takes full advantage of the spectral correlations via the SVD and produces physically meaningful components using only the information contained in the data.

Geometrically, CCA is very similar to Craig's DFP transform [11]. Although the concept of the MVT essentially accomplishes the same goals as CCA, it is different in essence. While the MVT assumes that endmembers are estimated by the vertices of the smallest simplex that embraces the data, our method looks for vertices that are as far away from each other as possible, subject to the positivity constraint. Therefore, we seek to maximize the difference between the estimated component spectra. Another significant difference between the two approaches is that the convex cone is a unique function of the eigenvectors of the correlation matrix, while the MVT is one of a possibly very large set of solutions obtained by iterative optimization.

As with any new approach, there are some unresolved issues that may present challenges over time. The first is the classic problem of determining the number of components in the scene. In this paper, we have used examples in which the number of classes is known *a priori*. In a fully automated system, we must estimate the number of components from the scene data. Reference [4] contains a long list of methods that are used to estimate the intrinsic dimensionality of multivariate data. A simple approach is to look at the plot of the ordered eigenvalues from the SVD of the sample correlation matrix [27]. Information theoretic measures, such as an information criterion (AIC) or minimum description length (MDL) [21], can also be used, although they tend to overestimate the true dimensionality. In some cases, the number of components is equal (or close) to the intrinsic dimensionality, but that is not always the case in images with spatially extended objects. Very often, objects in a multispectral or hyperspectral image require two or more spectral dimensions, and thus, they increase the intrinsic dimensionality of the data.

A second issue, related to the first, is that of using the eigenvectors of the spectral correlation matrix. In imaging spectrometry data, the information is not necessarily associated with the directions of maximum variance, so a portion may be discarded along with the discarded eigenvectors. These two issues may be addressed simultaneously by using an alternative orthogonal transformation, such as MNF [5] or projection pursuit [28], to reduce the dimensionality of the data while preserving more of the information. This is the subject of our current research.

ACKNOWLEDGMENT

REFERENCES

[1] G. Vane and A. F. H. Goetz, "Terrestrial imaging spectrometry," *Remote Sens. Environ.,* vol. 24, pp. 1–29, 1988.

[2] J. B. Breckenridge, "Evolution of imaging spectrometry: Past, present, and future," *Proc. SPIE,* vol. 2819, pp. 2–6, 1996.

[3] S. K. Rogers, J. M. Colombi, C. E. Martin, J. C. Gainey, K. H. Fielding, T. J. Burns, D. W. Ruck, M. Kabrisky, and M. Oxley, "Neural networks for automatic target recognition," *Neural Networks,* vol. 8, no. 7, pp. 1153–1184, 1995.

[4] E. Malinowski, *Factor Analysis in Chemistry,* 2nd Ed. New York: Wiley, 1991.

[5] A. A. Green, M. Berman, P. Switzer, and M. D. Craig, "A transformation for ordering multispectral data in terms of image quality with implications for noise removal," *IEEE Trans. Geosci. Remote Sensing,* vol. 26, pp. 65–74, Jan. 1988.

[6] M. L. Mavrovouniotis, A. M. Harper, and A. Ifarraguerri, "Classification of pyrolysis mass spectra of biological agents using convex cones," *J. Chemometrics,* vol. 8, pp. 305–333, 1994.

[7] ——, "A method for extracting patterns from pyrolysis mass spectra," in *Computer Assisted Analytical Spectroscopy.* Chichester, U.K.: Wiley, 1996, pp. 189–240.

[8] W. H. Lawton and E. A. Silvestre, "Self modeling curve resolution," *Technometrics,* vol. 13, no. 3, pp. 617–633, 1971.

[9] J. C. Hamilton and P. J. Gemperline, "Mixture analysis using factor analysis II: Self-modeling curve resolution," *J. Chemometrics,* vol. 4, pp. 1–13, 1990.

[10] O. S. Borgen and B. R. Kowalski, "An extension of the multivariate component-resolution method to three components," *Anal. Chimica Acta,* vol. 174, pp. 1–26, 1985.

[11] M. D. Craig, "Minimum-volume transforms for remotely sensed data," *IEEE Trans. Geosci. Remote Sensing,* vol. 32, pp. 542–552, May 1994.

[12] J. W. Boardman, "Automated spectral unmixing of AVIRIS data using convex geometry concepts," in *Proc. Summaries, 4th JPL Airborne Geosci. Workshop,* JPL Publ. 93-26, 1993, vol. 1, pp. 11–14.

[13] ——, "Geometric mixture analysis of imaging spectrometry data," in *Proc. Int. Geosci. Remote Sensing Symp.,* 1994, vol. 4, pp. 2369–2371.

[14] V. C. Klema and A. J. Laub, "The singular value decomposition: Its computation and some applications," *IEEE Trans. Automat. Control,* vol. 25, pp. 164–176, 1980.

[15] S. N. Deming, J. A. Palasota, and J. M. Nocerino, "The geometry of multivariate object preprocessing," *J. Chemometrics,* vol. 7, pp. 393–425, 1993.

[16] M. L. Mavrovouniotis, A. M. Harper, and A. Ifarraguerri, "Convex-cone analysis of the time profiles of pyrolysis mass spectra of biological agents," U.S. Army Edgewood Res., Develop. Eng. Center, Tech. Rep. ERDEC-CR-130, July 1994.

[17] S. S. Shen, "Multiband sensor system design tradeoffs and their effects on remote sensing and exploitation," *Proc. SPIE,* vol. 3118, pp. 296–307, 1997.

[18] C.-I. Chang, Q. Du, T.-L. Sun, and M. L. G. Althouse, "A joint band prioritization and band decorrelation approach to band selection for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sensing,* to be published.

[19] I. S. Reed and X. Yu, "Adaptive multiple-band CFAR detection of an optical pattern with unknown spectral distribution," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 38, pp. 1760–1770, Oct. 1990.

[20] R. H. Myers, *Classical and Modern Regression Analysis with Applications,* 2nd ed. Belmont, CA: Duxbury, 1990.

[21] D. H. Kil and F. B. Shin, *Pattern Recognition and Prediction with Applications to Signal Characterization.* Woodbury, NY: AIP, 1996.

[22] J. Harsanyi and C.-I. Chang, "Hyperspectral image classification and dimensionality reduction: an orthogonal subspace projection approach," *IEEE Trans. Geosci. Remote Sensing,* vol. 32, pp. 779–785, July 1994.

[23] R. C. Gonzalez and R. E. Woods, *Digital Image Processing.* Reading, MA: Addison-Wesley, 1992.

[24] W. S. Aldrich, M. E. Kappus, and R. G. Resmini, "HYDICE post-flight data processing," in *Proc. SPIE,* 1996, vol. 2758, pp. 354–363.

[25] R. W. Basedow, W. S. Aldrich, J. E. Colwell, and W. D. Kinder, "HYDICE system performance update," in *Proc. SPIE,* 1996, vol. 2821, pp. 76–84.

[26] R. E. Crippen, "The regression intersection method of adjusting image data for band ratioing," *Int. J. Remote Sensing,* vol. 8, no. 2, pp. 137–155, 1987.

[27] J. W. Boardman and F. A. Kruse, "Automated spectral analysis: A geological example using AVIRIS data, North Grapevine Mountains, NV," in *Proc. 10th Thematic Conf. Geol. Remote Sensing,* 1994, pp. 407–418.

[28] J. H. Friedman, "Exploratory projection pursuit," *J. Amer. Stat. Assoc.,* vol. 82, no. 397, 1987.

Agustin Ifarraguerri (S'95) received the M.S. degree in bioengineering from The University of Michigan, Ann Arbor. He is currently pursuing the Ph.D. degree in electrical engineering at the University of Maryland, Baltimore County.

He has been with the United States Army Edgewood Research, Development and Engineering Center, Aberdeen Proving Ground, MD, for over seven years, where he has contributed to the development of several systems for the detection of chemical and biological agents and is currently participating in the development of an imaging spectrometer for standoff detection of battlefield chemical agents.

Mr. Ifarraguerri is a Student Member of SPIE and Phi Kappa Phi.

Chein-I Chang (S'81–M'87–SM'92), for a photograph and biography, see p. 268 of the January 1998 issue of this TRANSACTIONS.