L1-Endmembers: A Robust Endmember Detection and Spectral Unmixing Algorithm

Alina Zare and Paul Gader

Department of Computer & Information Science & Engineering, University of Florida, Gainesville, FL 32611

ABSTRACT

A hyperspectral endmember detection and spectral unmixing algorithm based on an l_1 norm factorization of the input hyperspectral data is developed and compared to a method based on l_2 norm factorization. Both algorithms, the L1-Endmembers algorithm based on the l_1 norm and the SPICE algorithm based on the l_2 norm, simultaneously and autonomously estimate endmember spectra, abundance values and the number of endmembers needed for a hyperspectral image. The l_1 norm factorization of the hyperspectral data is approximated through the use of the Huber M-estimator. Results showing the stability of the L1-Endmembers algorithm in terms of the number of endmembers estimated with noise and outliers are presented. Results indicate that the proposed algorithm is more consistent in estimating the correct number of endmembers over SPICE. However, when both algorithms determine the correct number of endmembers, SPICE results provide a better estimate of endmembers and a lower variance of endmember estimates over many runs with random initialization.

Keywords: Hyperspectral, Endmember Detection, Spectral Unmixing, Huber M-Estimator, Matrix Factorization

1. INTRODUCTION

Spectral unmixing decomposes mixed hyperspectral pixels into their respective *endmembers* and *abundances*. Endmembers are the spectral signatures of the "pure" or constituent materials in a hyperspectral data set. Abundances are the proportions of the endmembers in each pixel in a hyperspectral image. In order to perform spectral unmixing, a mixing model is needed. The standard mixing model is the *convex geometry model* (also known as the *linear mixing model*) which assumes that every pixel is a convex combination of endmembers in the scene.[1–3] Under the convex geometry model, endmembers are the spectra found at the corners of a convex region enclosing all the spectra in a hyperspectral scene.

$$\mathbf{x}_i = \sum_{k=1}^{M} p_{ik}\mathbf{e}_k + \epsilon_i \quad i = 1, \ldots, N \tag{1}$$

where N is the number of pixels in the image, M is the number of endmembers, ϵ_i is an error term, p_{ik} is the proportion (abundance) of endmember k in pixel i, and $\mathbf{e}_k$ is the k^{th} endmember. The dimensionality of each data point, $\mathbf{x}_i$, and each endmember, $\mathbf{e}_k$, is equal to the number of bands in the hyperspectral data, D. The proportions of this model satisfy the constraints in Equation 2,

$$p_{ik} \geq 0 \quad \forall k = 1, \ldots, M; \quad \sum_{k=1}^{M} p_{ik} = 1. \tag{2}$$

Several endmember detection and spectral unmixing algorithms have been developed in the literature. However, the majority of these methods do not provide an autonomous way to estimate the number of

Send correspondence to A. Zare
A. Zare: E-mail: azare@cise.ufl.edu
P. Gader: E-mail: pgader@cise.ufl.edu

Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XVI, edited by Sylvia S. Shen, Paul E. Lewis, Proc. of SPIE Vol. 7695, 76951L · © 2010 SPIE
CCC code: 0277-786X/10/$18 · doi: 10.1117/12.851065

endmembers and, thus, require the number of endmembers in advance. These methods include those based on Non-negative Matrix Factorization,[4–7] based on Indepenent Components Analysis,[8,9] and others.[10–15]

Methods to estimate the number of endmembers from a data set have been developed as well. These methods include Virtual Dimensionality (VD), Transformed Gerschogorin Disk (TGD), the Noise-Adjusted TGD, and the Partitioned Noise-Adjusted Principal Components Analysis (PNAPCA) methods.[16–18] The VD method estimates the number of endmembers using the eigenvalues of the covariance and correlation matrices of the hyperspectral data set. The number of endmembers is set to the number of eigenvalues from the covariance and correlation matrices that differ based on some computed threshold. Due to the variances used when computing the thresholds, the VD method can be sensitive to noise in the data. The PNAPCA method relies on the use of the Maximum Noise Fraction (MNF) algorithm.[19] MNF simultaneously diagonalizes the data covariance matrix and whitens the noise covariance matrix for a data set. This requires an estimate of the noise covariance matrix in advance. PNAPCA partitions the noise-adjusted covariance matrix found by MNF and compares eigenvalues from the partitions to estimate the number of endmembers. However, an accurate estimate of the noise covariance matrix can be difficult to provide. The TGD method for estimating the number of endmembers uses the Gerschogorin disk theorem. This theorem provides a method of estimating the locations of eigenvalues for a matrix.[20] The TGD method defines a transformation that aims to cause the Gerschgorin disks associated with noise to have small radii and be located far from the disks associated with signal in the data. The Noise-Adjusted TGD applies this method to the noise-adjusted covariance matrix found using MNF. The TGD methods are also sensitive to noise.

The presented algorithm, the L1-Endmembers algorithm, provides a noise and outlier robust method to simultaneously estimate endmember spectra, proportion values and the number of endmembers using an l_1 norm error term.

In their respective papers, both Ke and Kanade[21] and Mangasarian and Musicant[22] use the l_1 norm and the Huber M-Estimator for matrix factorization. However, in hyperspectral endmember detection, the constraints on the abundances shown in Equation 2 are required to provide physically meaningful results. These constraints are not enforced in the matrix factorization methods presented by Ke and Kanade[21] and Mangasarian and Musicant.[22] In addition, the L1-Endmembers algorithm provides a sparsity-promoting term used to simultaneously estimate the number of endmembers needed for the image. This sparsity-promoting term is not used in the matrix factorization methods.

2. ENDMEMBER DETECTION USING THE HUBER M-ESTIMATOR

The L1-Endmembers algorithm estimates endmembers, proportions and the number of endmembers by iteratively optimizing an objective function containing three terms. The first term of the objective function computes the error between the input data points and their reconstruction using the estimated endmembers and proportion values.

$$E_{l_1}(\mathbf{E},\mathbf{P}) = \|\mathbf{X}-\mathbf{EP}\|_{l_1} \tag{3}$$

where $\|\mathbf{Z}\|_{l_1} = \sum_i \sum_j |z_{ij}|$ where z_{ij} is the element in the i^{th} row and j^{th} column of a matrix $\mathbf{Z}$, $\mathbf{E}$ is matrix where the k^{th} column contains the endmember spectra, $\mathbf{e}_k$, $\mathbf{X}$ is the matrix of data points where the i^{th} column contains the data point $\mathbf{x}_i$, and $\mathbf{P}$ is the matrix of abundances where the i^{th} column corresponds to the proportions associated with the i^{th} data point and each row corresponds to the proportions associated with the corresponding endmember. Therefore, this computes the l_1 error between the data, $\mathbf{X}$, and its estimate using endmembers and abundances, $\mathbf{EP}$. In contrast, the l_2 norm is used in the SPICE and ICE algorithms.[23,24] The use of the l_1 norm in this term provides resiliance against noise and outliers in the data. The robustness of the l_1 norm when compared to the l_2 is established[21,25] and is illustrated in Figure 1(a). In the figure, the y-axis corresponds to the penalty for a given error using either the l_1 or l_2 norm. As you can see, the l_1 norm has a lower penalty value

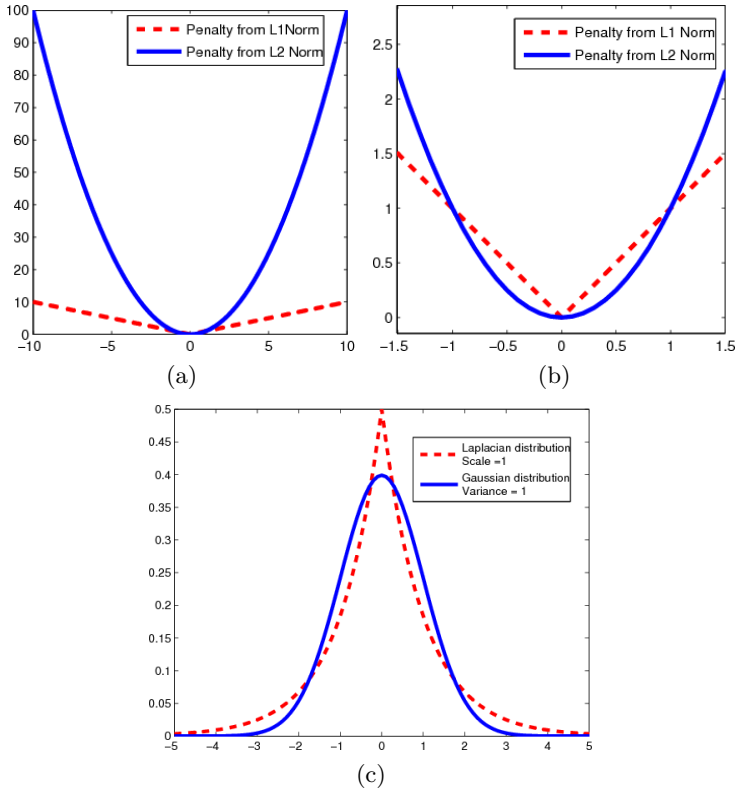


Figure 1. (a) Plot of the penalty values for the L_1 and L_2 norms for the interval of [-10, 10] and (b) for the interval of [-1.5, 1.5]. (c) Plot showing the Laplacian distribution with a scale parameter of 1 and Gaussian distribution with variance of 1.

for large errors providing a higher tolerance for outliers when compared to the l_2 norm. The l_2 norm prefers a series of many small errors instead of a small number of large errors.

The Huber M-Estimator cost function approximates the l_1 norm and is shown in Equation 4.

$$\rho(t) = \begin{cases} \frac{1}{2}t^2 & \text{if } |t| \leq \gamma \\ \gamma|t| - \frac{1}{2}\gamma^2 & \text{if } |t| > \gamma \end{cases} . \tag{4}$$

where γ is a positive constant parameter. As $\gamma \to 0^+$, the Huber M-estimator becomes the l_1 norm.[26] Figure 2 shows a comparison of the Huber M-Estimator for many γ values in comparison to the l_1 norm.

Using the Huber M-estimator, the first term of our l_1-Endmembers objective function becomes

$$E_H(\mathbf{E}, \mathbf{P}) = \sum_{i=1}^{N} \sum_{j=1}^{D} \rho\left((\mathbf{x}_i - \mathbf{E}\mathbf{p}_i)_j \right) \tag{5}$$

where $(\mathbf{x}_i - \mathbf{E}\mathbf{p}_i)_j$ is the j^{th} element in the vector $\mathbf{x}_i - \mathbf{E}\mathbf{p}_i$. Each term in this cost function, as shown in the paper by Mangasarian and Musicant,[21,22] can be rewritten as the quadratic programming problem shown in Equation 6 when solving for the proportion values.

$$\begin{aligned} \min_{\mathbf{p}\in R^M, \mathbf{z}\in R^D, \mathbf{t}\in R^D} & \frac{1}{2}\|\mathbf{z}\|_{L_2}^2 + \gamma \mathbf{1}^T \mathbf{t} \\ s.t. & -\mathbf{t} \leq \mathbf{E}\mathbf{p}_i - \mathbf{x}_i - \mathbf{z} \leq \mathbf{t} \end{aligned} \tag{6}$$

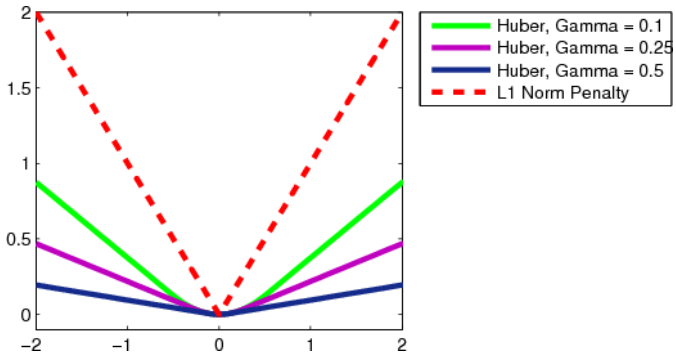


Figure 2. Comparison of the Huber M-Estimator with Gamma Values of 0.5, 0.25 and 0.1 to the L1-Norm.

where $\mathbf{x}_j$ is the i^{th} data point, $\mathbf{p}_i$ is the vector of proportion values for the i^{th} data point, D is the data dimensionality, M is the number of endmembers, $\mathbf{1}$ is a vector of ones, and the constraint is such that each element of the vector $\mathbf{E}\mathbf{p}_i - \mathbf{x}_i - \mathbf{z}$ is bounded by the corresponding element in the vector $\mathbf{t}$. A similar form is used when solving for endmembers.

The second term of our objective function promotes endmembers that provide a tight fit around the data.

$$E_V(\mathbf{E}) = \frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{M}(\mathbf{e}_i - \mathbf{e}_j)^T(\mathbf{e}_i - \mathbf{e}_j) \tag{7}$$

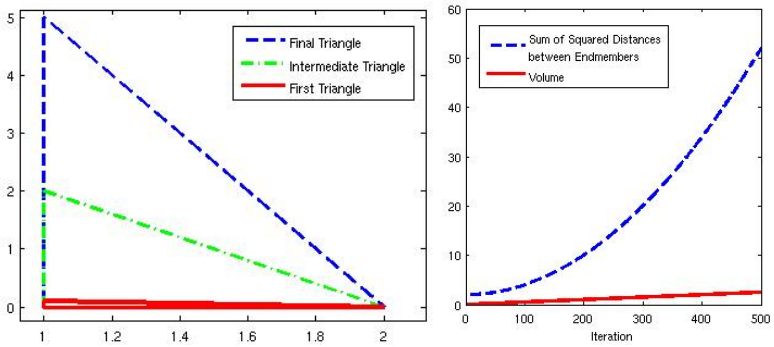
where M is the number of endmembers. Many hyerspectral endmember detection algorithms rely on the simplex volume rather than this sum of squared distances term to promote a tight fit around the data set. The simplex volume often is computed using the determinant of the endmember matrix[12] which may require dimensionality reduction so that the determinant can be computed. An advantage of using the sum of distances is that dimensionality reduction is not required. However, the weight added by the volume and the sum of squared distances term differs. Figure 3 shows a comparison of the sum of squared distances term and the value computed using a volume term for a triangle increasing in size. Specifically, the original triangle is composed of the endmembers located at [(1,0), (1, 0.1), (2,0)]. Then, one endmember (1,0.1) is moved away at a constant rate (0.01 step size) until reaching the location (1,5) in 500 iterations. Figure 3 shows a comparison of the volume and sum of squared distances term as the endmember is incrementally moved farther from the remaining two endmembers. The volume term increases linearly as the endmember moves away whereas the sum of squared distances term increases quadratically.

The final term of our objective function uses the sparsity promoting properties of the Laplacian distribution to determine the number of endmembers. This term determines the number of endmembers by driving the proportions of unneeded endmembers to zero. Endmembers whose proportion values are driven to zero can be removed without any effect on the value of the cost function in Equation 5.

$$E_S(\mathbf{P}) = \sum_{k=1}^{M}\lambda_k\sum_{i=1}^{N}|p_{ik}| = \sum_{k=1}^{M}\lambda_k\sum_{i=1}^{N}p_{ik} \tag{8}$$

where $\lambda_k = \frac{\Lambda}{\sum_{i=1}^{N} p_{ik}}$ using the proportion values from the previous iteration where Λ is a parameter value set to control the degree of sparsity imposed on the proportion values. The second equality in Equation 8 comes from the non-negativity constraints on the proportion values in Equation 2. Both Equations 7 and 8 are also used in the SPICE objective function.[23]

The complete objective function is $E = E_H + \beta E_V + E_S$ where β is a coefficient determining the weight of the corresponding term. The Λ value used in E_S of the objective function controls the degree of sparsity used to determine the number of endmembers. In other words, Λ controls the scale value



(a) (b)

Figure 3. (a) Illustration of the simplex with increasing size used to generate the results in (b). (b)Comparison of the simplex volume and sum of square distances between endmembers as one endmember is moved away from the other two endmembers at a constant rate.

in the Laplacian prior on the abundance values. A larger value causes the Laplacian to have a sharper peak at zero.

The L1-Endmembers algorithm iteratively updates the proportions, endmembers, and number of endmembers. During an proportion update, the quadratic programming problem is shown in Equation 9.

$$\begin{aligned} \min_{\mathbf{p}\in R^M, \mathbf{z}\in R^D, \mathbf{t}\in R^D} & \frac{1}{2}\|\mathbf{z}\|_{L_2}^2 + \gamma \mathbf{1}^T\mathbf{t} + E_S \\ s.t. & -\mathbf{t} \leq \mathbf{E}\mathbf{p}_i - \mathbf{x}_i - \mathbf{z} \leq \mathbf{t} \\ & p_{ik} \geq 0 \quad \forall i,k \\ & \sum_{k=1}^{M} p_{ik} = 1. \end{aligned} \tag{9}$$

Similarly, the quadratic programming problem for solving for the endmembers during an endmember update is shown in Equation 10.

$$\begin{aligned} \min_{\mathbf{e}\in R^M, \mathbf{z}\in R^N, \mathbf{t}\in R^N} & \frac{1}{2}\|\mathbf{z}\|_{L_2}^2 + \gamma \mathbf{1}^T\mathbf{t} + \beta E_V \\ s.t. & -\mathbf{t} \leq \mathbf{P}\mathbf{e}^j - \mathbf{x}^j - \mathbf{z} \leq \mathbf{t} \end{aligned} \tag{10}$$

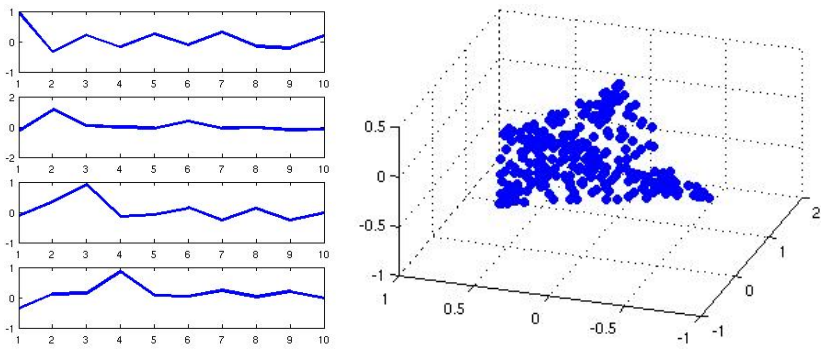
where $\mathbf{x}^j$ is the vector of values from the j^{th} band across all data points, $\mathbf{e}^j$ is the vector containing the value from the j^{th} band over all endmembers, N is the number of data points, M is the number of endmembers, and $\mathbf{1}$ is a vector of ones.

After an iteration of updating endmembers and proportions, the proportions associated with each endmember can be examined. If the maximum proportion value falls below some threshold, the endmember can be pruned. During experimentation, the number of endmembers was found to be insensitive to this threshold value and, therefore, it can be set to a very small value.

3. EXPERIMENTAL RESULTS

The first set of experimental results presented are on simulated data in which data points were generated from four ten-dimensional endmembers as shown in Figure 4(a). The data was simulated by generating proportion values for 300 data points from a uniform distribution. This data, following dimensionality

reduction using principal components analysis (for visualization), is shown in Figure 4(b). All experiments were conducted on the full ten-dimensional data. After generating the data, zero-mean Gaussian random noise (with a variance of 0.1) was added to each data point and each dimensionality. Therefore, 300x10 zero-mean Gaussian-distributed random values were generated and added to the simulated data.



(a) (b)

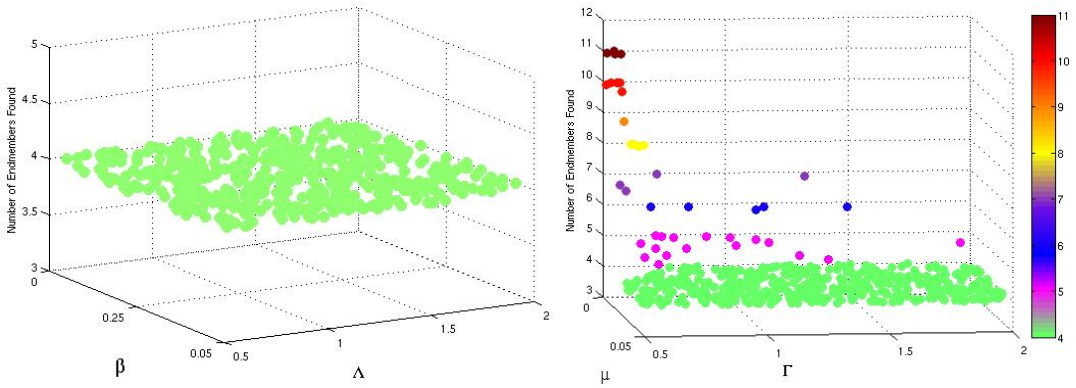
Figure 4. (a) Endmembers used to create simulated data. (b) Ten-dimensional simulated data prior to the addition of Gaussian noise following principal components analysis dimensionality reduction to three dimensions.

3.1 Sensitivity to Parameter Value Selection

In order to evaluate the sensitivity to parameter value settings, the parameter values β and Λ for the L1-Endmembers algorithm and the parameter values μ and Γ (using the notation from the SPICE paper[23]) for the SPICE algorithm were varied. The L1-Endmembers and SPICE algorithm differ in that the SPICE algorithm uses an l_2 norm and the L1-Endmembers algorithm uses the Huber M-estimator in the objective function for computing the difference between the data and its estimate using endmembers and proportion values. The parameter values for β and μ in L1-Endmembers and SPICE, respectively, were randomly sampled from a uniform distribution over the interval $[0, 0.05]$. The parameter values for Λ and Γ in L1-Endmembers and SPICE, respectively, were randomly sampled from a uniform distribution over the interval $[0.5, 2]$. The γ parameter for the L1-Endmembers algorithm was held constant at 1. The initial number of endmembers for both algorithms was set to 20. The parameters that were sampled for the SPICE and L1-Endmembers algorithm controlled the volume-related term and the sparsity-promoting terms in both algorithms. The plots in Figure 5 show the resulting number of endmembers found when run using the randomly sampled parameter values for the L1-Endmembers and the SPICE algorithm. As shown in the figures, the L1-Endmembers algorithm consistently and correctly estimated four endmembers for the simulated data over the entire range of parameter values. In contrast, the SPICE algorithm had several incorrectly estimated number of endmembers. Specifically, more runs of this experiment resulted in an incorrect esimate number of endmembers by SPICE as the μ parameter approached 0 and the Λ parameter approached 0.5. For all runs in this experiment, random initialization was employed for both algorithms. Table 1 shows these results in tabular form.

Table 1. Number of endmembers found using the L1-Endmembers and SPICE with randomly sampled parameter values. This table lists the number of times (out of 500) each number of estimated endmembers was returned by each algorithm.

	Number of Estimated Endmembers								
Algorithm	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
SPICE	460	17	5	4	4	1	5	4	0
L1-Endmembers	500	0	0	0	0	0	0	0	0



(a) (b)

Figure 5. (a) Scatter plot showing the randomly sampled parameter values (x- and y- axis) and the resulting number of endmembers found using the L1-Endmembers algorithm on the simulated data with Gaussian random noise (z-axis) (b) Scatter plot showing the randomly sampled parameter values (x- and y- axis) and the resulting number of endmembers found using the SPICE algorithm on the simulated data with Gaussian random noise (z-axis)

3.2 Sensitivity to Initialization

In order to examine the sensitity to initialization, a set of parameter values were selected using the results from the previous experiment. Given the results from the previous experiment, parameter values were chosen for both the SPICE and L1-Endmembers algorithm using the median parameter value that estimated the correct number of endmembers in the previous experiment. Specifically, β and μ values for the two algorithms were set to 0.05 and the Λ and Γ parameters were set to 1.24. Furthermore, γ for the L1-Endmembers algorithm was set to 0.1. Gaussian random noise was added to the simulated data five times for five experiments. In each of the five experiments, each algorithm was run three times with a random initialization in order to investigate stability of the algorithm with respect to random initialization. Table 2 shows the results of this experiment in terms of the number of endmembers estimated by L1-Endmembers, SPICE and the HySime algorithm.[27] In this case, both the SPICE and L1-Endmembers algorithms correctly estimated the number of endmembers. However, the SPICE algorithm provided a lower variance of the endmembers found across random initializations of the algorithm as shown in Table 3. The SPICE algorithm also produced lower error values in terms of the euclidean distance between the true endmembers used to generate the data and the estimated endmembers found by the algorithm.

3.3 Sensitivity to Outliers

The previous experiment was repeated using the L1-Endmembers, SPICE and HySime algorithm for data sets with outliers and higher levels of noise. The same parameters found using experiment 1 with lower noise levels and without any outliers were used for this experiment. Outliers were added to the data set by randomly sampling three outlying data points from a uniform distribution. For the first set of results, zero-mean Gaussian random noise (with variance 0.1) was added to the data. Table 4 lists the resulting number of endmembers for the three algorithms with random intialization. For the second set of results in this experiment, zero-mean Gaussian random noise (with variance 0.15) was added to the data resulting in a higher level of added noise. Table 5 lists the resulting number of endmembers for the three algorithms with random intialization.

4. CONCLUSIONS AND FUTURE WORK

An endmember detection and spectral unmixing algorithm using the Huber M-Estimator is presented. In the results on simulated data shown above, the presented L1-Endmembers algorithm demonstrated

Table 2. Number of endmembers found using the L1-Endmembers, SPICE and HySime algorithms. This table lists the number of endmembers found on simulated data with zero-mean Gaussian noise (variance of 0.1). Each column corresponds to a different simulated data set with Gaussian-distributed noise. Each repetition of the experiment was with an identical data set and random initialization of the L1-Endmembers algorithm.

		Experiment				
Algorithm	**Run**	**1**	**2**	**3**	**4**	**5**
L1-Endmembers	1	4	4	4	4	4
	2	4	4	4	4	4
	3	4	4	4	4	4
SPICE	1	4	4	4	4	4
	2	4	4	4	4	4
	3	4	4	4	4	4
HySime	1	3	3	3	3	3
	2	3	3	3	3	3
	3	3	3	3	3	3

Table 3. Mean Variance and Mean Euclidean Distance between Estimated Endmembers and True Endmembers for 10-Dimensional Simulated Data Set using the L1-Endmembers and SPICE Algorithms Across Three Runs of Each Algorithm with Random Initialization.

		Experiment				
Algorithm	**Results**	**1**	**2**	**3**	**4**	**5**
L1-Endmembers	Mean Variance	0.23	0.13	7.0×10^{-5}	0.21	0.25
SPICE	Mean Variance	2.6×10^{-7}	1.1×10^{-7}	8.6×10^{-8}	2.5×10^{-8}	1.6×10^{-7}
L1-Endmembers	Mean Error	3.15	1.72	2.12	1.98	2.71
SPICE	Mean Error	0.68	0.71	0.68	0.63	0.73

insensitivity to parameter values over the SPICE algorithm which is based on using an l_2 norm rather than the l_1 norm estimate used by the proposed algorithm. With increasing levels of noise and the addition of outliers, the L1-Endmembers algorithm consistently estimated the correct number of endmembers using a single parameter set. However, the SPICE algorithm, with appropriately chosen parameter values for the noise level, demonstrated lower error levels and variance in the estimated endmembers across several random initializations of the algorithm. In other words, when SPICE estimates the correct number of endmembers, it produces better endmembers for the data set. Further investigation into parameter value settings for the L1-Endmembers algorithm in terms of the variance and endmember error will be conducted.

ACKNOWLEDGMENTS

Research was supported by NSF program Optimized Multi-algorithm Systems for Detecting Explosive Objects Using Robust Clustering and Choquet Integration (CBET-0730484). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of NSF. The U. S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

REFERENCES

[1] Keshava, N. and Mustard, J. F., "Spectral unmixing," *IEEE Signal Processing Magazine* **19**, 44–57 (2002).

[2] Nascimento, J. M. P. and Bioucas-Dias, J. M., "Does independent component analysis play a role in unmixing hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing* **43**, 175–187 (Jan. 2005).

Table 4. Number of endmembers found using the L1-Endmembers, SPICE and HySime algorithms. This table lists the number of endmembers found on simulated data with zero-mean Gaussian noise (variance of 0.1) and three outliers added to the data. Each column corresponds to a different simulated data set with Gaussian-distributed noise. Each repetition of the experiment was with an identical data set and random initialization of the L1-Endmembers algorithm.

		Experiment									
Algorithm	**Run**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
L1-Endmembers	1	4	4	4	4	4	4	4	4	4	4
	2	4	4	4	5	4	4	4	4	4	4
	3	4	4	4	5	4	4	4	4	4	4
SPICE	1	6	5	6	5	5	5	5	6	5	5
	2	5	5	5	5	5	5	7	5	5	5
	3	5	5	5	5	5	5	6	5	6	6
HySime	1	4	4	3	3	3	3	4	3	3	3
	2	4	4	3	3	3	3	4	3	3	3
	3	4	4	3	3	3	3	4	3	3	3

Table 5. Number of endmembers found using the L1-Endmembers, SPICE and HySime algorithms. This table lists the number of endmembers found on simulated data with zero-mean Gaussian noise (variance of 0.15) and three outliers added to the data. Each column corresponds to a different simulated data set with Gaussian-distributed noise. Each repetition of the experiment was with an identical data set and random initialization of the L1-Endmembers algorithm.

		Experiment									
Algorithm	**Run**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
L1-Endmembers	1	4	4	4	4	5	4	4	4	5	4
	2	4	4	4	4	5	4	4	4	4	4
	3	4	4	4	4	5	4	4	4	5	4
SPICE	1	9	10	9	11	10	13	11	11	11	10
	2	10	11	9	11	11	10	11	10	10	11
	3	9	9	11	10	11	10	10	9	12	10
HySime	1	3	3	3	3	2	3	3	3	3	3
	2	3	3	3	3	2	3	3	3	3	3
	3	3	3	3	3	2	3	3	3	3	3

[3] Manolakis, D., Marden, D., and Shaw, G. A., "Hyperspectral image processing for automatic target detection applications," *Lincoln Laboratory Journal* **14**(1), 79–116 (2003).

[4] Lee, D. and Seung, H., "Algorithms for non-negative matrix factorization," in [*Advances in Neural Information Processing Systems 13*], 556–562 (2000).

[5] Miao, L. and Qi, H., "Endmember extraction from highly mixed data using minimum volume constrained nonnegative matrix factorization," *IEEE Transactions on Geoscience and Remote Sensing* **45**, 765–777 (Mar. 2007).

[6] Pauca, V. P., Piper, J., and Plemmons, R. J., "Nonnegative matrix factorization for spectral data analysis," *Linear Algebra Applications* **416**, 321–331 (Jul. 2005).

[7] Jia, S. and Qian, Y., "Constrained nonnegative matrix factorization for hyperspectral unmixing," *IEEE Transactions on Geoscience and Remote Sensing* **47**, 161–173 (Jan. 2009).

[8] Tu, T.-M., "Unsupervised signature extraction and separation in hyperspectral images: A noise-adjusted fast independent components analysis approach," *Optical Engineering* **39**(4), 897–906 (2000).

[9] Wang, J. and Chang, C.-I., "Applications of independent component analysis in endmember extraction and abundance quantification for hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing* **44**, 2601–2616 (Sept. 2006).

[10] Craig, M. D., "Minimum-volume transforms for remotely sensed data," *IEEE Transactions on Geoscience and Remote Sensing* **32**, 542–552 (May 1994).
[11] Boardman, J., Kruse, F., and Green, R., "Mapping target signatures via partial unmixing of AVIRIS data," in [*Summaries of the 5th Annu. JPL Airborne Geoscience Workshop*], Green, R., ed., **1**, 23–26, JPL Publ., Pasadena, CA (1995).
[12] Winter, M. E., "Fast autonomous spectral end-member determination in hyperspectral data," in [*Proceedings of the Thirteenth International Conference on Applied Geologic Remote Sensing*], 337–344 (1999).
[13] Plaza, A., Martinez, P., Perez, R., and Plazas, J., "Spatial/spectral endmember extraction by multidimensional morphological operators," *IEEE Transactions on Geoscience and Remote Sensing* **40**, 2025–2041 (Sept. 2002).
[14] Nascimento, J. M. P. and Bioucas-Dias, J. M., "Vertex component analysis: A fast algorithm to unmix hyperspectral data," *IEEE Transactions on Geoscience and Remote Sensing* **43**, 898–910 (Apr. 2005).
[15] Ritter, G. X., Urcid, G., and Schmalz, M. S., "Autonomous single-pass endmember approximation using lattice auto-associative memories," *Neurocomputing* **72**, 2101–2110 (2009).
[16] Chang, C.-I. and Du, Q., "Estimation of number of spectrally distinct signal sources in hyperspectral imagery," *IEEE Transactions on Geoscience and Remote Sensing* **42**, 608–619 (Mar. 2004).
[17] Tu, T.-M., Huang, P. S., and Chen, P.-Y., "Blind separation of spectral signatures in hyperspectral imagery," in [*Proceedings of the IEEE: Vision, Image and Signal Processing*], **148**, 217–226 (Aug. 2001).
[18] Wu, H.-T., Yang, J.-F., and Chen, F.-K., "Source number estimators using transformed Gerschgorin radii," *IEEE Transactions on Signal Processing* **43**, 1325–1333 (June 1995).
[19] Green, A. A., Berman, M., Switzer, P., and Craig, M. D., "A transformation for ordering multispectral data in terms of image quality with implications for noise removal," *IEEE Transactions on Geoscience and Remote Sensing* **26**, 65–73 (Jan. 1988).
[20] Horn, R. A. and Johnson, C. R., [*Matrix Analysis*], Cambridge University Press (1985).
[21] Ke, Q. and Kanade, T., "Robust l1 norm factorization in the prsence of outliers and missing data by alternative convex programming," in [*Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '05)*], 739746 (Jun. 2005).
[22] Mangasarian, O. L. and Musicant, D. R., "Robust linear and support vector regression," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**(9), 950–955 (2003).
[23] Zare, A. and Gader, P., "Sparsity promoting iterated constrained endmember detection for hyperspectral imagery," *IEEE Geoscience and Remote Sensing Letters* **4**, 446–450 (July 2007).
[24] Berman, M., Kiiveri, H., Lagerstrom, R., Ernst, A., Donne, R., and Huntington, J. F., "ICE: A statistical approach to identifying endmembers in hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing* **42**, 2085–2095 (Oct. 2004).
[25] Kwak, N., "Principal component analysis based on l1-norm maximization," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **30**, 1672–1680 (Sept. 2008).
[26] Li, W. and Swetits, J., "The linear l1 estimator and the huber m-estimator," *SIAM J. Optimization* **8**, 45–475 (1998).
[27] Bioucas-Dias, J. M. and Nascimento, J. M. P., "Hyperspectral subspace identification," *IEEE Transactions on Geoscience and Remote Sensing* **46**, 2435–2445 (Aug. 2008).