Improved Locally Linear Embedding for Big-data Classification

Andres Ramirez
Texas A&M University-Corpus Christi
Corpus Christi, Tx

Maryam Rahnemoonfar*
Texas A&M University-Corpus Christi
Corpus Christi, Tx
maryam.rahnemoonfar@tamucc.edu

ABSTRACT

A hyperspectral image provides a multidimensional data consisting of hundreds of spectral dimensions. Even though having an abundance of spectral might seem favorable, classification of hyperspectral data tends to collide with the curse of dimensionality. Therefore, reducing the number of dimensions before classification is always favorable. For this research, the feature extraction method will consist of a nonlinear manifold learning technique named locally linear embedding (LLE). Additionally, another problem that we attempt to overcome is the high computational time required to run manifold learning methods. In order to help overcome this problem, this research compares one implementation of LLE against an improved version that runs much quicker than the original version.

CCS CONCEPTS

• **Computing methodologies** → *Feature selection*;

KEYWORDS

Locally Linear Embedding, Hyperspectral, Big-data

ACM Reference Format:
Andres Ramirez and Maryam Rahnemoonfar. 2017. Improved Locally Linear Embedding for Big-data Classification. In *BigSpatial'17: BigSpatial'17:6th ACM SIGSP A TIAL Workshop on Analytics for Big Geospatial Data , November 7–10, 2017, Redondo Beach, CA, USA.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3150919.3150925

1 INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have seen unprecedented level of growth in recent years. UAVs are increasingly used for agriculture, search and rescue, surveillance, fire detection, reconnaissance, and mapping, to name a few application domains. The main sensor that has been used in UAVs for majorities of application are optical and Infrared sensors with maximum 4-5 bands. Although hyperspectral sensors are able to obtain an abundance of spectral information, their use is limited to airborne, ground-based, and satellite imagery due to large and expensive computational powers that is needed for processing those images. Hyperspectral images can become uncomfortably large, which introduces new processing challenges. One of the goals in this research is to reduce the computational time that is needed to process hyperspectral imagery so they can ultimately be processed on-board in UAV platforms.

The difficulty of using hyperspectral data is that it consists of redundant and strongly correlated data which can lead into two problems when classifying the data. The first problem is that the classification accuracy of hyperspectral images increases gradually in the beginning as the number of spectral bands increases, but accuracy decreases dramatically when the number of bands reaches a critical value. Additionally, a second problem that arises with the classification of hyperspectral data is that as the number of bands increases, then more training samples are required for each class in order to obtain accurate classification of objects. The second problem is more prominent especially when a class in the hyperspectral data is fairly small. If the pixel size of the class is much less than the number of dimensions in the hyperspectral image, then it is virtually impossible to correctly detect small classes due to the lack of training samples. Therefore, we need to reduce the number of dimensions of hyperspectral data in order to improve classification accuracy even when the class size is small.

In order to reduce dimensions there are two groups of approaches available in the literature: linear and non-linear .Non-linear dimension reduction methods, namely manifold learning, have been investigated in order to extract the nonlinear characteristics embedded within hyperspectral data. The linear dimensional reduction algorithms include Principal Component Analysis (PCA), Singular Value Decomposition (SVD) and Independent Component Analysis (ICA). Even though linear dimensionality reduction is useful, it is sometimes not enough to extract the linear variance among the data because the data can potentially have a non-linear variance[6].

Manifold learning can be classified as global or local approaches in order to extract non-linear variance from high dimensional datasets. The global manifold methods simply keep the fidelity of the overall data set but unfortunately have a greater computational overhead for large data sets, while local methods preserve local geometry and are computationally efficient.

The solutions to reduce dimensions for this research is a non-linear feature extraction approach referred to as LLE. We investigate and compare results obtained by an initial version of LLE vs. an improved version of LLE in order to classify vegetation hyperspectral images.

1.1 Related work

Several researchers have used PCA[5, 13, 17]. Overall, the use of PCA results in high computational analysis time and can inaccurately represent non-linear data which can reside within a hyperspectral image. Even though PCA is a time consuming algorithm, researchers have attempted to improve the time complexity by creating algorithms such as folded or segmented PCA[17] resulting in a 10% efficiency increase.

*Corresponding author

BigSpatial'17, November 7–10, 2017, Redondo Beach, CA, USA

ACM ISBN 978-1-4503-5494-3/17/11…$15.00
https://doi.org/10.1145/3150919.3150925

Additionally, the non-linear implementations of feature extraction are manifold learning techniques. There are both local and global manifold learning techniques where the former tends to be significantly faster compared to the latter[1, 6, 8, 10, 16]. Global manifold learning techniques include:multidimensional scaling, maximum variance unfolding, and isometric mapping (ISOMAP), Kernel-PCA (KPCA). Localized manifold learning include: Laplacian Eigenmaps (LE), Local Linear Embedding (LLE) and Local Tangent Space Alignment (LTSA).

Regarding LLE, S. Roweis et al.[10] propose and describe the mathematical derivation of LLE where it is referred to as an unsupervised learning algorithm in order to compute low-dimensional, neighborhood-preserving embeddings of high-dimensional inputs. Their implementation of LLE was tested on images of faces and vectors of word-documented counts. They conclude by stating that LLE is likely to be even more useful in combination with other methods in data analysis and statistical learning.

L. Ding et al.[3] propose a new object-oriented mapping approach based on nonlinear subspace feature analysis of hyperspectral remote sensing images. They implemented the standard LLE, laplacian eigenmaps (LE), and local tangent space alignment (ltsa) along with the their proposed method of the local manifold learning algorithms. The dataset used was a sub-image of AVIRIS data and the Indiana Indian Pine with 200 of the original 220 spectral channels and 12 of the 16 original classes were tested. They concluded that their new proposed method returns more accurate mapping results than the pixel-wise classification where LLE and LTSA consistently yielded better results than LE.

K. Uto et al.[15] propose a new method to acquire inherent vegetation-related coordinates on hyperspectral manifold by the combination of unsupervised manifold learning and supervised vegetation-related coordinates estimation. Their experimental results show high estimation performance in vegetation-related quantities by the proposed method i.e. nonlinear structure extraction and improved generalization performance in comparison with multivariate linear regression based on hyperspectral data.

Thenkabail, P.S. et al. [14] investigated the best hyperspectral wavebands in the study of vegetation and agricultural crops between the spectral range 400-2500 nm; and assess the vegetation and agricultural crop classification accuracies achievable using the various combinations of the best hyperspectral narrow wavebands. They applied principal component analysis (PCA), lambda-lambda R2 models (LL R2M), stepwise discriminant analysis (SDA), and derivative greenness vegetation indices (DGVI) to concluded that 22 optimal bands best characterize and classify vegetation and agricultural crops. Using more than 22 bands would only increase accuracy marginally up to 30 bands. Beyond 30 bands, accuracies would become asymptotic or near zero.

2 PROPOSED METHODOLOGY

In this section a detailed description of LLE along with the improved algorithm is provided.

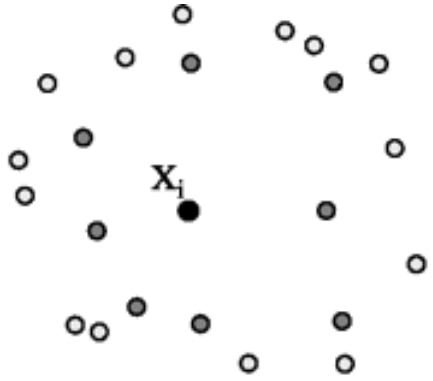

Figure 1: Obtaining K-NN's for X_i

2.1 Locally Linear Embedding

The general LLE[6][4] consists of three main steps: search for the nearest neighbors, calculate reconstruction weights and then determine low-dimensional embedding. We can mathematically describe the three steps with the following expressions.

We first find the k-nearest neighbors:

$$d = \sqrt{(\vec{x_i} - \vec{x_j})^T (\vec{x_i} - \vec{x_j})} \tag{1}$$

where d is the Euclidean distance between data point $\vec{x_i}$ and $\vec{x_j}$ in column vectors for $i, j = 1, 2...n$ and n is the number of data points. For this research, n is equivalent to the total number of pixel vectors as shown in figure **??**. Additionally, a visual representation of this step is shown in figure 1.

Then, according to the second step, we calculate the reconstruction weights by minimizing:

$$\left\| \vec{x_i} - \sum_{j=1}^{k} w_{ij} \vec{x_j} \right\|^2 \tag{2}$$

subject to:

$$\sum_{j=1}^{k} w_{ij} = 1 \tag{3}$$

where k is the number of nearest neighbors used for reconstructing each data point. It might not be obvious, but this is a constrained least squares problem that has the following closed-form solution:

$$\vec{w_i} = S_i^{-1} \vec{v} \tag{4}$$

where $\vec{v} = [1, 1, \ldots, 1]^T$ and $S_i = (X_i - N_i)^T (X_i - N_i)$. Then X_i is a $d \times k$ matrix whose columns are duplicates of $\vec{x}$ and N_i is also a $d \times k$ matrix whose columns are the k nearest neighbors of $\vec{x}$. Lastly,d is the dimensionality of the original space and the size of $\vec{v}$ is equivalent to the k value previously stated. This step is visualized in figure 2 where we take the k values from 1 and then we reconstruct the linear weights.

The last step is to determine the low-dimensional embedding which is implemented by minimizing:

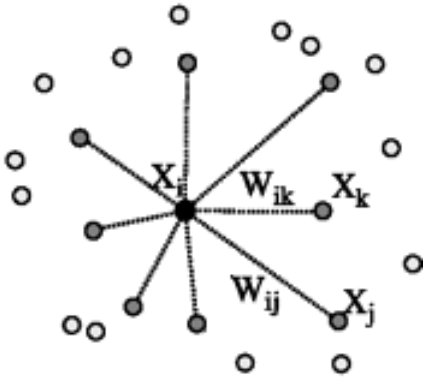

Figure 2: Obtaining reconstruction coefficients with linear weights

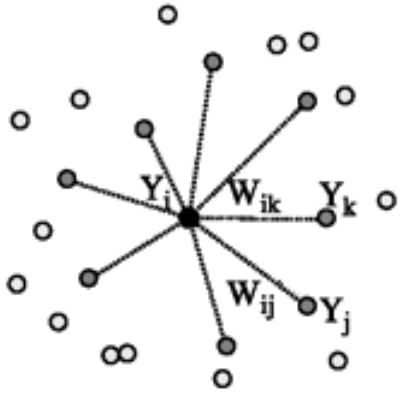

Figure 3: Mapping coordinates to lower dimension

$$M = \sum_{i=1}^{n} \left\| \vec{y_j} - \sum_{j=1}^{n} w_{ij} \vec{y_j} \right\|^2 \tag{5}$$

where $\vec{y_i}$ is the coordinate of the data point $\vec{x_i}$ in the low-dimensional embedding to be determined with the dimensionality of r. In order to facilitate the low-dimensional embedding, it has been proven [11] $\vec{y_j}$ is equivalent to calculating the eigenvectors corresponding to the r smallest nonzero eigenvalues of $(I-W)^T(I-W)$[4] where I is a $n \times n$ identity matrix and W is also a $n \times n$ matrix, each of whose rows are composed of a data points reconstruction weights. Therefore, by extracting the eigenvector with the smallest eigenvalues, we have successfully minimized equation 5. Additionally, the eigenvector with the smallest eigenvalue (i.e. the eigenvector with an eigenvalue of 0) corresponds to the mean of equation 5 and can be discarded to enforce $\sum_{i=1}^{n} y_j = 0$ [2]. We illustrate this step with figure 3 where we calculate the embedding of coordinates for point X_i and we obtain the new point Y_i.

In order for the original LLE algorithm to work correctly, matrix S_i produced in the second step, equation 4, has to be regulated only when the number of nearest neighbors is more than the input dimensionality (when $k > d$). This can be done by multiplying the diagonal of S_i by a very small number. Furthermore, the between-band correlation for hyperspectral data is significant and matrix S_i often becomes singular when $k < d$ [4]. Therefore, in order to solve for the reconstruction weights, regulation will always be required which is achieved by adding a small value to all the diagonal elements of S_i.

Additionally, it is important to note that LLE is much more intensive in computation and memory consumption compared to Principal Component Analysis (PCA). In the worst case scenerio, the computing complexities of the three LLE steps are of order $O(dn^2)$, $O(dnk^3)$ and $O(rn^2)$, respectively where d is the input data dimensionality, k is the number of nearest neighbors, n is the number of data vectors, and r is the output dimensionality. In order to add to complexity, the size of the reconstruction weight matrix W is nxn. Now, lets assume that we have a 640×512 hyperspectral image, then $n = 327680$. The matrix of such size goes beyond the memory capacity of average PC's even though W is sparse (W is the matrix containing the data points reconstructions weights).

2.2 Initially Restarted Arnoldi Method

The Arnoldi iteration[12][7] is an eigenvalue algorithm that finds the eigenvalues of any general square matrix A. Note that in order to find the eigenvalues/vectors of A we can form the characteristic polynomial and then find the roots of the polynomial as the eigenvalues/vectors of A, but this method is not suitable for large sparse matrices. Therefore, to overcome that we run the eigendecomposition starting like the *Power Iteration* method which starts with Ab where b is an arbitrary vector. Then we continue with Ab^2, and so we can write this as: $[Ab, A^2b, A^3b, A^nb]$ where A^nb will converge to the largest or smallest eigenvector of the matrix A. Although, much potentially useful computaiton is wasted by using only the final result $A^{n-1}b$ so instead we create the *Krylov matrix* K such that $K_n = [b, Ab, A^2b, A^3b, A^n - 1b]$. This method is unstable, therefore in order to stabilize this method we ensure that each vector is orthogonal to the previous vector, which is referred to as the *Arnoldi iteration*. We ensure the that each vector is orthogonal to the previous vectors by applying the *Gram-Shmidt* orthogonalization procedure. Therefore we start the Arnoldi iteration with an arbitrary vector q_1 with a norm of 1 in order to have an orthonormal basis:

$$||q_1|| = 1 \tag{6}$$

Then we repeat for q_k through q_n such that $q_k = Aq_{k-1}$. In order to orthogonalize the vector to all the vectors before k, then we state the following:

$$\text{for } j = 1, 2, \ldots, k-1$$

$$h_{j,k-1} = q_j^T q_k \tag{7}$$

$$q_k = q_k - (q_j^T)(h_{j,k-1})q_j \tag{8}$$

Where equation 7 is the scalar that defines the length of the projection onto the vector j. Then we can normalize q_k, so we end the loop and proceed by setting $h_{k,k-1} = ||q_k||$ and scale q_k to be normal with: $q_k = \frac{q_k}{h_{k,k-1}}$ which finalizes the Arnoldi iteration. After the previous procedure is complete, we produce matrix H_n such that:

$$\mathbf{H}_n = \begin{pmatrix} h_{1,1} & h_{1,2} & h_{1,3} & \dots & h_{1,n} \\ h_{2,1} & h_{2,2} & h_{2,3} & \dots & h_{2,n} \\ 0 & h_{3,2} & h_{3,3} & \dots & h_{3,n} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & h_{n,n-1} & h_{n,n} \end{pmatrix} \quad (9)$$

Where H_n is the upper *Hessenberg* matrix formed by the numbers $h_{j,k}$ computed by the Arnoldi algorithm. An upper *Hessenberg* matrix is a special kind of square matrix that has zero entries below the first sub-diagonal. In comparison, a lower *Hessenberg* matrix has zero entries above the first super-diagonal. Since the Arnoldi iteration projects matrix A onto the *Krylov* subspace, the projection represented by H_n, we can calculate the eigenvalues and eigenvectors efficiently by applying the QR algorithm. The QR decomposition of a matrix involves decomposing matrix H_n into a product $H_n = QR$ of an orthogonal matrix Q and an upper triangular matrix R.

2.3 Improved Embedding Coordinate Step

For the LLE, Arnoldis iterative algorithm is useful because we want to only keep a few eigenvalue/eigenvector pairs. Furthermore, it is useful to run Arnoldis method for LLE since we are handling a large sparse matrix towards the end of the algorithm. We apply Arnodis algorithm during equation 5 where we first find $M = (I-W)^T(I-W)$ and then apply Arnoldis algorithm on M where we extract the smallest eigenvectors in order to minimize equation 5. In this case, we also only require the eigenvectors that pertain to the smallest eigenvalues rather than obtaining all n eigenvalues/vectors where n = pixel vectors. This procedure is much more helpful when compared to the extraction of all eigenvalues/eigenvectors of all large sparse matrix. Obtaining all eigenvalue/eigenvector pair of a large sparse matrix is typically impractical since keeping the first n eigenvalue/vectors can retain most of the variance and thus lowering the amount of memory space required to obtain the eigenvalues/vectors.

The second enhancement applied to the original LLE algorithm pertains to the calculation of embedding from eigenvectors of the error cost matrix. Rather than creating iteratively a final sparse matrix and directly access the coefficients of a sparse matrix that requires a complexity of $O(n^2)$, we instead obtain the product of $M = (I-W)^T(I-W)$, where I is the identity matrix and W is a sparse matrix holding all reconstruction weights. This part improves the time because we take advantage of data reuse by maximizing the utilization of fast local storage. For example, asssuming that we have matrix C such that $C = AB$, then we can decompose both A and B into block matrices. Then we can recursively apply matrix-matrix multiplication on the block matrices which allows for better locality of reference both in space and time of the data used in the product. Therefore this procedure takes advantage of the cache on the system. Additionally, if the system running the program has more than one level of cache, then the blocking can be applied a second time to improve with data access time.

3 RESULTS AND DISCUSSION

In this paper we applied LLE and our proposed technique on several hyperspectral imagery from AVIRIS sensor. We calculated running time both for LLE and our proposed method for different K nearest neighbours.

The spectrometer used to obtain the hyperspectral scene is NASA's Airborne Visible/Infrared Imaging Spectrometer (AVIRIS)[9] which is an instrument for Earth Remote Sensing flown aboard various aircraft types.

According to the graph seen in figure 4, we notice that as the value for K increases, the time it took to run the initial version of the program takes much longer compared to the improved program. It is apparent that as the number of KNN increases, then the time it takes to run the initial program increases in a similar manner to exponential growth. In comparison, the improved program seems to increase in a similar manner to linear growth.

Additionally, this graph displays the following results because the first initial version created[10] contains an iterative process in order to create matrix M from equation 5 that takes $O(n^2)$ time followed by the extraction of the Arnoldi method in order to obtain the eigenvectors. Therefore, it makes sense that the time to run the initial program has a time increase similar to the growth of $O(n^2)$. Thus, by removing that iterative process and instead applying a direct matrix multiplication to finalize matrix M, then we can get a much quicker runtime in order to complete LLE.

4 CONCLUSION

For this research, we propose an improvement of the Locally Linear Embedding algorithm in order to extract local non-linear patterns from a hyperspectral dataset. The improvement applied dealt with a faster matrix-matrix multiplication step for equation 5 that reuses local data. Then, after the matrix-matrix multiplicaiton step, we apply Arnoldis method in order to find the smallest eigenvalues/vectors which finalizes the equation 5 of the LLE algorithm.

As future work we can focus on improving the time it takes to run Arnoldis method in order to further reduce the time complexity of LLE.

5 ACKNOWLEDGMENT

This work was partially supported by NSF 1429518.

REFERENCES

[1] Junhwa Chi and M.M. Crawford. 2013. Selection of Landmark Points on Nonlinear Manifolds for Spectral Unmixing Using Local Homogeneity. *Geoscience and Remote Sensing Letters, IEEE* 10, 4 (July 2013), 711–715. https://doi.org/10.1109/LGRS.2012.2219613

[2] Dick de Ridder and Robert P.W. Duin. 2002. Locally Linear Embedding For Classification. (2002).

[3] Ling Ding, Ping Tang, and Hongyi Li. 2014. Subspace Feature Analysis of Local Manifold Learning for Hyperspectral Remote Sensing Images Classification. *Applied Mathematics & Information Sciences* 4 (2014), 1987–1995.

[4] Tian Han and D.G. Goodenough. 2005. Nonlinear feature extraction of hyperspectral data based on locally linear embedding (LLE). In *Geoscience and Remote Sensing Symposium, 2005. IGARSS '05. Proceedings. 2005 IEEE International*, Vol. 2. 1237–1240. https://doi.org/10.1109/IGARSS.2005.1525342

[5] R. Heylen, D. Burazerovic, and P. Scheunders. 2011. Non-Linear Spectral Unmixing by Geodesic Simplex Volume Maximization. *Selected Topics in Signal Processing, IEEE Journal of* 5, 3 (June 2011), 534–542. https://doi.org/10.1109/JSTSP.2010.2088377

[6] Wonkook Kim. 2011. *Manifold Learning for Robust Classification of Hyperspectral Data.* Ph.D. Dissertation. Purdue University, West Lafayette, Indiana.

[7] R. B. Lehoucq and D. C. Sorensen. 1996. Deflation Techniques for an Implicitly Restarted Arnoldi Iteration. *SIAM J. Matrix Anal. Appl.* 17, 4 (Oct. 1996), 789–821. https://doi.org/10.1137/S0895479895281484

[8] D. Lunga, S. Prasad, M.M. Crawford, and O. Ersoy. 2014. Manifold-Learning-Based Feature Extraction for Classification of Hyperspectral Data: A Review of

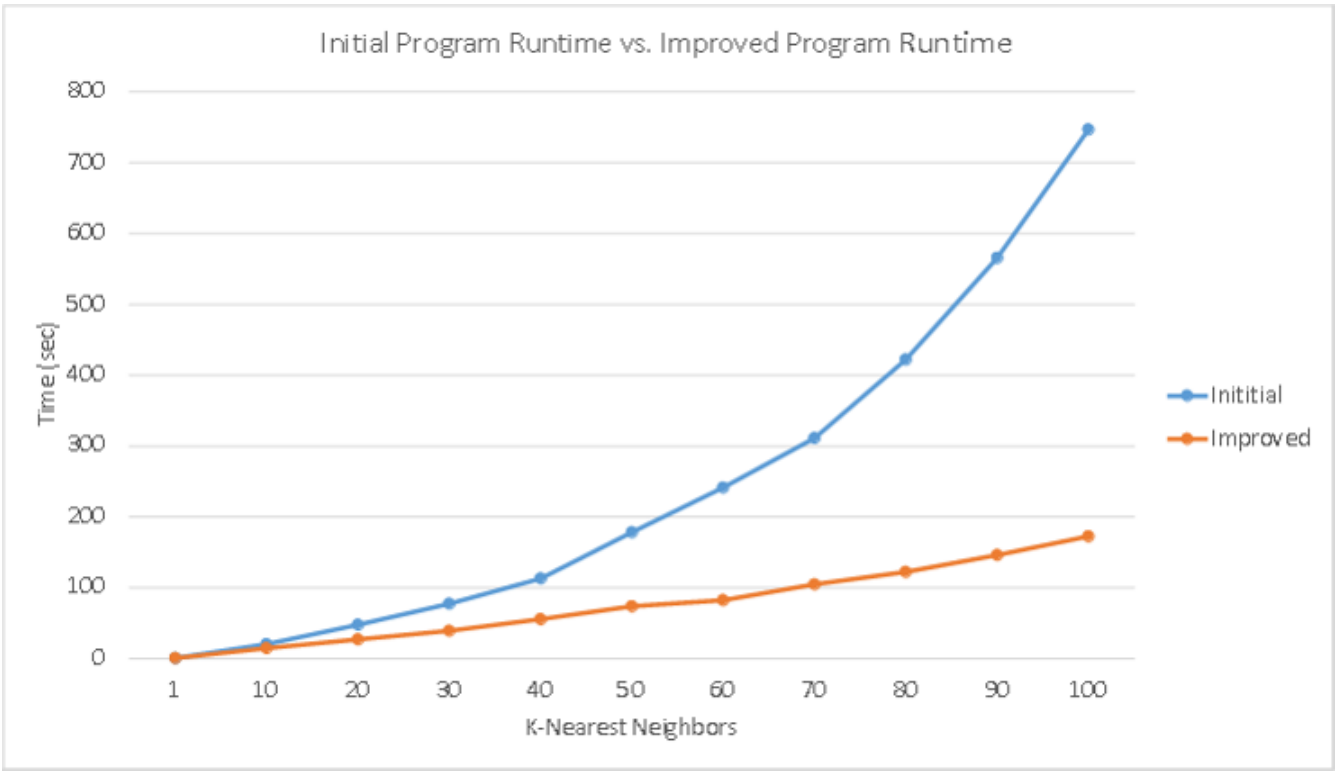

Figure 4: LLE Program Time Comparison

Advances in Manifold Learning. *Signal Processing Magazine, IEEE* 31, 1 (Jan 2014), 55–66. https://doi.org/10.1109/MSP.2013.2279894

[9] NASA-Jet Propulsion Laboratory. 2007. AVIRIS Concept. (2007). http://aviris.jpl.nasa.gov/html/aviris.concept.html

[10] Sam T. Roweis and Lawrence K. Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *SCIENCE* 290 (2000), 2323–2326.

[11] Sam T. Roweis and Lawrence K. Saul. 2000. Nonlinear dimensionality reduction by locally linear embedding. *SCIENCE* 290 (2000), 2323–2326.

[12] Youcef Saad. 2011. *Numerical methods for large eigenvalue problems.* SIAM. https://doi.org/10.1137/1.9781611970739

[13] Prasad S. Thenkabail. 2001. Optimal hyperspectral narrowbands for discriminating agricultural crops. *Remote Sensing Reviews* 20, 4 (2001), 257–291. https://doi.org/10.1080/02757250109532439 arXiv:http://dx.doi.org/10.1080/02757250109532439

[14] Prasad S. Thenkabail, Eden A. Enclona, Mark S. Ashton, and Bauke Van Der Meer. 2004. Accuracy assessments of hyperspectral waveband performance for vegetation analysis applications. *Remote Sensing of Environment* 91, 3âĂŞ4 (2004), 354 – 376. https://doi.org/10.1016/j.rse.2004.03.013

[15] K. Uto, T. Harano, and Y. Kosugi. 2012. Rice growth state estimation by hyperspectral manifold learning. In *Geoscience and Remote Sensing Symposium (IGARSS), 2012 IEEE International*. 4178–4181. https://doi.org/10.1109/IGARSS.2012.6350936

[16] Sreenath Rao Vantaram and Eli Saber. 2012. Survey of contemporary trends in color image segmentation. *Journal of Electronic Imaging* 21, 4 (2012), 040901–1–040901–28. https://doi.org/10.1117/1.JEI.21.4.040901

[17] Jaime Zabalza, Jinchang Ren, Mingqiang Yang, Yi Zhang, Jun Wang, Stephen Marshall, and Junwei Han. 2014. Novel Folded-PCA for improved feature extraction and data reduction with hyperspectral imaging and SAR in remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing* 93, 0 (2014), 112 – 122. https://doi.org/10.1016/j.isprsjprs.2014.04.006