Sammon mapping for preliminary analysis in Hyperspectral Imagery

Nicolae APOSTOLESCU*,[1], Daniela BARAN[2]

*Corresponding author
*,[1]AEROSPACE Consulting
B-dul Iuliu Maniu 220, Bucharest 061126, Romania
apostol@incas.ro
[2]INCAS – National Institute for Aerospace Research "Elie Carafoli"
B-dul Iuliu Maniu 220, Bucharest 061126, Romania
baran.daniela@incas.ro

DOI: 10.13111/2066-8201.2016.8.1.2

Received: 05 October 2015 / Accepted: 22 December 2015 / Published: March 2016

The 36th "Caius Iacob" Conference on Fluid Mechanics and its Technical Applications
29 - 30 october, 2015, Bucharest, Romania, (held at INCAS, B-dul Iuliu Maniu 220, sector 6)
Section 4. Mathematical Modeling

Abstract: *The main goal of this paper is to present the implementation of the Sammon algorithm developed for finding N points in a lower m-dimensional subspace, where the original points are from a high n-dimensional space. This mapping is done so interpoints Euclidian distances in m-space correspond to the distances measured in the n-dimensional space. This method known as non-linear projection method or multidimensional scaling (MDS) aims to preserve the global properties of points. The method is based on the idea of transforming the original,* n*-dimensional input space into a reduced,* m*-dimensional one, where* m<n*, and it may be used to clustering hyperspectral data. The Principal Component Analysis (PCA) may be applied as a pre-processing procedure for starting, in order to obtain the N points in the lower subspace. The algorithm was tested on hyperspectral data with spectra of various lengths. Depending of the size of the input data (number of points), the number of learning iterations and computational facilities available, Sammon mapping might be computationally expensive.*

Key Words: *hyperspectral data, Sammon mapping method, Euclidian distances, non-linear mapping techniques*

1. INTRODUCTION

Hyperspectral sensors used in Hyperspectral Imagery collect information of earth surfaces as a set of images that correspond to the same spatial scene, but are acquired at many different spectral bands with high resolution. These images contain abundant spatial, spectral, and radiometric information, which makes earth observation and information acquisition much more efficient for real life applications. Hyperspectral data have details and accuracy allowing the investigation of phenomena and concepts that greatly extend the scope of traditional remote sensing [2]. In terms of spectral properties, the high resolution has the capability of uncovering unknown sources, which cannot be identified by visual inspection.

The spectral signature leads to a better separation between physical materials and objects [1]. This is based on the fact that in the hyperspectral image, reflectance information depends only of the materials spectral responses in the scene.

A mixed pixel is either linear or nonlinear combination of pure pixels signatures weighted by the correspondent abundance fraction. Many techniques of unmixing in hyperspectral image analysis are based on geometric approach where each pixel is seen as a spectral vector of p (number of spectral bands). Under the linear model we assume that the number of substances and their spectra are known, but in reality they are not and, then, hyperspectral unmixing falls into the blindly classes. When the mixture between materials is macroscopic, the linear mixing model of spectra is generally admitted because this model assumes no interaction between materials [4, 7].

Although the techniques of classical spectroscopy can be used in hyperspectral remote sensing to examine, for example, atmospheric gases, hyperspectral remote sensing examines very detailed spectra for images of the earth's surface and matches them to spectra of known features [2]. The large dimensionality of a hyperspectral dataset often requires data transformation which can effectively reduce dimension of data sets with minimum loss of information [3, 5, 6, 10]. These are intended to find the minimum number of parameters required to represent the observed properties of the data [9]. Several methods have been implemented for determining the dimension of signal subspace that is done by the smallest number of parameters needed to contain all of the variability in the data [8, 11]. This means a dimensionality reduction. The entire subject is based around the idea that we have this big set of data, and we want to analyses that set in terms of the relationships between the individual points in that data set. Hyperspectral images are characterized by a high number of bands, which are highly correlated for neighboring bands. Sammon mapping is one of the first and most popular nonlinear dimensionality reduction techniques [16, 18]. Also it has been widely applied to the visualization of the high dimensional data. The Sammon mapping is based on the idea of transforming the original, n-dimensional input space into a reduced, m-dimensional one, where $m<n$. It is also known as non-linear projection method or multidimensional scaling (MDS) method. Through this mapping function we give the minimum embedding dimension from the hyperspectral data. It may be applied as a pre-processing procedure, and its resulting components may be used as inputs to clustering and supervised classification models. Sammon method places data items in an abstract space with a chosen number of dimensions in such a way that the distances between the positions reflect the differences (dissimilarities) between the data items.

A hyperspectral image can be illustrated as an image cube with the two dimensions of the cub face representing the spatial information and the third dimension representing the spectral information. The information available in a hyperspectral image (cube) is organized in a tridimensional matrix (each plane is an image corresponding to one wavelength band) denoted by *HS3D*(h,w,p). We use here this notation to show that the tridimensional matrix represents a hyperspectral image, h and w are the height and width of image, respectively. In this matrix the item x_{ijk}, $i \in 1,\cdots,h$; $j \in 1,\cdots,w$; $k \in 1,\cdots,p$ is the pixel value from i row, j column and k spectral band. The pixels $X_{ij} = \lfloor x_{ij1}, x_{ij2},\cdots,x_{ijp} \rfloor$ in an image can be considered as vectors in R^p space.

From the point of view of subsequent calculations we abandon the idea that the three-dimensional matrix represents the image; it is considered now that it is just a three-dimensional matrix that will be recognized in a bidimensional matrix, HS2D(h*w,p), by the rule: first column of the bidimensional matrix includes all the columns of HS3D, placed

successively, where, p=1; second column will contain all columns of the HS3D matrix, placed successively, where p=2, and so on. For each image pixel, the spectral bands can then be written as vectors. So, the item $hs3d\,(i,j,k) = hs2d\,[\,(j-1)*h+i,k\,]$. We note further $m = h*w,\ n = p$ and *HS2D=X*. Note $X = X(i,j); i = 1,2,\cdots,m; j = 1,2,\cdots,n.$

Using the statistical language, we consider the vectors pixels of a hyperspectral image as observations and their components as variable associated features.

2. MAPPING MULTIDIMENSIONAL DATA TO A LOWER DIMENSION

Using the statistical language, we consider the vectors pixels of a hyperspectral image as observations and their components as variable associated features. In this context, **Principal component analysis** (**PCA**) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called **principal components**. The number of the principal components is less than or equal to the number of original variables because PCA constructs a low-dimensional representation of the data that describes as much of the variance in the data as possible. This is done by finding a linear basis of reduced dimensionality for the data, where the variance of data is maximal [13]. Therefore PCA is a powerful tool capable of reducing dimensions and revealing relationships among data items. It is intimately related to the mathematical technique of singular value decomposition (SVD). Hyperspectral data are well suited for analyse using SVD and PCA [12]. The authors generally not to use PCA as a pre-processing step for clustering algorithms.

The Singular Value Decomposition requires that the full set of data be available during processing, that can involve a significant storage cost. Technically, a principal component can be defined as a linear combination of optimally-weighted observed variables. "Optimally weighted" refers to the fact that the observed variables are weighted in such a way that the resulting components account for a maximal amount of variance in the data set. The main feature of the PCA method is that the *k* component will be correlated with at least some of the observed variables and will be uncorrelated with all of the preceding components. PCA maximize the variance in each successive hyperspectral band. Therefore, only the first few components are retained, interpreted, and used in subsequent analyses. PCA is similar in many respects to exploratory factor analysis.

Summarizing, the goal of the PCA method is to transform a/an *X* data of dimension p to an/ a new data *Y* of smaller dimension $s<p$. PCA is normally conducted in a sequence of steps. In our implementation of the PCA method, we use the covariance method because it is better to use the singular value decomposition and go through the steps:

1. calculate the mean value along each dimension $\mu(j) = \frac{1}{m}\sum_{i=1}^{m} X(i,j),\ j = 1,2,\cdots,n$
2. calculate deviations from the mean value $X = X - U\mu^T$, where U is an *mX1* vector, $U(i) = 1,\quad i = 1,2,\cdots,m$. It results $X(m,n)$. Each row of X is an observation, and each of its columns is a variable.
3. find the covariance matrix $C(n,n)$ of *X*. In general, the covariance of two random vectors X, Y is their tendency to vary together. It can be explicitly written as: $Cov(X,Y) = \sum_{i=1}^{N} \frac{(x_i - \bar{x})\cdot(y_i - \bar{y})}{N}$ where N is the vector's dimension,

$\bar{x} = mean(X)$, $\bar{y} = mean(Y)$. In some cases it is preferred to use the correlation matrix instead of the covariance matrix.

4. decompose the covariance matrix to find its eigenvectors. Compute the matrix $\boldsymbol{V}$ of eigenvectors which diagonalizes the covariance/ correlation matrix $V^{-1}CV = D$ where $D(n,n)$ is the diagonal matrix of eigenvalues of *C*. $C = VDV^T$; $VV^T = I$.
5. $D(i,j) = \lambda_i$ for $i = j$ and $D(i,j) = 0$ for $i \neq j$;
6. calculate $g(i) = \sum_{j=1}^{n} \lambda_j, \quad i = 1,2,\cdots,n$
7. sort the columns of the eigenvector matrix *V* and eigenvalues in decreasing order and,
8. retain only *s* eigenvalues, *s<n;* $\lambda_1, \lambda_2, \cdots, \lambda_s$, *so that* $\frac{g(s)}{g(p)} \geq theta$, *where* $theta \approx 0.9$.
9. choose the subspace dimension *s* and construct the matrix X^s from X :
10. $X^s = VX$. Finaly, $X = X^T$.

The first principal component is the eigenvector corresponding to the largest value λ. This vector is the direction along which the data have the most variance. The second principal component is the second eigenvector. Its direction is orthogonal to the first component. Because it is orthogonal to the first eigenvector, their projections will be uncorrelated. In fact, projections onto all the principal components are uncorrelated with each other. The eigenvalues will give the total variance described by each component. The variance of the projections onto the first s principal components is then $\sum_{i=1}^{s} \lambda_i$.

3. SAMMON ALGORITHM

Consider the original bidimensional matrix $X = X(i,j); i = 1,2,\cdots,m; j = 1,2,\cdots,n$ as a set of points in R^n space. Suppose each row X_i, $i = 1,2,\cdots,m.$ of *X* is a point in the R^n space and corresponding to these we define also m points in a R^s (s-space) designated Y_i, $i = 1,2,\cdots,m.$ Seek to identify the geometric relationships among subsets of the data vectors in the both space. Sammon projection or Sammon mapping is a procedure that maps a high-dimensional space to a space of lower dimensionality in such a way as to preserve the structure of inter-point distances on each space [16, 17, 18]. The distance-preserving aspect can be of importance when one wants to use classifiers sensitive to these distances, such as the nearest-neighbor classifiers. If we denote d_{ij} as the interpoint distances from X_i to X_j measured in the *n*-dimensional space and d_{ij}^* the interpoint distances from Y_i to Y_j measured in the s-dimensional space, the Sammon mapping goal is to minimize an error function: $E = \frac{1}{\sum_{i<j} d_{ij}} \sum \frac{(d_{ij} - d_{ij}^*)^2}{d_{ij}}$. This function is often referred to as Sammon stress or Sammon error and it is minimized by a gradient descent technique that adjusts the position of the points in the space.

1. calculate Euclidian distance from each point X_i to each point X_j .

$$d_{ij} = D(i, j) = \sqrt{\sum_{k=1}^{n} \left(x_{ik} - x_{jk}\right)^2}, \quad i = 1,2,\cdots,m;\ j = 1,2,\cdots,m .$$

2. Use the Singular Value Decomposition (SVD) of X matrix. SVD decomposes a matrix into a set of rotation and scale matrices, which is used in computing the rank of matrix. The general form of SVD decomposition is:
$X = USV^T$ where $X \in R^{mxn}$ with $m \geq n, U \in R^{mxm}, V \in R^{nxn}, S \in R^{mxn}$. U and V are orthogonal and called the left singular vector and the right singular vector, respectively. The S matrix is called the singular value and its elements are only nonzero on the diagonal. From the SVD calculation, the PCA output is calculated by outer product of the columns of U with the singular values diag(S).
3. estimate the mapping dimension as $s<n$
4. calculate a matrix $Y=U*S$ so that the dimension of Y is mXs.
Another option to build the matrix Y is the random mode.
5. calculate Euclidian distance from each point Y_i to each point Y_j,
6. $d_{ij}^* = D^*(i, j) = \sqrt{\sum_{k=1}^{s} \left(y_{ik} - y_{jk}\right)^2}, \quad i = 1,2,\cdots,m;\ j = 1,2,\cdots,m .$
7. calculate the Sammon stress function $E = \frac{1}{\sum_{i<j} d_{ij}} \sum_{i<j} \frac{(d_{ij} - d_{ij}^*)^2}{d_{ij}}$ and minimize it.

The minimization of the error E is an optimization problem with discrete variables and can be performed by gradient descent ∇f , and by evaluate Hessians $\nabla^2 f$, usually involving iterative methods. The number of iterations needs to be experimentally determined and convergent solutions are not always guaranteed. It may be transcribed into a step-by-step format for easy implementation into a computer. The iterative process can be written in the component form as:

$$y_{il}(t+1) = y_{il}(t) - \alpha \left[\frac{\frac{\partial E(t)}{\partial y_{il}(t)}}{\frac{\partial^2 E(t)}{\partial y_{il}^2(t)}} \right] \tag{1}$$

where t is the iteration step and α is a nonnegative constant so called "magic-factor" that is founded experimentally as $\alpha \approx 0.3 - 0.4$ and it represents the step size for gradient search in the direction of $\frac{\partial E(t)}{\partial y_{il}(t)}$.

$$\frac{\partial d_{ij}^*}{\partial y_{il}} = \left(\sqrt{\sum_{k=1}^{s} \left(y_{ik} - y_{jk}\right)^2} \right)' = \frac{2\left(y_{il} - y_{jl}\right)}{2\sqrt{\sum_{k=1}^{s} \left(y_{ik} - y_{jk}\right)^2}.} = \frac{\left(y_{il} - y_{jl}\right)}{d_{ij}^*} \tag{2}$$

$$\frac{\partial E(t)}{\partial y_{il}(t)} = -\frac{2}{\lambda} \sum_{k=1, k\neq i}^{N} \left[\frac{d_{ki} - d_{ki}^*}{d_{ki} d_{ki}^*} \right] (y_{il} - y_{kl}) \tag{3}$$

Note: From the fact that *D* is a symmetrical matrix

$D(i,j)=D(j,i),\quad D^*(i,j)=D^*(j,i);\quad i=1,2,\cdots,m;\ j=1,2,\cdots,m$ and

$D(i,i)=0;\qquad D^*(i,i)=0;\qquad i=1,2,\cdots,m;$

To avoid dividing by zero, the computer program use only upper diagonal distances matrices.

$$\frac{\partial^2 E(t)}{\partial y_{il}^2\ (t)}=-\frac{2}{\lambda}\sum_{k=1,k\neq i}^{N}\frac{1}{d_{ki}d_{ki}^*}\left[(d_{ki}-d_{ki}^*)-\left(\frac{(y_{il}-y_{kl})^2}{d_{ki}^*}\right)(1+\frac{d_{ki}-d_{ki}^*}{d_{ki}})\right] \tag{4}$$

Depending of the size of the input data (number of points), the number of iterations and computational facilities available, Sammon's mapping might be computationally expensive.

4. COMPUTER PROGRAMS AND DATA FILES

The nonlinear mapping algorithm was tested and /evaluated on several artificially generated data set [14]. For starters the *s*-space is 2-and 3-dimensional spaces since the resultant data was evaluated by human observation. The source code of the computer programs were written in Matlab R2013a. To demonstrate the applicability of the software tools developed we present some results using images scenes acquired by Airborne Visible IR Imaging Spectrometer (AVIRIS) using 224 channels ranging from 0.4 to 2.5μm with spatial and spectral resolutions of approximately 20m and 10nm, respectively: the Purdue Indiana Indian Pine test site, covering an area of mixed agriculture and forestry, in a North-western Indiana image scene. In this case the hyperspectral data is a tridimensional matrix hs3d (145,145,200), therefore hs2d (145*145,200). The covariance matrix is C (200,200). When we choose $\mathrm{theta}=0.85$ the number of the principal components is 20.

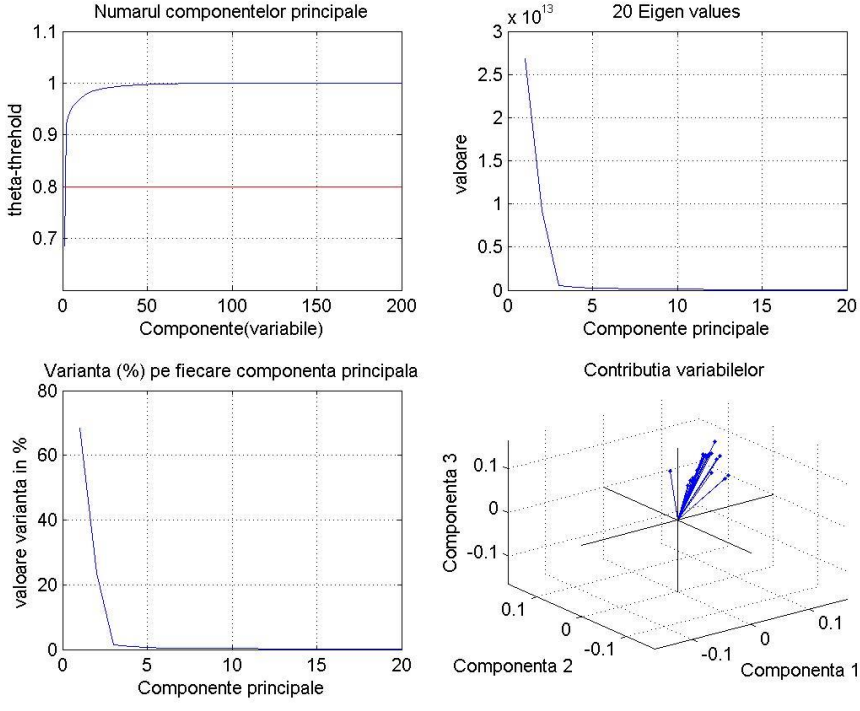

Fig. 1 Results obtained after the PCA

For better handling the Sammon application we build a graphical user interface (GUI). It allows the user to choose different options on the number of the principal components, on the initialization method of Y matrix (PCA or random mode), or relative to the display mode of results [15].

Also, the user can specify the number of iterations and cumulative error of distances between points. GUI enables repeated applications and comparisons of the results to specify the best number of classification clusters.

This GUI is written on Matlab environment. It is created using the "guide" tool which allows a programmer to places components on it.

Also, it creates a working program including skeleton functions that the programmer can modify to implement the behavior of the application.

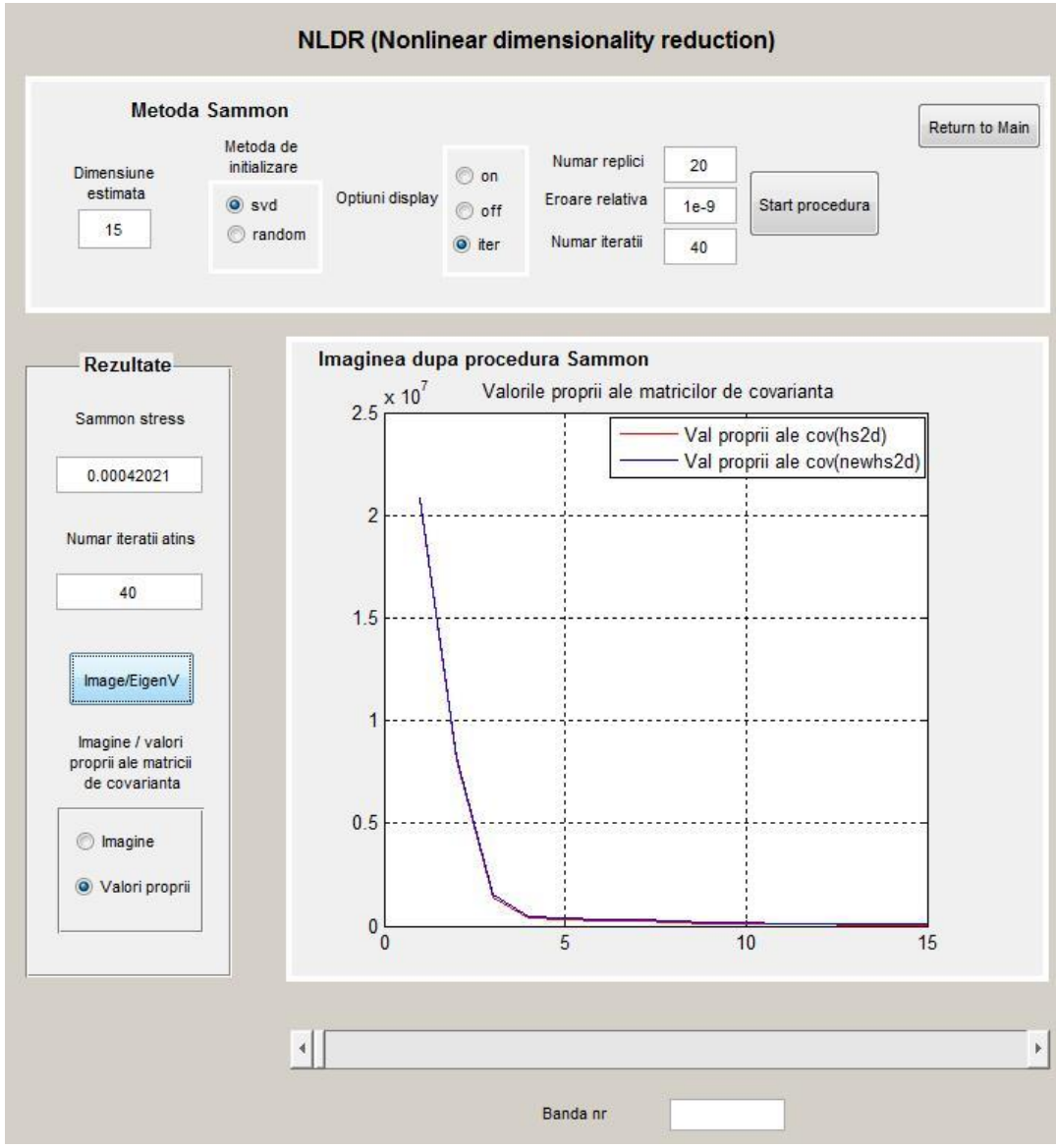

Fig. 2 Graphical User Interface for Sammon method (Eigen values after Sammon)

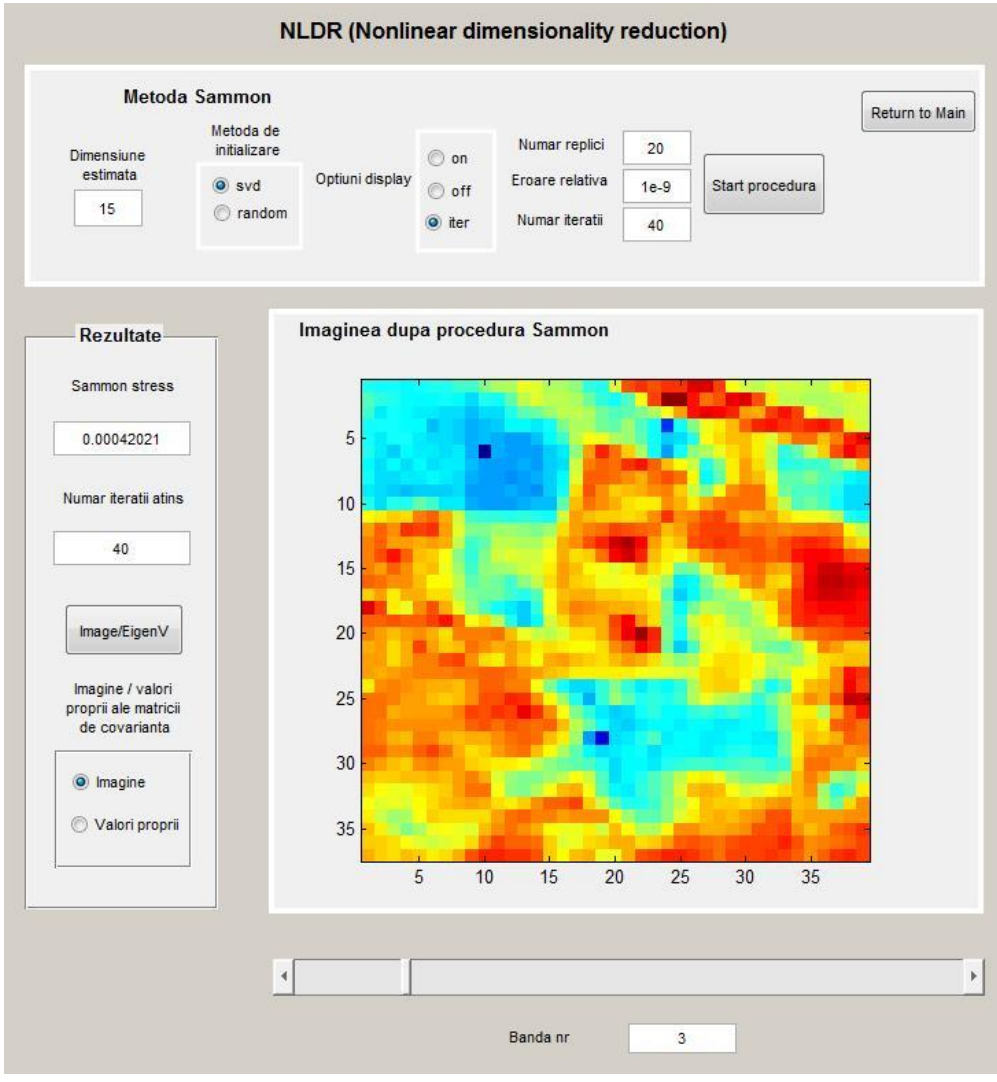

Fig. 3 Graphical User interface for Sammon method (an image after Sammon)

Table 1. The cumulative error of distances between points.

Iteration	Error	Iteration	Error
1	0.4987781290	100	0.1781135170
10	0.4399404276	150	0.1080544463
50	0.2364040024	199	0.0006019682

5. CONCLUSION

Using Sammon mapping on complex hyperspectral data, the user can explore the distribution of the objects, discover groupings of similar objects and detect outliers. More, the user can examine clusters produced by any clustering method and investigate the sensitivity of the clustering results to the parameters. For example the k-means procedure divides the set of observations into a user-specified number of subsets such that the dissimilarities within the subsets are minimized and the dissimilarities between the subsets are maximized.

REFERENCES

[1] C.-I Chang, *Hyperspectral Imaging: Techniques for Spectral Detection and Classification*, Kluwer/Plenum Acad. Publ., 2003.

[2] C.-I Chang and Q. Du, Estimation of Number of Spectrally Distinct Signal Sources in Hyperspectral Imagery, *IEEE Transactions on geoscience and remote sensing*, vol. **42**, no. 3, pp. 608-619, march 2004.

[3] K. Fukunaga and D. R. Olsen, An algorithm for finding intrinsic dimensionality of data, *IEEE Trans. on Computers*, C-20:176, 183, 1971.

[4] W. Xiong, C.-I Chang, C.-T. Tsai, *Estimation of virtual dimensionality in hyperspectral imagery by linear spectral mixture analysis*, Conference Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International, Honolulu, Pages 979-982, ISSN: 2153-6996, E-ISBN: 978-1-4244-9564-1, Print ISBN: 978-1-4244-9565-8, 2010/7/25.

[5] K. Cawse, A. Robin, M. Sears, *The effect of noise whitening on methods for determining the intrinsic dimension of a hyperspectral image*, DOI: 10.1109/WHISPERS.2011.6080974 Conference: Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS), 2011.

[6] K. C.-Nicholson, S. B. Damelin, A. Robin and M. Sears, Determining the intrinsic dimension of a hyperspectral Image using random matrix theory, *IEEE Transactions on Image Processing* 11/2012, **22**(4). DOI:10.1109/TIP.2012.2227765.

[7] Y. Tarabalka, J. A. Benediktsson, J. Chanussot and J. C. Tilton, Multiple Spectral–Spatial Classification Approach for Hyperspectral Data, *IEEE Transactions on Geoscience and Remote Sensing*, vol. **48**, no. 11, november 2010.

[8] J. Bioucas-Dias and J. Nascimento, Estimation of signal subspace on hyperspectral data, *Proc. SPIE*, vol. **5982**, pp.191-198, May 2005.

[9] C. C. Wu, W. Liu, and C.-I Chang, Exploration of methods for estimation of number of endmembers in hyperspectral imagery, *Proc.SPIE*, vol. **7**, nr.43, pp.1-11, 2006.

[10] K. Fukunaga, *Intrinsic dimensionality extraction, in Classification, Pattern Recognition and Reduction of Dimensionality*, Eds. Amsterdam 1982, vol. 2 Handbook of Statistics.

[11] J. Harsanyi, W. Farrand, and C. –I Chang, *Determining the number and identity of spectral endmembers: An integrated approach using Neyman-Pearson eigenthresholding and iterative constrained RMS error minimization*, Proc. 9th Thematic Conf. Geologic Remote Sensing, Feb. 1993.

[12] A. V. Little, J. Lee, Y.-Mo Jung, M. Maggioni, *Estimation of interinsic dimensionality of samples from noisy low-dimensional manifolds in high dimensions with multiscale SVD*, Department of Mathematics, Duke University, P.O.Box 90320, Durham, NC, 27708, USA, IEEE/SP 15th Workshop on Statistical Signal Processing, 2009.

[13] M. Fan, N. Gu, H. Qiao and B. Zhang. *Intrinsic dimension estimation of data by principal component analysis*, Feb. 2010.

[14] J. Yin, C. Gao, J. Sun, *Verification of statistical properties for hyperspectral images: heteroscedasticity and non-stationarity*, Conference: The Third International Conference on Digital Information Processing and Communications (ICDIPC2013), Publisher: The Society of Digital Information and Wireless Communication, 2013, pages 191-195, School of Astronautics, Beihang University, Beijing 100191, China.

[15] B. Balasko, J. Abonyi and B. Feil, *Fuzzy Clustering and Data Analysis Toolbox For Use with Matlab*, Department of Process Engineering University of Veszprem, Hungary, 2005.

[16] J. W. Sammon, Jr., A Nonlinear Mapping for Data Structure Analysis, *IEEE Transactions on Computers*, vol. **C-18**, no. 5, pp 401-409, May 1969.

[17] R. Karbauskaitė, G. Dzemyda, *Multidimensional data projection algorithms saving calculations of distances*, Information Technology and Control, ISSN 1392 – 124X, Vol. **35**, No. 1, 2006, Institute of Mathematics and Informatics Akademijos St. 4, 08663 Vilnius, Lithuania.

[18] M. Seul, L. O'Gorman, M. J. Sammon, *Practical Algorithms for Image Analysis: Description, Examples, and Code*, ISBN:0 521 66065 3, Cambridge University Press, 2000.