ELSEVIER

Computers & Geosciences 31 (2005) 689–697

COMPUTERS & GEOSCIENCES

www.elsevier.com/locate/cageo

Dimensionality reduction and visualization of geoscientific images via locally linear embedding☆

Fabio Boschetti*

CSIRO Exploration & Mining, PO Box 1130, Bentley WA 6102, Australia

Received 7 July 2004; received in revised form 21 October 2004; accepted 21 October 2004

Abstract

The locally linear embedding (LLE) algorithm is useful for analyzing sets of very different geoscientific images, ranging from smooth potential field images, to sharp outputs from modeling fracturing and fluid flows via cellular automata, to hand sketches of geological sections. LLE maps the very high-dimensional space embedding the images into 2-D, arranging the images on a plane. This arrangement highlights basic relationships between the features contained in the images, thereby greatly simplifying the visual inspection of the entire data set. Other applications include image classification, and visualization of the results of inverse modeling of geological problems in order to characterize domains of different mechanical behavior.

Keywords: Image processing; Visualisation; Data mining; Image classification

1. Introduction

The image processing and machine learning communities have long addressed the problems involved in the analysis of large multi-dimensional data sets. Applications range from data storage and compression, to data interpretation, data mining, and image retrieval and classification, to name a few. One approach, called dimensionality reduction, involves the design of algorithms to map images (or data) into spaces of much lower dimensionality than the space originally embedding the images. The rationale is that analysis and computation in such reduced dimensionality is considerably simplified. 'Classic' approaches to the problem, involving principal component analysis (Jolliffe, 1986) and multi-dimensional scaling (Cox and Cox, 1994), assume linear relationships between data, which may not work well on some complex applications. In recent years, this field of research has undergone renewed activity, mostly galvanized by the publication of two novel techniques that try to overcome the limitations of fully linear approaches: Isomap (Tenenbaum et al., 2000; Balasubramanian et al., 2002) and locally linear embedding (LLE) (Roweis and Saul, 2000; Saul and Roweis, 2002; Kouropteva et al., 2002). Mathematicians tend to be very reluctant to give up assumptions of linearity, since it is very convenient both from an analytical and computational perspective. So, when fully linear methods fail, the next option involves methods that are locally linear, that is, linear within closed regions of the original space. This applies to both

☆A Matlab implementation of the Locally Linear Embedding algorithm is publicly available at http://www.cs.toronto.edu/~roweis/lle/code.html.

*Corresponding author. Computational Geoscience, CSIRO Exploration; Mining, ARRC, 26 Dick Perry Avenue, Kensington, WA 6151, Australia. Tel.: +61 8 6436 8621; fax: +61 8 6436 8555.

E-mail address: fabio.boschetti@csiro.au (F. Boschetti).

0098-3004/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cageo.2004.10.002

Isomap and LLE. Between the two, LLE has desirable analytical and computational properties, like proof of convergence, fast implementation, and the ability to process new data without rerunning the overall algorithm. These features are very important when applications involve very large data sets, and make the use of LLE very attractive.

The following work tests the use of LLE on a number of very different geoscientific image data sets. The aims are fast image classification, and visualization of the results of inverse modeling. After explaining the basic concepts behind the approach, and the details of the algorithm, LLE is applied to data sets containing potential field images, cellular automaton modeling results for coupled mechanical and fluid flow in the Earth's crust, and hand sketches of geological sections. The nature of the images and the features contained in these data sets vary considerably. Their only commonality lies in being very broadly and weakly defined as 'geoscientific' data. The performance of LLE on such disparate images is particularly encouraging. Like all complex techniques, LLE should not be used as a 'black box', and the discussion below introduces its limitations and the two important tuning parameters. Finally, this is a field in very active development, and suggestions for future work are included.

2. Motivation

Fig. 1 contains a gravity image with a resolution of 64 pixels in both the horizontal and vertical directions. The image processing 'algorithm' hardwired into a human brain allows one to recognize 'an object' (in this case, a single gravity anomaly) roughly in the center of the image. However, when processed by any computer algorithm, the image is represented by an array of $64 \times 64 = 4096$ values. In other words, the image is a mathematical object (a point) embedded in a space of 4096 dimensions. Fig. 2 shows an array of 36 images, each of them consisting of 64×64 pixels. The set of images in this case can be seen as 36 points in a 4096-dimensional space. The human brain, once again, is able to extract a far simpler representation of the set in Fig. 2. The images are characterized by the gravity anomaly changing position smoothly along the horizontal and vertical axes. Accordingly, the set of images could be described by only two parameters, that is, by the horizontal and vertical positions of the anomaly. In other words, for the human brain, the set of images belongs to a 2-D space: a considerable dimensionality reduction from the original 4096 dimensions.

The problem of dimensionality reduction can be summarized as follows: devise an algorithm capable of mapping data from a high-dimensional space to a (much) lower-dimensional one, in such a way that basic relationships in the data are respected, and the data interpretation in such a lower-dimensional space is considerably simplified. Following the example above, the task is to map the 36 images in Fig. 2 into a much lower-dimensional space, in which the geometrical relationships between the gravity anomaly positions become obvious.

The assumption underlying this approach is that images arising from a physical process (e.g., gravitational attraction law), as a function of a relatively small number of varying parameters (e.g., horizontal and vertical position of the source of the gravity signal), lie on a manifold (surface) of much lower dimension than the embedding space. The dimensionality reduction problem can then be viewed as identifying such a

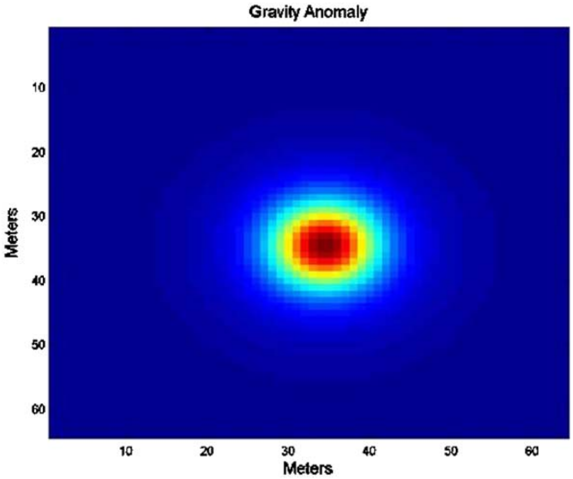

Fig. 1. Single gravity anomaly.

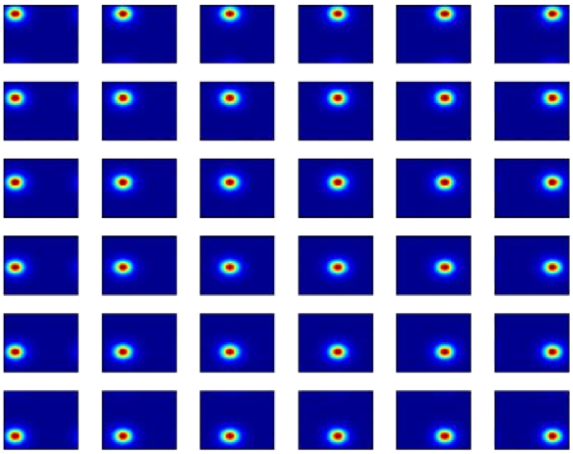

Fig. 2. Array of gravity images, with anomaly changing position in both X and Y.

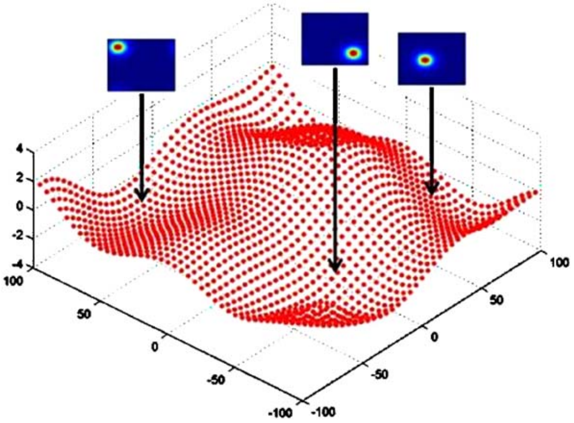

Fig. 3. Simple sketch of a manifold embedded into a higher-dimensional space. Location of each image (red dots) in a hypothetical N-D space (3-D in this case) defines a lower-dimensional manifold. Positions on manifold can be defined by a 2-D coordinate system.

manifold, and representing the data with local coordinates within the manifold itself (Fig. 3 gives a simple sketch of the concept). Kouropteva et al. (2002) and Saul and Roweis (2002) show that such an assumption holds for a number of image databases. Classic examples are photographs of human faces in different poses and under different camera orientations and ambient lighting, hand-written numerals, and movement of human lips for speech recognition. The examples below test the validity of this assumption for a number of geoscientific data sets.

3. The locally linear embedding algorithm

The idea underlying the LLE approach is very simple, and its description will considerably help understand the algorithm's implementation. LLE finds local representations by expressing each data point (image) as a linear combination of neighboring points (images). As an example, in Fig. 4a, we have 5 points in a 3-D space. We express the point in the center as a linear combination of the 4 surrounding points. The weights of the linear combination for each point become the local coordinate system for the representation. LLE then maps points into a lower-dimensional space in such a way that the new coordinates of each point in the lower-dimensional space are a linear combination of the same neighboring points, with the same weights as the ones calculated in the original embedding. We can see this in Fig. 4b. The five points are mapped into a 2-D space, in such a fashion that the central point is a linear combination of the mapping of the same 4 surrounding points, with the same weights. The result is a mapping that respects local relationships between neighboring points. Linearity is imposed only locally, not on the overall, global mapping. Consequently, LLE is expected to map curved manifolds in an acceptable approximation (for an example of application to curved manifolds the reader is referred to the 'swiss roll' benchmark case in Saul and Roweis, 2002).

The following brief description of the LLE algorithm is complemented by details in Roweis and Saul (2000), Saul and Roweis (2002), and Kouropteva et al. (2002). Assuming m points embedded in an N-D space with coordinates X, LLE seeks to map the points to an n-D space with coordinates x, with $n<N$. The algorithm can be divided into three stages.

(1) the neighborhood for each point. For a point i, the neighborhood $\vec{X}_i^j, j = 1, \ldots, K$ can then be defined either as the set of K closest points or as the set of points within a certain radius. Standard implementations use the first option.

(2) For each point, determine the weights W_i that allow the point to be represented as a linear combination of the K points in the neighborhood. This involves minimizing the cost function

$$C(W) = \sum_{i=1,\ldots,m} \left| \vec{X}_i - \sum_{j=1,\ldots,K} W_{i,j} \vec{X}_i^j \right|^2, \tag{1}$$

where $i = 1 \ldots m$ represent the points in the original N-D space and $j = 1, \ldots, K$ are the neighboring points. The minimization of Eq. (1) is carried out under the constraints

$$\sum_{j=1,\ldots,K} W_{i,j} = 1 \text{ and } W_{i,j} = 0 \text{ if } X_i \text{ is not a neighbor of } X_j. \tag{2}$$

The solution to Eq. (1), under the constraint of Eq. (2), is invariant to rotation, rescaling, and translation (Saul and Roweis, 2002, p. 124, 131). This is a crucial feature of the LLE algorithm. It means that the weights W_i do not depend on the local frame of reference. They represent local relationships between data points, expressed in a frame of reference that is valid globally.

(3) Map the m points into the lower-dimensional space with coordinates x_i. This is achieved by minimizing the cost function

$$E(x) = \sum_{i=1,\ldots,m} \left| \vec{x}_i - \sum_{j=1,\ldots,K} W_{i,j} \vec{x}_i^j \right|^2 \tag{3}$$

(where the weights W_i obtained from Eq. (1) and (2) are kept fixed, conserving the local relationship between neighboring points) under the constraint

$$\sum_{i=1,\ldots,n} \vec{x}_i = \vec{0}. \tag{4}$$

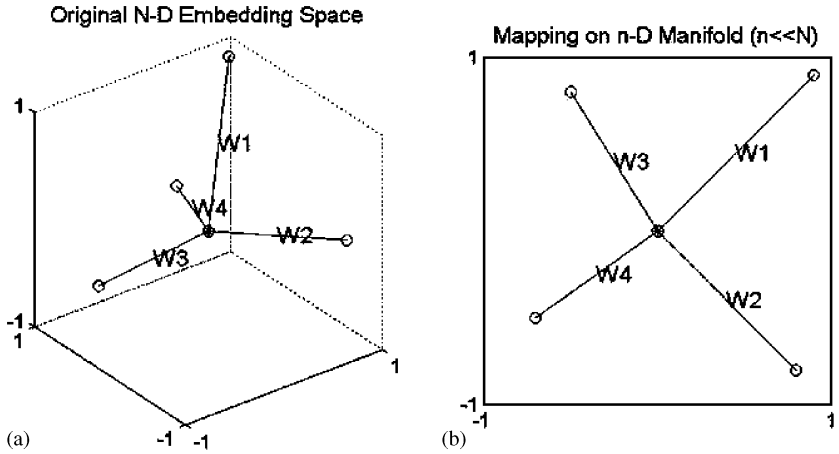

Fig. 4. Basic concept behind LLE approach. In original N-D embedding space, a point (black, in center) is expressed as a linear combination of neighboring points (four, in this case). Weights W_i for linear combination are stored. (b) Point is mapped into a lower-dimensional manifold (n-D, with $n \ll N$), in such a way that it can be expressed as a linear combination of same neighbors, with weights approximately equal to original W_i.

This centers the x coordinate around zero by removing translational invariance, and

$$\frac{1}{m}\sum_{i=1,\dots,n} \vec{x}_i\vec{x}_i^{\mathrm{T}} = 1 \tag{5}$$

removes rotational invariance (see Saul and Roweis, 2002, p. 134).

This equates to finding some global coordinates x_i, over the lower-dimensional manifold, that conserve the local relations between neighboring points in the original embedding space. Each individual coordinate is obtained only from local information within its neighborhood. Importantly, the overlapping of each neighborhood generates the global reference. Solving Eq. (3) is the most delicate and computationally intensive part of the LLE algorithm. Fortunately, it can be achieved without an iterative scheme. Via the Rayleitz–Ritz theorem (Horn and Johnson, 1990), this reduces to finding n eigenvectors of the matrix

$$M_{i,j} = \partial_{i,j} - W_{i,j} - W_{j,i} + \sum_{K} W_{k,i}W_{j,k}. \tag{6}$$

The n eigenvectors correspond to the n smallest non-zero eigenvalues of M (Saul and Roweis, 2002, p. 134–135).

The overall LLE algorithm only involves searching for closest points and basic matrix manipulations, which can easily be implemented via standard computational tools (Matlab, Numerical Recipes, LAPACK, Mathematica, etc.). A Matlab version of the LLE algorithm, which follows closely Saul and Roweis (2002) implementation is publicly available at http://www.cs.toronto.edu/~roweis/lle/code.html.

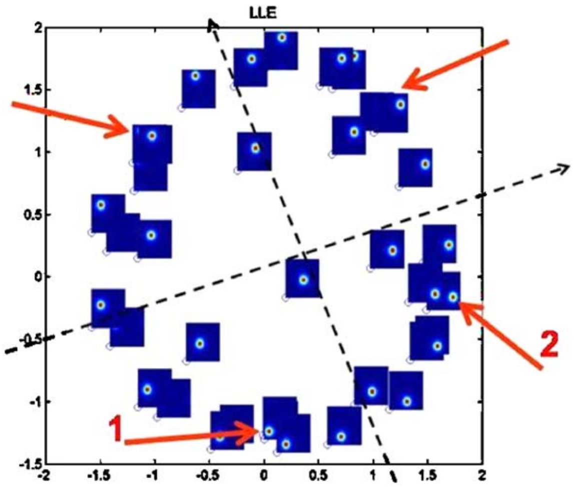

Fig. 5. LLE mapping of 36 images in Fig. 2. Red arrows indicate four corner images. Images with anomaly in center are located approximately in center of mapping. Dashed lines show approximate axes of 2-D mapping.

4. Applications

4.1. Potential field data

The first test maps the 36 images in Fig. 2 onto a 2-D plane (Fig. 5). The image with the anomaly in the center is correctly located close to the center of Fig. 5. The red arrows show the locations of the four corner images, that is, the images in which the anomaly is located close to the image corners. The four images are uniformly spread on the plane. Following an imaginary line connecting Arrow 1 to Arrow 2, we see the gravity anomaly moving smoothly from the bottom left corner to the bottom right corner of the image. Similar smooth variations characterize all other transition between the arrows. LLE has reconstructed the overall approximate mapping, in 2-D, of the initial 4096-D images. The mapping is only approximate, due mostly to the sparsity of 36 samples in a 4096-dimensional space. Also, the Euclidean distance, calculated as the sum of squared differences between image pixels, does not necessarily capture the correct distance between the images as perceived by the human brain. Nevertheless, the overall structure of the images in the 2-D manifold, as characterized by the physical translation of the source of the gravity signal, is recovered.

Turning to a more challenging data set, Fig. 6 shows the LLE mapping of a set of synthetic gravity images produced from causative sources of different overall shape. The images illustrate the gravity response of classic 3-D geometries used in potential field interpretation (see Holden et al., 2000). Not only do the locations of the sources vary from image to image, but also the number of sources, their shapes and their depth, affect the intensity of the images. The images can no longer be expected to lie on a 2-D manifold, which is nevertheless the goal for visualization purposes. This inevitably imposes a further distortion in the ordering of the images' output coordinates. Despite the distortion, it is quite easy to recognize a meaningful structure in the LLE plot (Fig. 6). The images are ordered with horizontal anomalies grouped mostly to the left-hand side, and from bottom to top of the LLE mapping, these anomalies migrate from the lower part to the upper part of the gravity image. The more vertical anomalies are grouped to the right-hand side of the LLE mapping, with the circular anomalies correctly located near the center of the map.

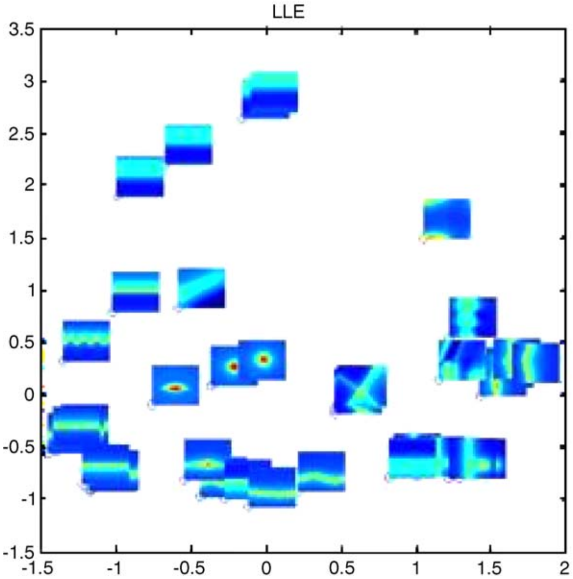

Fig. 6. LLE mapping of a set of synthetic gravity images. Images are grouped with horizontal anomalies to left-hand side of map, and from bottom to top of LLE mapping, anomaly migrates from lower part to upper part of gravity images. More vertical anomalies are grouped to right-hand side of LLE mapping, with circular anomalies correctly located near center.

4.2. Image classification

Image classification using LLE is potentially a very useful application in the geosciences. In a database containing several images (potential field images, seismic sections, geological sections, satellite images, etc.), call the image set X. Supposing a new image X^{new} is added, it should be classified according to its relationship/similarity to images already in the database. Also suppose that an LLE mapping of the data already exists. The new image can be incorporated in several ways. The most obvious is to include the new image in the database and rerun the LLE algorithm. For large databases, this would be inefficient, and, most importantly, not necessary. From Eq. (3), the calculation of the mapping in the low-dimensional n-D space depends only on the weights W_i. The weights carry information about the distribution of the X coordinates in the original embedding space, but those coordinates are not explicitly needed to solve Eq. (3). This suggests that, for this application, most of the computation involved in steps 1 and 2 of the LLE algorithm can be skipped.

Kouropteva et al. (2002) provide two ways to add new data. The first option involves (a) finding the closest point X^j to X^{new} in the database, (b) determining a mapping G between the N-D and n-D coordinates for the point X^j ($x^j = GX^j$), which is valid only locally, and (c) applying the same mapping to find the coordinates of the new image in n-D, $x^{new} = GX^{new}$. This approach relies on the new point being close to an existing point in the database. The second option involves (a) finding the K neighboring images to X^{new} in the database, (b) calculating the weight W^{new} corresponding to X^{new}, (c) adding W^{new} to the original W, and (d) calculating the new coordinates for all points in n-D by applying the full step 3 of the LLE algorithm. Clearly, option 2 is more computationally expensive (and more accurate) than option 1. The advantage of option 2 compared to rerunning the full LLE algorithm is relatively small, since, as explained above, the most computationally costly part of the LLE algorithm is solving Eq. (3) in step 3.

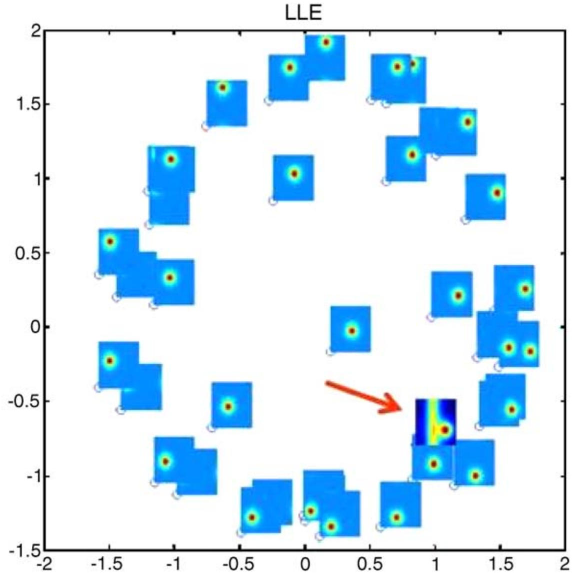

Fig. 7. Image at bottom left of Fig. 5 (indicated by red arrow) is mapped within images in Fig. 2. LLE can be used for quick image recognition.

Consequently, the following, third option is preferable. Steps (a) and (b) are as in the second option above, but the calculation of the coordinates x^{new} is done by applying the W^{new} to the neighborhood $Neig_{x^{new}}$ which is the *n-D* correspondent of the neighborhood of X^{new} only, i.e., $x^{new} = W^{new} Neig_{x^{new}}$. This is equivalent to calculating x^{new} while leaving all other x coordinates untouched, and avoids solving Eq. (3) again, with considerable computational efficiency.

The outcome of this option is illustrated in Fig. 7, by taking one of the images from Fig. 6 and mapping it onto the LLE plot in Fig. 5 (the image indicated by the red arrow in Fig. 7). Despite the new image containing more than a single source (isolated circular anomaly plus a linear anomaly), the LLE finds the correct similarity, and the new image is plotted in the correct position.

4.3. Visualization of the results of inverse modeling

In Boschetti and Moresi (2001) and Wijns et al. (2003a), numerical inversion is proposed as a method to recover the initial conditions for mechanical models that result in certain geological patterns. Boschetti et al. (2002) propose that visualizing the simulations generated via numerical inversion be an integral part of the inverse approach. The aim is not only to recover initial conditions generating certain geological scenarios, but also to understand the effect of parameters in the physical processes, with the aim of partitioning the parameter space into regions of markedly different mechanical behavior. Within this framework, visualization of high-dimensional space is essential for the user to interact with, and comprehend, the progress of the inversion. Boschetti et al. (2002) use a self-organizing map (Kohonen, 2001) for visualization, while Wijns et al. (2003b) employ a Sammon's map (Sammon, 1969). Here, the LLE algorithm is trialed.

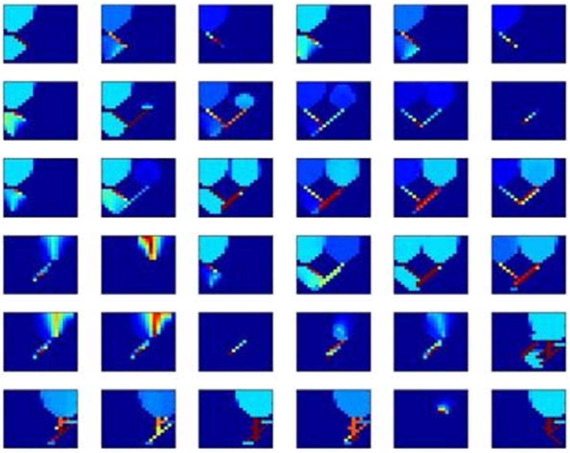

Fig. 8. Simulations from cellular automaton modeling of coupled fluid flow and fracturing. Images are result of mixing two fluids injected from two sources at different locations along bottom boundary of model.

Fig. 8 shows 36 simulations obtained by running a cellular automaton (CA) model for fracturing and fluid flow in a cracked medium (from Boschetti, 2005). Each simulation depicts a 2-D vertical section of the Earth's crust. Two different fluids are injected under pressure from the bottom boundary of the model. Some CA nodes are fractured (not shown), while others are not; fluid can flow in fractured nodes, or can cause fracture in nodes if the fluid pressure exceeds a certain threshold. The presence of fluid weakens a node, which can consequently further crack and allow more fluid to flow, and, similarly, the removal of fluid can strengthen a node, decreasing the amount of fluid that can subsequently flow. When the two fluids come into contact, they react and generate a third fluid; the outcome of the CA is the path of deposition from the third fluid. The presence of the underlying crack patterns makes the problem strongly non-linear. The purpose of the inverse problem is to determine the location and amount of fluid for each source, given a final depositional pattern. Further details of the implementation, and results of the inversion, are in Boschetti (2005).

In the LLE plot of the CA simulations (Fig. 9), as in Fig. 5, two axes are overlaid (dashed lines) to show the direction of clear variation between the simulations. Along the pseudo-vertical axis (pointing toward the top left), images show deposition on a single side of the CA model (deposition on the left at the top of the axis and

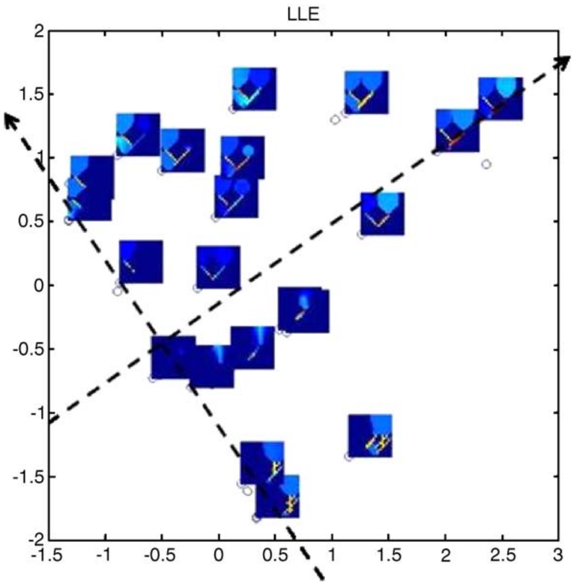

Fig. 9. LLE plot for CA simulations in Fig. 8.

toward the right at the bottom). Along the other axis, simulations tend to vary from a single depositional pattern on the left to two depositional patterns on the right-hand side. Once again, the LLE has found a rough structure among the 36 high-dimensional images, and respected this structure in the 2-D representation.

4.4. Hand-drawn sketches of geological sections

The final test is the most challenging. LLE is applied to a set of black and white images representing hand-drawn sketches of geological sections, containing combinations of faults, flat and folded layers, and isolated blocks in different geometrical relationships. The sketches were generated to train a machine learning algorithm to mimic the way a geologist evaluates similarity between different geological scenarios (Kaltwasser et al., 2004). Each of the 63 sketches in Fig. 10 consists of 20 × 100 pixels, resulting in a 2000-dimensional space. Fig. 11 shows the LLE plot. To facilitate visual analysis, only some images are displayed, but the black circles mark the locations of all 63 simulations. This is a difficult test, mostly because the black and white images contain only sharply bounded objects. Sharply bounded objects result in distances that do not change smoothly between simulations. This makes the identification of the neighborhoods less stable. Nevertheless, a satisfactory mapping is obtained. The simulations with folds of high frequency and high amplitude are mapped at the top of the LLE plot. Simulations with no folds and mostly isolated blocks are mapped in the middle of the plot. Simulations with short folds occupying only the left-hand side of the geological section are mapped in the lower part of the LLE plot. Also, the orientations of the two faults change from the bottom of the plot (faults opening upward) to the top (faults opening downward).

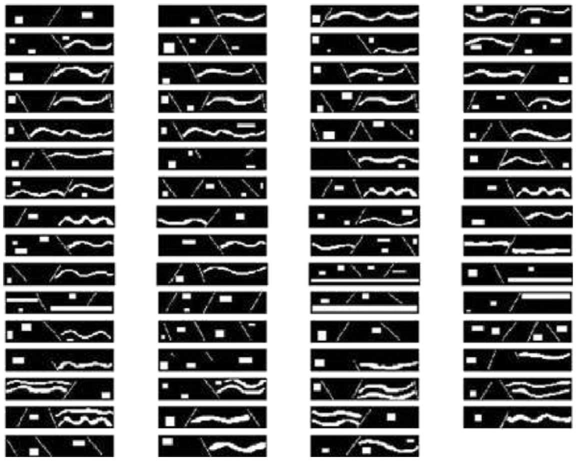

Fig. 10. Hand sketches of 63 geological sections containing different combinations of faults (thin pseudo-vertical lines), flat and folded layers, and isolated bodies.

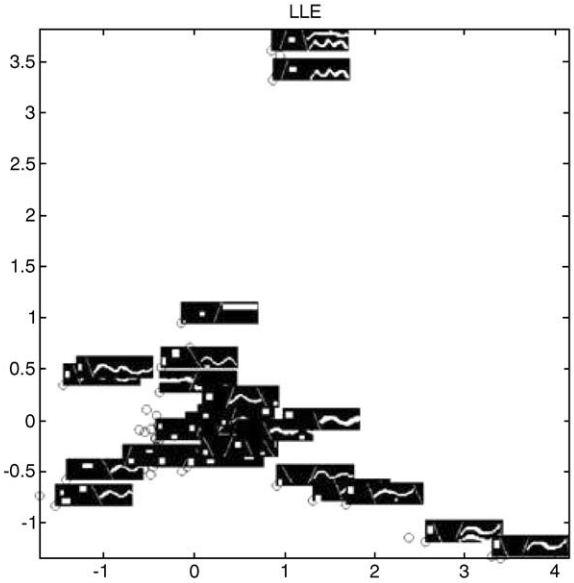

Fig. 11. LLE plot of 63 hand-sketched geological sections.

5. Discussion, limitations, and direction for further work

The LLE algorithm is controlled by the two parameters n and K, the dimensionality of the space into which simulations are mapped, and the size of the local neighborhood, respectively. In the applications above, as for most LLE applications, $n = 2$, since the aim is visualization on a computer screen. The assumption is that the human brain can more easily recognize relationships between data in a familiar 2-D setting than in higher-dimensional ones. For other applications, reconstructing the 'correct' n, that is, the correct dimensionality of the manifold in which the data lie, may be useful. Brand (2002) and Kégl (2002) propose several approaches to the problem, which are based on variations of classical fractal dimension measures. Both methods rely on large volumes of data, which are rarely available in geoscientific problems, and further research is needed before being able to reliably estimate the correct, or optimal n underlying problems like the ones described above.

Kouropteva et al. (2002) propose a method to determine the optimal size of the neighborhood K. It is based on minimizing the standard linear correlation coefficient (see Press et al., 2002, p. 630) between the matrices containing pair-wise distances between points in the original N-D and the output n-D spaces. The optimal K is chosen as the K for which the highest correlation is found between distances in the original and output spaces. The method is easy to implement and conceptually intuitive, since it basically measures the success of the LLE algorithm itself. In all the examples presented above, K is chosen according to this principle. The main drawback is that it requires running the LLE algorithm a number of times for different Ks. Both experience, and results from the literature, suggest that LLE performance is particularly sensitive to the value of K, and the increase in computational time involved in its proper choice seems to be, at this stage, unavoidable.

In general, it seems that LLE can be a very useful tool for the inspection of large sets of geoscientific images, but, as for many other algorithms, it should not be used in a 'black box' fashion. The assumptions behind the algorithm need to be understood in order to evaluate how well they apply to the data set under analysis. For example, it is important to remember that the algorithm is based on the reconstruction of each point as a linear combination of its neighbors. Whether this locally linear reconstruction is valid depends on the nature of the problem, but also on the number of points available for analysis. In a high-dimensional problem with very sparse sampling, the size of the neighborhood (implied by the choice of K) of a point may exceed the range within which linear approximation can be accepted. Consequently, it is very useful to directly assess the quality of the local linear reconstruction. Once the weights W_i have been calculated, each point (image) can be reconstructed as a linear combination of its neighboring points (images), and the departure of the reconstruction from the original point (image) can be evaluated, or, even better, visualized. This adds only limited computational effort (recalling that by far the greatest expenditure is in solving Eq. (3) in step 3 of the algorithm), and it is effort well spent, since a bad reconstruction will prevent good mapping.

The above problems are minor compared to those of alternative techniques. The SOM and Sammon's map involve a much larger computational effort, and cannot guarantee an optimal mapping, due mostly to their iterative algorithms. Consequently, running a SOM or Sammon's map several times on the same data set will result in several different maps. Assessing the quality of the maps is then more difficult than in the case of LLE. From this perspective, LLE offers a considerable improvement.

A number of avenues for future research are available and worthwhile addressing. A first step is to test variants of the LLE method that have been recently proposed, such as Laplacian LLE (Belkin and Niyogi, 2003) and Hessian LLE (Donoho and Grimes, 2003). These are based on projecting local neighboring points onto a subspace tangent to each point before proceeding with the analysis. A second task is to compare the LLE algorithm to Isomap (Tenenbaum et al., 2000), which works by building a graph containing the data points and analyzing data relations within the graph. Probably more relevant to a geological application is to supply such schemes with better measures of distance between images. All algorithms mentioned above (as well as more standard algorithms like SOM and MDS) work on Euclidean distances. Euclidean distances may not necessarily represent concepts like geological similarity, which involve complex geometrical and temporal relationships between the features contained in an image. First attempts to address this problem (Kaltwasser et al., 2004) have barely touched the complexity behind the issue. Current LLE implementations already allow pair-wise distances to be provided as input. A better understanding of what constitutes geological similarity, and the development of a robust measure of it, would greatly improve LLE performance and make it even more relevant to geological applications.

6. Conclusions

Dimensionality reduction algorithms in the family of local linear embedding can be very useful in the analysis of large databases of geoscientific images, by highlighting basic relationships between the main features contained in the images. The considerable advantages offered, compared to more traditional approaches like PCA, SOM, or MDS, include local linear approximations (versus globally linear ones), a fast algorithm, and a guaranteed optimal mapping given a set of input parameters. The performance of the LLE algorithm on data sets containing very different kinds of images is encouraging, and suggests that further work on similar algorithms, such as Laplacian or Hessian LLE and Isomap, is particularly worthwhile.

Acknowledgements

I would like to thank Darren Holden of Fractal Technologies for providing the potential field images used in Fig. 6.

References

Balasubramanian, M., Schwartz, E.L., Tenenbaum, J.B., de Silva, V., Langford, J.C., 2002. The isomap algorithm and topological stability. Science 295(5552), 7a.

Belkin, M., Niyogi, P., 2003. Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation 15 (6), 1373–1396.

Boschetti, F., 2005. Controlling and investigating cellular automata behavior via interactive inversion and visualization of search space. New Generation Computing, Special Issues on Intertactive Evolutionary Computation, Vol. 23, No. 2, February 2005 (in print).

Boschetti, F., Moresi, L., 2001. Interactive inversion in geosciences. Geophysics 64, 1226–1235.

Boschetti, F., Wijns, C., Moresi, L., 2002. Effective exploration and visualisation of geological parameter space. G Cubed 4 (10), 1086.

Brand, M., 2002. Charting a manifold. Neural Information Processing Systems 15.

Cox, T., Cox, M., 1994. Multidimensional Scaling. Chapman & Hall, London 213 pp.

Donoho, D.L., Grimes, C.E., 2003. Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. Proceedings of the National Academy of Arts and Sciences 100, 5591–5596.

Holden, D., Archibald, N., Boschetti, F., Jessell, M., 2000. Inferring geological structures using wavelet-based multiscale edge analysis and forward models. Exploration Geophysics 31, 617–621.

Jolliffe, I.T., 1986. Principal Component Analysis. Springer, New York, 271 pp.

Kaltwasser, P., Boschetti, F., Hornby, P., 2004. Measure of similarity between geological sections accounting for subjective criteria, Computers & Geosciences, in press.

Kégl, B., 2002. Intrinsic dimension estimation using packing numbers. Neural Information Processing Systems 15.

Kohonen, T., 2001. Self-Organizing Maps, third ed. Springer, New York, NY, 501 pp.

Kouropteva, O., Okun, O., Hadid, A., Soriano, M., Marcos, S., Pietikäinen, M., 2002. Beyond locally linear embedding algorithm. Technical Report MVG-01-2002, University of Oulu, Machine Vision Group, Information Processing Laboratory, 49 pp

Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T., 2002. Numerical Recipes in Fortran. Cambridge University Press, New York, NY, 963 pp.

Roweis, S., Saul, L., 2000. Nonlinear dimensionality reduction by locally linear embedding. Science 290, 2323–2326.

Sammon Jr., J.W., 1969. A nonlinear mapping for data structure analysis. IEEE Transactions on Computers C 18, 401–409.

Saul, L., Roweis, S., 2002. Think globally, fit locally: unsupervised learning of nonlinear manifolds. Technical Report MS CIS-02-18, University of Pennsylvania, 37 pp

Tenenbaum, J.B., de Silva, V., Langford, J.C., 2000. A global geometric framework for nonlinear dimensionality reduction. Science 290, 2319–2323.

Wijns, C., Boschetti, F., Moresi, L., 2003a. Inverse modelling in geology by interactive evolutionary computation. Journal of Structural Geology 25 (10), 1615–1621.

Wijns, C., Poulet, T., Boschetti, F., Griffiths, C., Dyt, C., 2003b. Interactive inverse methodology applied to stratigraphic forward modeling, geological prior information: value and quantification. In: Curtis, A., Wood, R. (Eds.), Geological Prior Information. Geological Society of London Special Publication.