

Seismic graph analysis to aid seismic interpretation

Rodrigo S. Ferreira¹, Emilio Vital Brazil¹, Reinaldo Silva¹, and Renato Cerqueira¹

Abstract

During the seismic interpretation process, geoscientists rely on their experience and visual analysis to assess the similarity between seismic sections. However, evaluating all of the seismic sections in a 3D survey can be a time-consuming task. When interpreters are working on a data set, a common procedure is to divide the cube in increasingly finer grids until they are satisfied with the result of the interpretation. We have developed a method based on graph theory and image texture in which we represent a seismic data set as a complete weighted undirected graph — which we call a seismic graph. The vertices of this graph represent the seismic sections, and the weight of the edges represents the distance between the texture feature vectors of the vertices they connect, allowing for a powerful yet concise representation of potentially large data sets. We have investigated the potential of graph analysis to build an adaptive grid that is more likely to capture the underlying structures present in a survey, providing a tool for a faster and more precise interpretation. The main idea is that such a grid would be finer in regions with more geologic variations and coarser otherwise. To demonstrate the capabilities of our technique, we apply it on a public data set called Netherlands F3. Using our method, we suggest which seismic sections — key sections — should be considered in the interpretation process. The results of our experiments indicate that our methodology has great potential to aid the seismic interpretation process.

Introduction

The seismic reflection method is known for its ability to provide essential knowledge about the subsurface. Besides providing stratigraphic and structural information, the seismic method, along with other sources of data, helps in the estimation of rock physical properties and possible oil and gas reservoir locations. Although seismic data are paramount for the oil and gas industry, the interpretation of such data represents an increasing challenge for geoscientists, primarily because this procedure is a time-consuming and human-intensive task. In addition, interpreters must frequently deal with the continuous growth of seismic data sets and the demand for faster results on their daily workflow.

In this sense, there have been several works focused on providing new methods to assist geoscientists and optimize the interpretation process. Saraswat and Sen (2012) use an artificial immune (AI) system and self-organizing maps (SOMs) to perform automatic facies classification and facies map generation. With the application of the proposed algorithm, the authors provide an alternative to automatic unsupervised facies classification using only SOM, which is more sensitive to noise (Taner et al., 2001; de Matos et al., 2003).

Another example is the work of Roden et al. (2015). The authors use principal component analysis (PCA) to determine which seismic attributes in a given set, or a

combination of those, have interpretative significance. They also apply SOM to produce meaningful and easily interpretable results for the seismic attributes that result from the PCA.

Long et al. (2018) investigate typical texture attributes for the labeling of four classes — chaotic, faults, salt dome, and other — in seismic volumes. The authors extract gray-level co-occurrence matrix (GLCM) features, semblance, local binary patterns (LBPs), local radius index (LRI), among other texture features, and use them as input for a support vector machine classifier. Such a workflow could be used to highlight structures of interest and speed up the interpretation process.

Wang et al. (2018) and AlRegib et al. (2018) provide a thorough review of the most recent methods for seismic structural interpretation. The authors argue that by treating seismic data as images rather than signal traces, researchers have been able to apply modern machine and deep-learning techniques to aid seismic interpretation. They discuss methods such as gradient of textures, GLCM features, convolutional neural networks, among others, for the detection of faults, salt domes, channels, and gas chimneys.

The recent work of Amin and Deriche (2016) presents a supervised codebook-based learning model for salt dome detection in seismic images. The proposed algorithm applies a combination of GLCM features and

¹IBM Research, Rio de Janeiro, Brazil. E-mail: rosife@br.ibm.com; evital@br.ibm.com; rmozart@br.ibm.com; rcerq@br.ibm.com.

Manuscript received by the Editor 28 October 2018; revised manuscript received 22 February 2019; published ahead of production 08 April 2019; published online 28 May 2019. This paper appears in *Interpretation*, Vol. 7, No. 3 (August 2019); p. SE81–SE92, 12 FIGS.

<http://dx.doi.org/10.1190/INT-2018-0185.1>. © 2019 Society of Exploration Geophysicists and American Association of Petroleum Geologists. All rights reserved.

Gabor filters to delineate salt boundaries in seismic data. A similar approach is investigated by Ferreira et al. (2016) whose main contribution is to evaluate several texture descriptors in a multiscale scheme. In the same context, Shafiq et al. (2018) propose a workflow based on saliency detection for the delineation of salt dome boundaries.

More focused on texture features, Long et al. (2015) evaluate several texture features such as steerable pyramid, curvelet features, LBP, LRI, and seismic similarity (SeiSIM) for the retrieval of seismic structures in migrated volumes. SeiSIM produced the best performance for the retrieval of classes clear horizon, chaotic horizon, faults, and salt domes. Similarly, Zujovic et al. (2009), Alfarraj et al. (2016), and Mattos et al. (2017) evaluate similar texture attributes, such as LBP, structural similarity (SSIM), and mean squared error (MSE), for the retrieval of texture images in public databases.

Although the list of works that focus on aiding the seismic interpretation procedure is extensive, to the best of our knowledge, this is the first work to propose an automatic extraction of the most representative sections in a 3D seismic data set. Extending the work of Ferreira et al. (2018), we propose a method based on the texture analysis and graph theory, which can automatically compute the similarity between seismic sections and suggest which seismic sections — called key sections — should be considered in the interpretation process.

We investigate GLCM features, LBP, SSIM, and MSE in a workflow that uses texture features to compute the similarity between seismic sections. This information is then used to build a graph over which a clustering is performed to select the most representative seismic sections. During this analysis, we consider two clustering algorithms: agglomerative clustering (Mullner, 2011) and spectral clustering (Von Luxburg, 2007).

The method may accelerate the interpretation by presenting the interpreter with a more concise representation of the seismic volume, in which areas that present more variations in strata geometry will be thoroughly scanned through a finer grid, whereas areas with fewer variations will be analyzed using a coarser grid. We evaluate the proposed methodology in the Netherlands F3 public seismic data set.

The remainder of this paper is organized as follows. First, we present the distance measures and clustering algorithms that will be investigated in this work. Then, we present our methodology, discussing how the seismic graph is built and key sections are extracted. Afterward, we describe our experiments and examine the results using the selected data set. Finally, we draw some conclusions and present our final remarks in the last section.

Distance measures

In this section, we present the distance/similarity measures to be investigated in this work. Two distance measures are based on texture attributes — GLCM and LBP. We also consider SSIM, which is based on low-

level local statistics in the spatial domain, and MSE, which is based on a pixel-wise comparison.

GLCM attributes

GLCM texture attributes were originally proposed by Haralick and Shanmugam (1973). For each analysis window, the algorithm creates a set of matrices $L \times L$, where L is the number of gray levels in the image. Each matrix $C_{d\theta}[i,j]$ is defined by two parameters, the angle θ and the distance d . Common values for the angles are 0° , 45° , 90° , and 135° . Each element (i,j) in these matrices represents the number of occurrences of gray level i adjacent to gray level j separated by distance d in direction θ . These matrices are usually normalized so that the sum of its elements is equal to one, which can be regarded as joint probability matrices $P_{d\theta}[i,j]$. Based on these matrices, several attributes are computed:

$$\text{contrast} = \sum_i \sum_j |i - j|^2 P_{ij}, \quad (1)$$

$$\text{entropy} = -\sum_i \sum_j P_{ij} \log P_{ij}, \quad (2)$$

$$\text{energy} = \left[\sum_i \sum_j P_{ij}^2 \right]^{\frac{1}{2}}, \quad (3)$$

$$\text{homogeneity} = \sum_i \sum_j \frac{1}{1 + (i - j)^2} P_{ij}, \quad (4)$$

$$\text{correlation} = \sum_i \sum_j \frac{(i - \mu_i)(j - \mu_j)P_{ij}}{\sigma_i \sigma_j}, \quad (5)$$

$$\text{dissimilarity} = \sum_i \sum_j |i - j| P_{ij}, \quad (6)$$

$$\text{inverse difference} = \sum_i \sum_j \frac{1}{1 + (i - j)^2} P_{ij}. \quad (7)$$

The number of features in the final feature vector is given by the product of the number of attributes and the number of matrices. GLCM texture attributes are computed over a single input image, and the distance between two seismic images is computed using the

Euclidean distance. It is noteworthy that other metrics such as Manhattan (L1) and cosine distances could also be used.

LBP

LBP is a technique that encodes the information about the neighborhood of a pixel into a binary number (Ojala et al., 2002). Given a radius $R \geq 1$ and several neighbors P , the neighborhood for pixel p is defined as $N = \{p_0, \dots, p_{P-1}\}$ where all pixels p_i are R distant from p . The code for pixel p is defined as a binary number where the i th digit is one if $p_i \geq p$ and zero otherwise. More formally,

$$\text{LBP}_{P,R} = \sum_{i=0}^{P-1} s(p_i - p) \cdot 2^i, \quad s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & \text{otherwise} \end{cases}. \quad (8)$$

The final feature vector is the normalized histogram of the LBP codes for a given window. The number of features in the descriptor is given by 2^P . Like GLCM, the distance between two seismic sections is computed using the Euclidean distance.

SSIM

SSIM is an index originally proposed to measure the quality of digital images and videos (Wang et al., 2004). It measures the similarity between two images using luminance, contrast, and structure information. Whereas luminance information is based on mean values, the other two are based on simple texture measures such as variances and covariances. It is usually computed over windows $N \times N$ where the measure between two windows x and y is given by

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}. \quad (9)$$

To normalize our results and represent SSIM as a distance, we use the equation $\text{DSSIM}(x, y) = (1 - \text{SSIM}(x, y))/2$.

MSE

MSE is a classical error measure that computes the average of the squares of the errors. As opposed to the texture measures presented previously, which take into consideration the relationships between neighboring pixels, this measure considers each pixel separately. It gives a direct measure of the dissimilarity between the pixels of two images A and B :

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (A_i - B_i)^2. \quad (10)$$

Clustering algorithms

In this section, we present the clustering algorithms considered in this work. Both algorithms can be executed over a distance/similarity matrix representation

of a graph and accept the number of desired clusters as a parameter.

Agglomerative clustering

This is a hierarchical clustering method that seeks to build a hierarchy of clusters (Mullner, 2011). As opposed to the divisive type, which has a top-down approach, in the agglomerative type, each observation starts in its own cluster and pairs of clusters are merged as we move up in the hierarchy (bottom-up). To decide which clusters should be merged, agglomerative clustering defines a dissimilarity measure and a linkage criterion.

The dissimilarity measure evaluates the distance between two data points. A common metric is the Euclidean distance, but any distance measure could be used. The linkage criterion determines the distance between two sets of observations as a function of the pair-wise distance between the observations in the two sets. Some common criteria are the maximum, minimum, or average pair-wise distances.

Spectral clustering

This method clusters data points based on the adjacency, degree, and Laplacian matrices of a graph representation of the observations (Von Luxburg, 2007). The adjacency matrix is a matrix with ones and zeros that tells whether two vertices v_i and v_j are connected, whereas the degree matrix is a diagonal matrix that contains the number of edges attached to each vertex. There are many definitions for the Laplacian matrix, but in this work we use the normalized Laplacian, which is defined as

$$L_{i,j} = \begin{cases} 1, & \text{if } i = j \text{ and } \deg(v_i) \neq 0 \\ -\frac{1}{\sqrt{\deg(v_i)\deg(v_j)}}, & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j, \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

where $\deg(v)$ is the degree of vertex v . We use a similarity matrix instead of the adjacency matrix. It resembles a high power of the adjacency matrix in which, instead of telling whether two vertices are connected, each element represents the similarity between two vertices. Thus, the elements in our degree matrix are given by the square root of the sum of the similarities (incoming edge weights) of the vertices. The next step of the technique is to extract from the Laplacian matrix the eigenvectors with the higher associated eigenvalues and apply a clustering algorithm, which in our case is the K-means.

Methodology

The methodology that we propose in this work is shown in Figure 1. Essentially, the workflow receives a seismic volume as input and computes the texture descriptors and pair-wise distances between seismic sections. With this information, we build a graph over which we perform a clustering to extract the most

representative sections of the data set. Each step of the workflow is explained in detail in the following sections.

Texture feature computation

In the first step, each seismic section of the input seismic volume (inlines and crosslines) is divided into tiles of size $t_s \times t_s$ (Figure 2). This procedure divides a seismic section into $n_{\text{rows}} \times n_{\text{cols}}$ tiles. For each tile, a texture descriptor is computed. The length len of the feature vector depends on the selected texture descriptor.

The final descriptor of a seismic section is a vector with $n_{\text{rows}} \times n_{\text{cols}} \times \text{len}$ features. The selection of the tile size t_s controls the scale in which the texture features will be considered, and it also impacts the length of the final vector. A smoothing step may also be performed for noisy data sets to reduce the noise impact on texture features. For MSE, the computation is done pixelwise, as discussed in the previous section. We will denote the final matrix with all features computed as $X_{n \times p}$, where p is the length of the feature vector and n is the number of seismic sections.

Seismic graph construction

In this work, we represent a seismic cube as two complete weighted undirected graphs, which we will

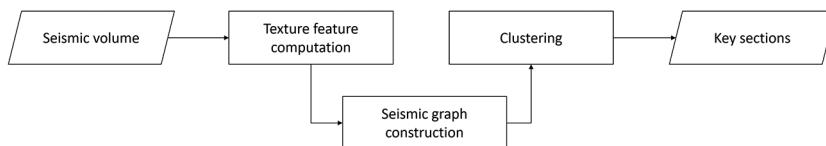


Figure 1. The proposed methodology.

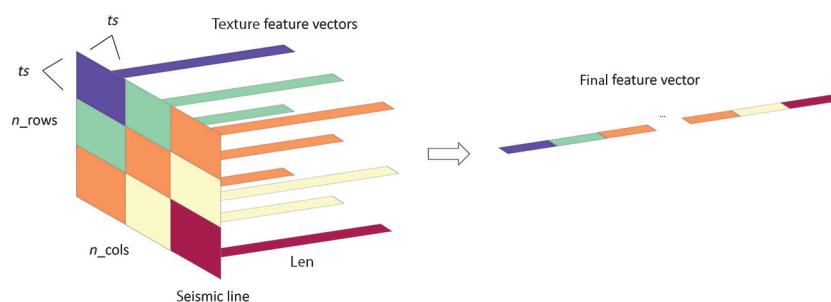


Figure 2. Texture feature computation.

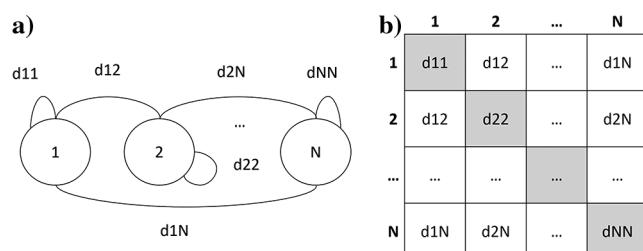


Figure 3. (a) Seismic graph and (b) its distance matrix.

refer to henceforth as inline and crossline seismic graphs (Figure 3a). The vertices of these graphs represent the seismic sections, and the weight of the edges represents the distance between the texture feature vectors of the vertices that they connect.

For the clustering algorithms considered in this work, we follow two different approaches. For the agglomerative clustering that takes a distance matrix as input, we use the graph's distance matrix D . It is a square matrix containing pair-wise distances between seismic sections i and j (Figure 3b). Spectral clustering, on the other hand, expects a nonnegative similarity score. To find a suitable similarity matrix S based on our distance matrix D , we use a Gaussian kernel radial basis function (RBF) where σ represents the width of the kernel:

$$S = e^{\left(\frac{-D^2}{2\sigma^2}\right)}. \quad (12)$$

There are three clear benefits of the representation we propose in this work. First, the representation of a large texture feature matrix $X_{n \times p}$, where p is the length of the feature vector and n is the number of seismic sections, may be much more efficiently represented by an $n \times n$ distance matrix depending on the value of n and p . Second, for attributes such as SSIM and MSE that represent dissimilarities and are defined for a pair of seismic sections, a workflow based on distance matrices seems an interesting fit. Finally, the distance matrix is the only structure representing the seismic data set that needs to be stored to extract key sections.

Clustering

In the last step of our methodology, we use the distance or the computed similarity matrix as input for a clustering algorithm. Any clustering algorithm that accepts precomputed distance/similarity matrices could be used in this step. If the geoscientist wants to control the number of sections that he will use in his interpretation, a clustering algorithm that allows the definition of the number of output clusters, such as the ones we consider in this work, is the best option.

On the other hand, if the interpreter just wants a first impression of a new data set, clustering algorithms that automatically find a suitable number of clusters (e.g., density-based spatial clustering of applications with noise (DBSCAN) [Ester et al., 1996], affinity propagation [Frey and Dueck, 2007]) may be a good alternative.

Regardless of the chosen approach, the output of this step is a grouping of seismic sections in a seismic volume according to their texture characteristics. Because we are not interested in groups of seismic sections,

after the clustering we must select the most representative section of each group. For clustering algorithms such as K-means, which expose the cluster centers, the most representative section can be computed by taking the closest seismic section to the cluster center.

For other algorithms, a reliable representative can be obtained by taking a similar approach, but in this case the cluster center must be computed afterward, based on the texture descriptors of the seismic sections in a cluster. Other centrality measures such as medoid could also be used. For attributes that are not defined for a single section such as SSIM and MSE — which are computed for a pair of sections — the mean or median of the section indexes assigned to a cluster could be used. In this case, however, the median is preferred to account for disjoint clusters for which the mean could return sections not included in the clusters.

Experiments

The data set analyzed in this work — Netherlands Offshore F3 Block Complete — consists of a 3D seismic survey carried out in the North Sea comprising an area of 24×16 km. The range of inlines goes from 100 to 750, and the range of crosslines goes from 300 to 1250. The bin size is 25 m in both directions with a Z range of 0–1848 and with a sample interval of 4 ms. We have assessed our methodology in other data sets (public and confidential), and the results were equally satisfying, but to be able to present a thorough qualitative analysis in this paper, we present only the results for the Netherlands F3 data set.

Texture feature computation

For our experiments, we considered the texture descriptors GLCM and LBP. We also compared these results with SSIM and MSE as baselines. For the texture descriptors, we evaluated several tile sizes: 25, 50, 100, and 150 pixels. To save space for our qualitative analysis, we will report only the results with a tile size of 150 pixels, which produced the best results in our experiments.

For GLCM, we used four angles (0° , 45° , 90° , and 135°), distance 1 and the features contrast, energy, entropy, homogeneity, correlation, dissimilarity, and inverse difference, whereas for LBP, we used four neighbors and distance 1. The seismic sections were rescaled, clipping the original intensities in 1% in both sides of the histogram. They were also smoothed using a Gaussian window with size 2 and rescaled to 64 gray levels.

Seismic graph construction

In this step, we computed the distances between the seismic sections for each texture attribute and built a complete undirected graph for the inlines (651 vertices) and the crosslines (951 vertices). The corresponding distance matrices are shown in Figure 4. For the texture descriptors, they represent the Euclidean distance between feature vectors of pairs of seismic sections. For SSIM and MSE, they represent the dissimilarity computed by those measures. The histogram of the images was equalized to improve the visualization. In Figure 4, we can see that LBP and GLCM distance matrices are fairly similar and seem to better capture the

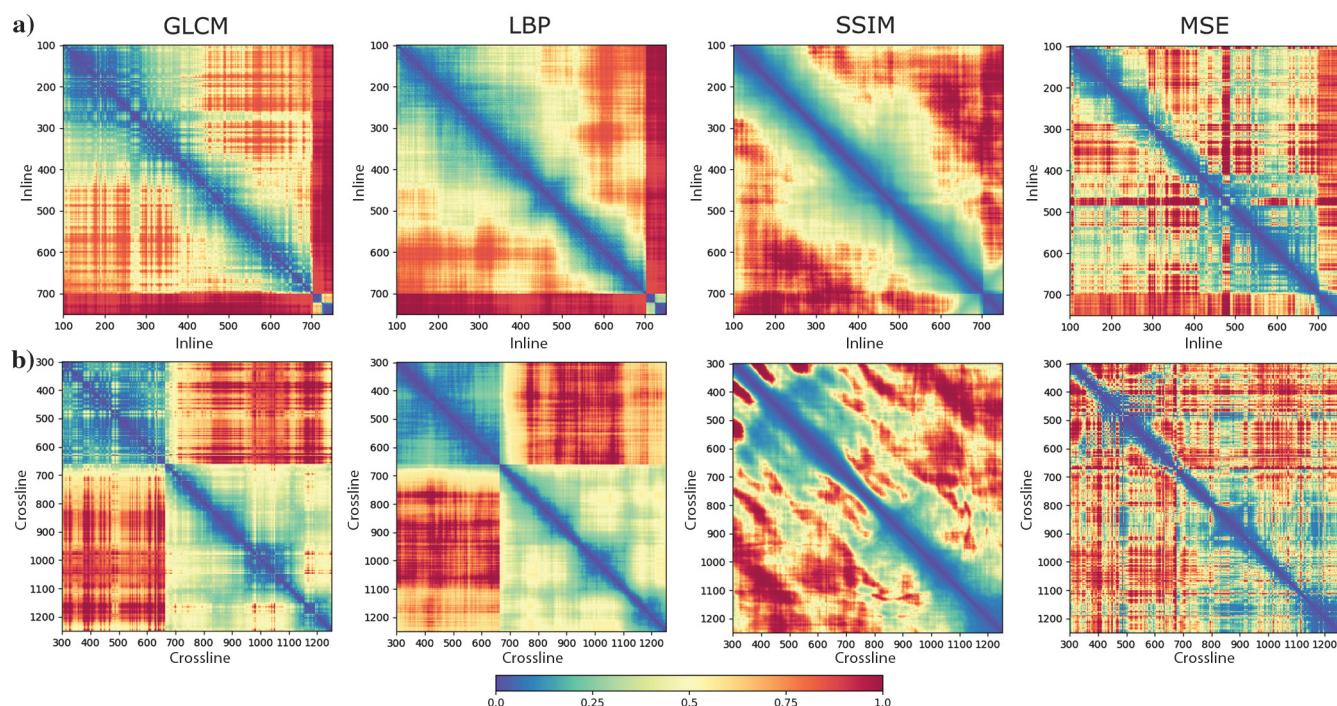


Figure 4. (a) Inline and (b) crossline distance matrices for GLCM, LBP, SSIM, and MSE. The images were equalized to improve the visualization.

differences in the data set, whereas MSE and SSIM present less structured matrices with more noise.

Agglomerative clustering

First, we investigated the agglomerative clustering. Figure 5 shows the clustering results and representative sections (in the dashed black lines) for the inlines and crosslines using all attributes. The number of clusters was set to five. For the selection of representatives for GLCM and LBP, we computed the cluster centers and selected the closest seismic sections, whereas for SSIM and MSE, we took the median of the section indexes in each cluster.

Overall, the clusters are compact, showing that there is a continuity in the geologic variations present in the

cube. However, depending on the texture attribute, some disjoint clusters may appear as is the case with the inlines for GLCM (Figure 5), in which some sections between the green and the purple clusters are mixed, and with the crosslines for MSE in which the green cluster has two disjoint parts (Figure 5). Another aspect is that the clusters are not equally spread throughout the cube, which indicates that the different attributes captured regions with more variations than others.

Figures 6 and 7 show the selected inlines and crosslines for LBP. We analyzed the results, and, due to space limitations, we will present only the representative seismic sections obtained with LBP because they produced the best results according to our analysis. Nevertheless, we compare the results with the other attributes whenever

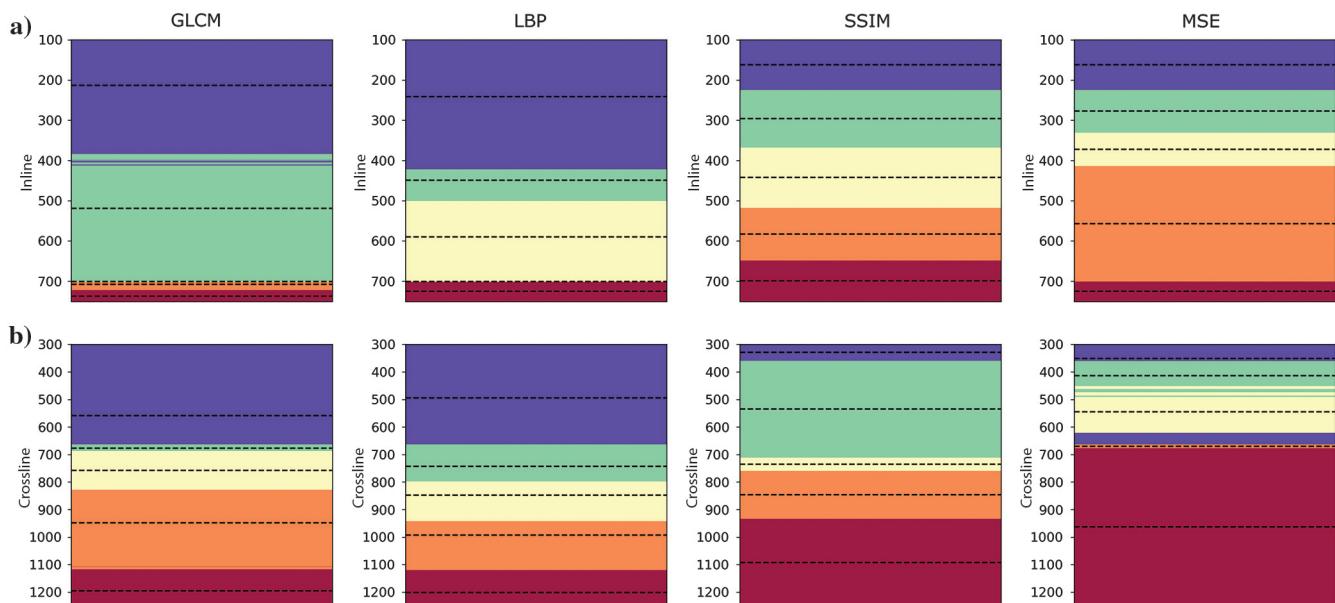


Figure 5. (a) Inline and (b) crossline agglomerative clusterings for GLCM, LBP, SSIM, and MSE. The number of clusters was set to five, and the representative sections are shown in the dashed black lines.

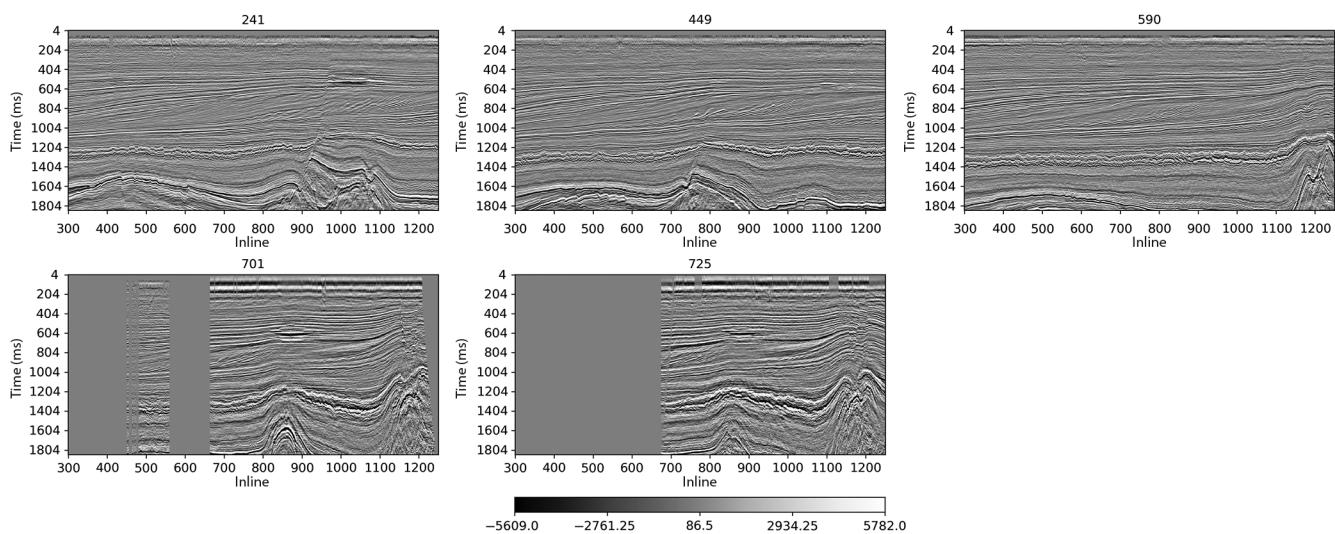


Figure 6. Selected key inlines using agglomerative clustering.

possible. For the inlines (Figure 6), one may notice the significant variations present in the inline direction, represented by the five seismic sections. In the first inline 241, there are two main geologic features to remark: the more bedded salt layer and another portion with a salt diapir deforming and faulting the upper layers. The inline 449 presents an evolution of the previous scenario, with a smaller presence of the bedded salt layer and the displacement of the salt diapir to the middle of the image along with the deformation of the upper layers, but now in the middle of the section. In the next inline 590, the presence of the bedded salt layer is almost unnoticed, with the uprise of a significant salt dome on the right. The last two inlines selected by the algorithm, 701 and 725, respectively, present a different scenario: The salt dome on the right, first seen in the inline 590, is accompanied by another one, in a smaller scale, more to the left, present in the last two sections. However, the comparison of the geology of the last two inlines is hampered by the presence of large missing data strips. This feature was probably the main reason why the clustering algorithm selected these inlines as key sections. Figure 8a shows a z -slice at 1600 ms with the key inlines highlighted.

In our view, LBP was able to satisfactorily capture the main changes in the inline direction. It enabled the hierarchical clustering to extract the seismic sections with data problems while also telling the spatial story of the data set for the inlines. GLCM also produced good results but gave too much attention to the sections with missing data, overlooking important regions in the data set like the one around section 600 with a single salt dome on the right (Figure 8a). SSIM and MSE produced

more equalized clusters, extracting more sections in a region with few variations (100–450) and missing important changes (Figure 8a). For instance, SSIM completely ignored the sections with missing data. It is noteworthy that sections with data problems could be removed beforehand if they are not relevant for the analysis.

For the crosslines, shown in Figure 7, one may also notice clear variations. In the first crossline 494, we observe a well-behaved scenario with the main stratigraphic portions of the Central North Sea Basin, the presence of the bedded layer of salt in the bottom of the seismic section, and a considerable strip of missing data on the right. In crossline 743, one can notice the early uprise of a salt dome layer in the middle of the section and a small deformation of the closest upper layers. In the next crossline 848, there is a displacement and enlargement of the previous salt dome. Besides, another salt dome rises, more on the right, also causing deformation, but only on the closest layers. In the next key line, crossline 993, the central salt dome from the previous crossline has moved to the left. In the last image, crossline 1201, we observe the presence of another salt diapir, on the right side of the seismic section. Both domes cause intense deformation on the upper layers and possibly faulting. The grid with the selected crosslines is presented in Figure 8a.

When we compare the selected crosslines for all attributes, LBP was the only one to capture all of the significant changes in the crossline direction. As with the inlines, LBP presented a good balance in extracting sections that presented data problems and sections with important geologic structures. The clustering with

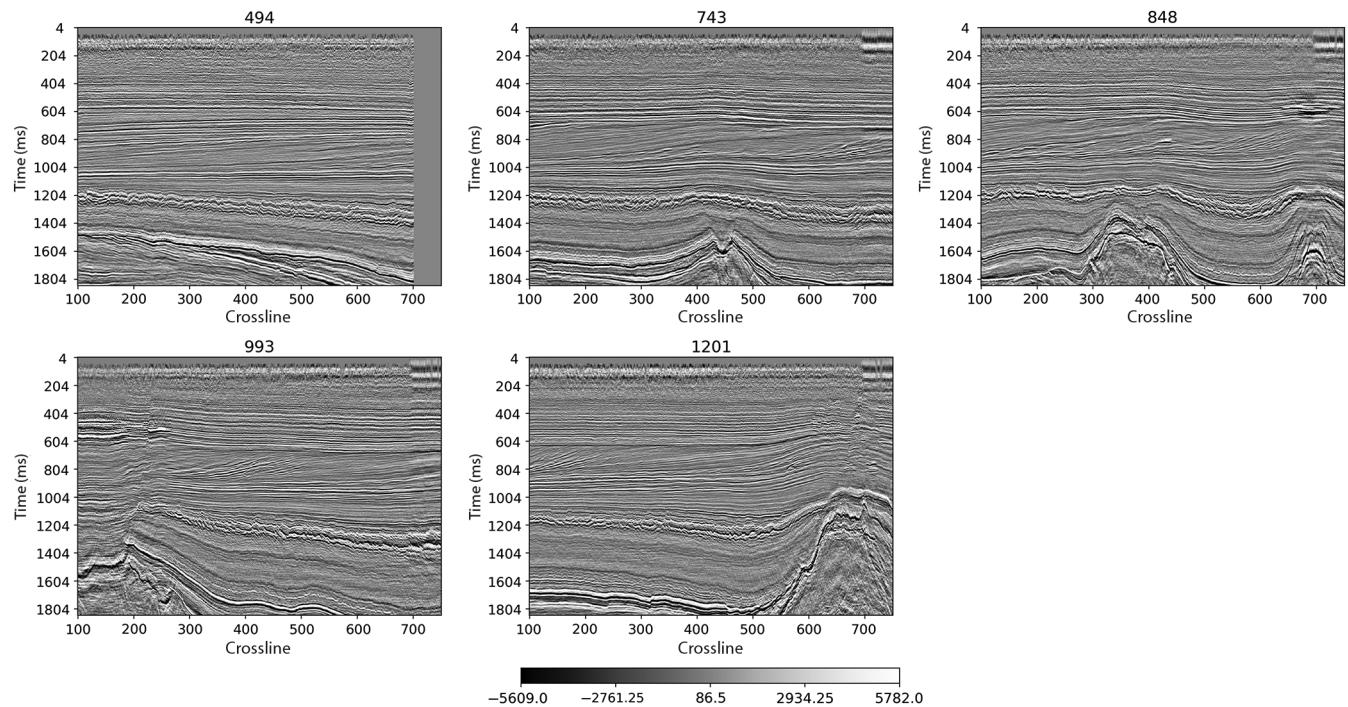


Figure 7. Selected key crosslines using agglomerative clustering.

the GLCM attribute missed the region around section 850, where two salt domes can be seen, and extracted two crosslines in the range of 300–450 where there is little geologic variation. Probably the reason for the selection of these crosslines was the change in the width of the missing data strip on the right (Figure 8a). Similarly, SSIM focused on the missing data in the first sections and missed the region (around section 1200) with the salt dome on the right. Finally, MSE mostly extracted sections with data problems (300–670) missing almost all of the important structures in the data set. Our best guess is that due to its pixel-based approach,

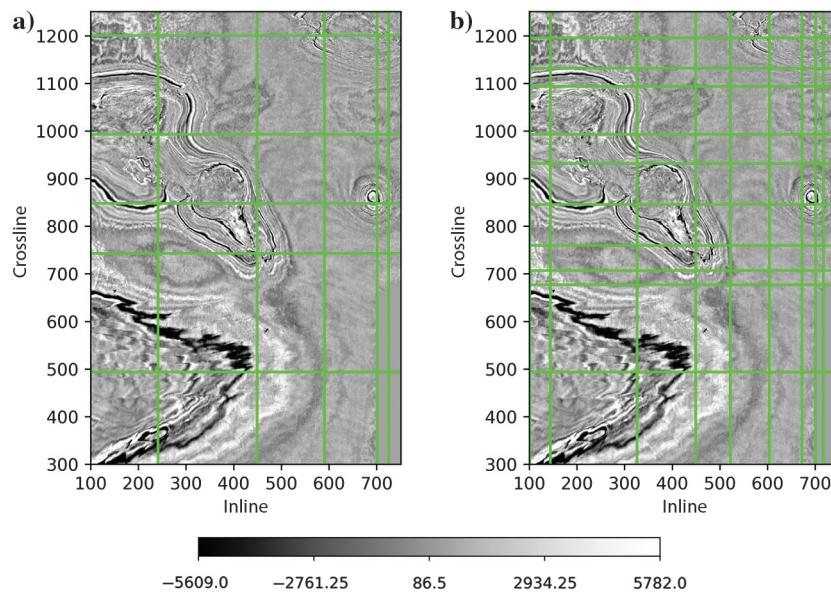


Figure 8. Grid of (a) five and (b) 10 key sections over time slice 1600 ms obtained with agglomerative clustering.

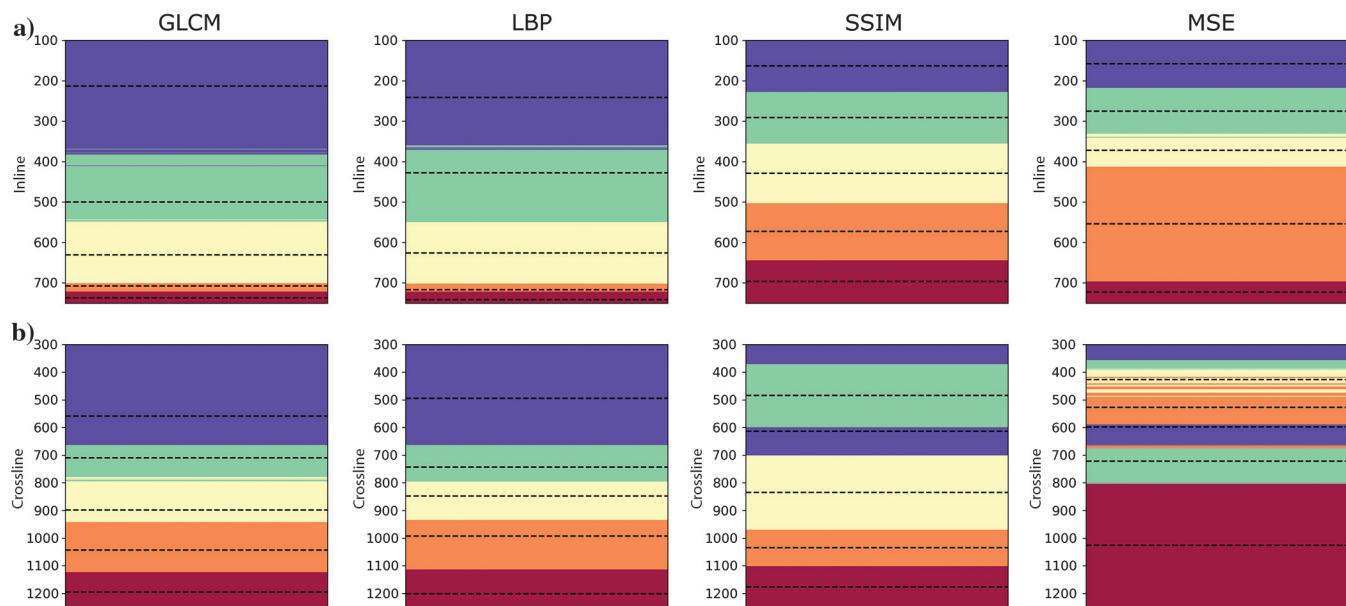


Figure 9. (a) Inline and (b) crossline spectral clusterings for GLCM, LBP, SSIM, and MSE. The number of clusters was set to five, and the representative sections are shown in the dashed black lines.

MSE was heavily influenced by noisy pixels present in this part of the data set (Figure 8a).

Although the last discussions focused on the results with five clusters, we also show in Figure 8b the grid obtained with 10 clusters for LBP. We observe a concentration of key sections around inline 700 and crossline 700 where there are transitions between full-record seismic sections and sections with missing data problems. Conversely, key crosslines show a low concentration before section 700, indicating that the methodology found little variation in the bedded salt layer shown on the lower left corner of the z -slice (Figure 8b). Most

of the extracted crosslines are after section 700 where there is a set of salt domes on the upper left (700–1150) and two salt domes on the upper right (850 and 1200). The two grids (Figure 8) illustrate how the methodology is able to extract key sections taking into consideration the changes in the geology of the subsurface.

Spectral clustering

In the second set of experiments, we investigated the spectral clustering algorithm. The results and representative sections (in the dashed black lines) for the inlines and crosslines using all attributes are shown in Figure 9. Again, the number of clusters was set to five. For GLCM and LBP, we computed the cluster centers and selected the closest seismic sections as representatives, whereas for SSIM and MSE, we took the median of the section indexes.

The clusters are mostly compact, indicating that the geologic variations throughout the cube are smooth. However, some disjoint clusters are present in the cross-lines for SSIM (Figure 9), in which the purple cluster has two disjoint parts, and for MSE in which all except the purple cluster have disjoint parts. As with agglomerative clustering, the clusters are not equally spread, which indicates that the different attributes captured regions with more variations than others. Nevertheless, this characteristic is less evident with spectral clustering.

The selected inlines and crosslines for LBP are shown in Figures 10 and 11. Again, we will present only

the representative seismic sections obtained with LBP because they produced the best results according to our analysis. However, we compare the results with the other attributes whenever possible. We will not repeat the thorough analysis that we performed for agglomerative clustering, comparing the structures and geologic variations of the data set with the ones captured by each section because the analysis would be too similar. Here, we will focus on the differences between the results obtained with spectral clustering with respect to agglomerative clustering. For the inlines (Figure 10), the results with LBP, SSIM, and MSE are quite similar.

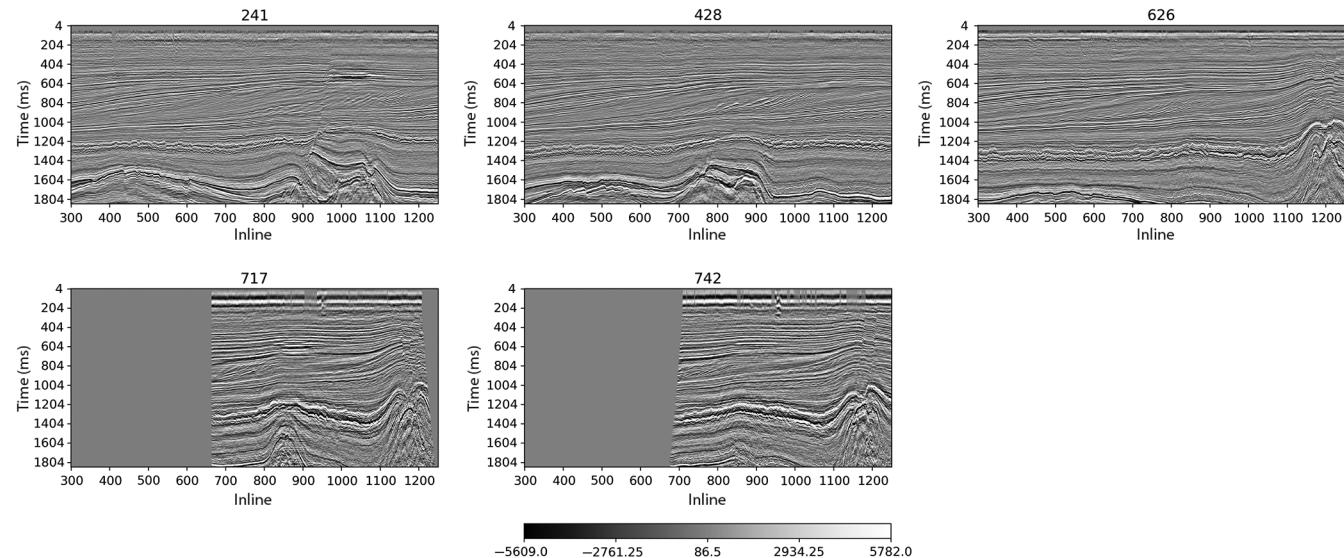


Figure 10. Selected key inlines using spectral clustering.

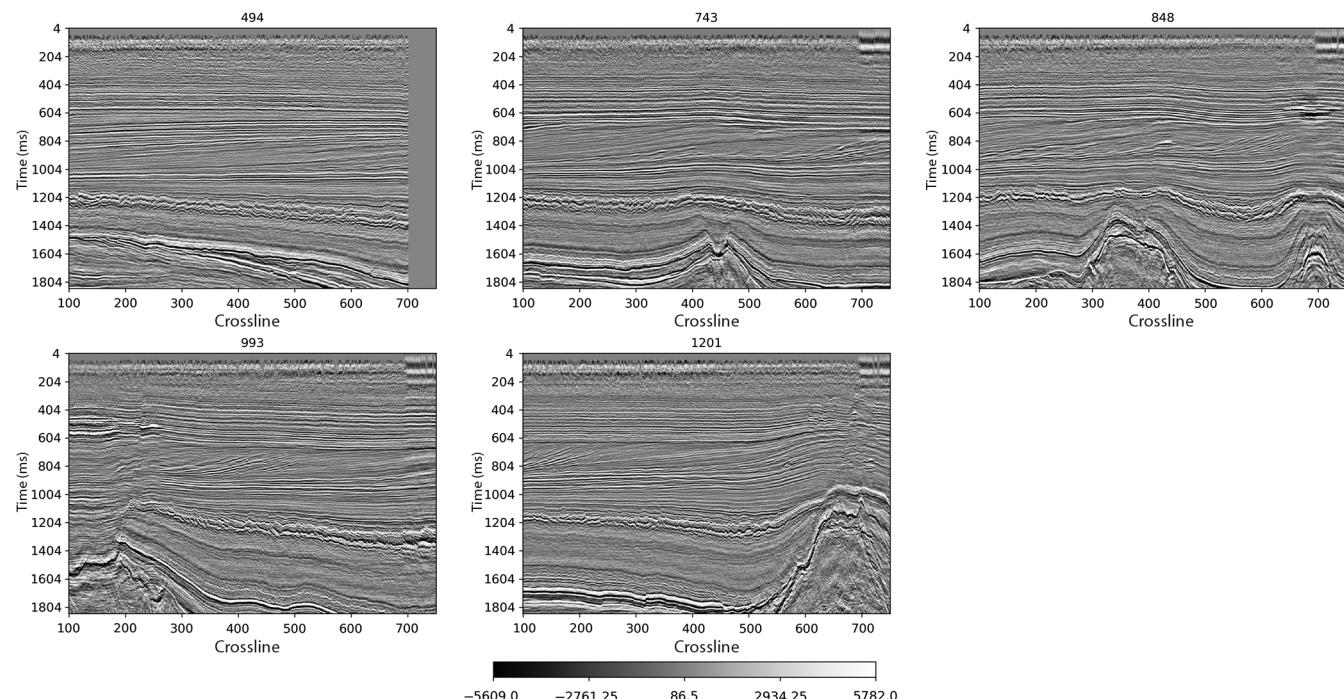


Figure 11. Selected key crosslines using spectral clustering.

The selected seismic sections are very close to the ones selected with the previous clustering algorithm. One interesting difference is that for spectral clustering, the results with GLCM were as good as the ones with LBP. Spectral clustering was able to produce more robust results with that texture attribute.

For the crosslines (Figure 11), the results with LBP and MSE were very similar to the ones obtained with the previous clustering algorithm. For GLCM, however, the results were improved. With spectral clustering, the selected seismic sections for that texture attribute were able to capture the main variations in the data set, producing results as good as the ones obtained with LBP. For SSIM, the result did not improve, but instead of missing the region with the salt dome on the right (around section 1200), it missed the single salt dome around section 750.

In Figure 12, we show the grid obtained with five (Figure 12a) and 10 (Figure 12b) clusters. For five clusters, the grid is very similar to the one obtained with agglomerative clustering. For 10 clusters, however, the grid is more regular, indicating a tendency of spectral clustering to select seismic sections equally distant when more clusters are considered.

As mentioned earlier, it is possible to control the scale at which the texture representations are captured by changing the tile size ts . The higher the tile size, the larger are the structures that will be taken into consideration in the computation of (dis)similarities. If smaller tile sizes are used, probably they will not be able to capture whole structures, but their constituent parts or patterns. Nevertheless, too-small tile sizes will potentially capture undesirable minor changes or noises with little meaning to the interpreter. Moreover, in the provided examples, we looked for five and 10 sections in each direction, being necessary, at the discretion of the geoscientist, to increase or not the grid density according to the task at hand.

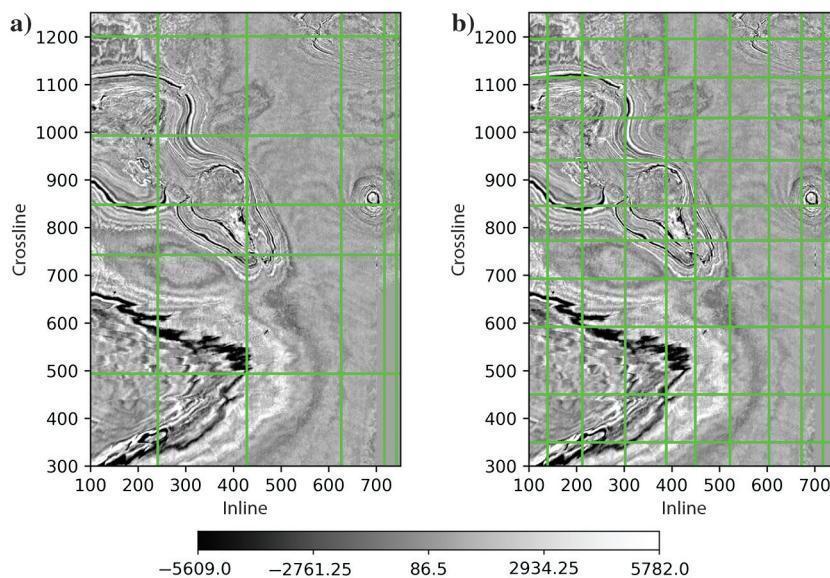


Figure 12. Grid of (a) five and (b) 10 key sections over time slice 1600 ms obtained with spectral clustering.

A possible workflow for the application of the proposed methodology is the following: The interpreter starts by selecting a small number of key inlines and crosslines to work with. These few key sections give him a first impression about the area under study and may be used to start the interpretation process. Once he is satisfied with the interpretation at that scale, he can ask for a denser grid (more key sections) and continue his analysis at the new scale. This refinement process can go on until the end of the interpretation process.

Conclusion

In this work, we proposed a graph representation of a seismic data set based on texture attributes to aid the seismic interpretation process. The seismic graph presents a powerful and concise representation of a 3D seismic survey, and its analysis may bring interesting insights about the data set and its main characteristics. We performed a graph analysis, evaluating different distance measures and clustering algorithms, and we showed that the methodology was able to capture the underlying geologic variations and data issues for a public data set. The final adaptive grid can be used by seismic interpreters to guide their interpretation process at different scales.

In our experiments, the LBP texture attribute, together with the agglomerative clustering algorithm, produced the best results, being able to satisfactorily describe the data set in the inline and crossline directions. The other attributes either missed important parts of the data set or focused too much on regions with little variation. Regarding the clustering algorithms, agglomerative clustering yielded the best results, producing grids that respected the characteristics of the data set. Although spectral clustering performed similarly well for five clusters, it tended to produce equally distributed clusters for 10 clusters.

In future work, we intend to investigate how the proposed methodology performs on time slices and which information it can bring to the interpretation process. In addition, the key sections obtained with the graph representation can be used to compare different seismic data sets in the search for analogs.

Data and materials availability

Data associated with this research are available and can be accessed via the following URL: <https://terranubis.com/datainfo/Netherlands-Offshore-F3-Block-Complete>.

References

- Alfarraj, M., Y. Alaudah, and G. AlRegib, 2016, Content-adaptive non-parametric texture similarity measure: 18th IEEE International Workshop on Multimedia Signal Processing, 1–6.

- AlRegib, G., M. Deriche, Z. Long, H. Di, Z. Wang, Y. Alaudah, M. Shafiq, and M. Alfarraj, 2018, Subsurface structure analysis using computational interpretation and learning: A visual signal processing perspective: IEEE Signal Processing Magazine, **35**, 82–98, doi: [10.1109/MSP.2017.2785979](https://doi.org/10.1109/MSP.2017.2785979).
- Amin, A., and M. Deriche, 2016, Salt-dome detection using a codebook-based learning model: IEEE Geoscience and Remote Sensing Letters, **13**, 1636–1640, doi: [10.1109/LGRS.2016.2599435](https://doi.org/10.1109/LGRS.2016.2599435).
- de Matos, M. C., P. L. M. Osório, and P. R. S. Johann, 2003, Unsupervised seismic reservoir characterization using wavelet transform and self organizing maps of a deep-water field, Campos Basin, Offshore Brazil: 73rd Annual International Meeting, SEG, Expanded Abstracts, 1458–1461, doi: [10.1190/1.1817566](https://doi.org/10.1190/1.1817566).
- Ester, M., H. P. Kriegel, J. Sander, and X. Xu, 1996, A density-based algorithm for discovering clusters in large spatial databases with noise: KDD, **96**, 226–231.
- Ferreira, R., E. V. Brazil, R. Silva, and R. Cerqueira, 2018, Texture-based similarity graph to aid seismic interpretation: Annual Convention and Exhibition.
- Ferreira, R. D. S., A. B. Mattos, E. V. Brazil, R. Cerqueira, M. Ferraz, and S. Cersosimo, 2016, Multi-scale evaluation of texture features for salt dome detection: IEEE International Symposium on Multimedia, 632–635.
- Frey, B. J., and D. Dueck, 2007, Clustering by passing messages between data points: Science, **315**, 972–976, doi: [10.1126/science.1136800](https://doi.org/10.1126/science.1136800).
- Haralick, R. M., and K. Shanmugam, 1973, Textural features for image classification: IEEE Transactions on Systems, Man, and Cybernetics, **SMC-3**, 610–621, doi: [10.1109/TSMC.1973.4309314](https://doi.org/10.1109/TSMC.1973.4309314).
- Long, Z., Y. Alaudah, M. Ali Qureshi, Y. Hu, Z. Wang, M. Alfarraj, and H. Di, 2018, A comparative study of texture attributes for characterizing subsurface structures in seismic volumes: Interpretation, **6**, no. 4, T1055–T1066, doi: [10.1190/INT-2017-0181.1](https://doi.org/10.1190/INT-2017-0181.1).
- Long, Z., Y. Alaudah, M. A. Qureshi, M. A. Farraj, Z. Wang, A. Amin, M. Deriche, and G. AlRegib, 2015, Characterization of migrated seismic volumes using texture attributes: A comparative study: 85th Annual International Meeting, SEG, Expanded Abstracts, 1744–1748, doi: [10.1190/segam2015-5934664.1](https://doi.org/10.1190/segam2015-5934664.1).
- Mattos, A. B., R. S. Ferreira, R. M. D. G. Silva, M. Riva, and E. V. Brazil, 2017, Assessing texture descriptors for seismic image retrieval: 30th SIBGRAPI Conference on Graphics, Patterns and Images, 292–299.
- Mullner, D., 2011, Modern hierarchical, agglomerative clustering algorithms: ArXiv preprint arXiv, 1109–2378.
- Ojala, T., M. Pietikäinen, and T. Mäenpää, 2002, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns: IEEE Transactions on Pattern Analysis and Machine Intelligence, **24**, 971–987, doi: [10.1109/TPAMI.2002.1017623](https://doi.org/10.1109/TPAMI.2002.1017623).
- Roden, R., T. Smith, and D. Sacrey, 2015, Geologic pattern recognition from seismic attributes: Principal component analysis and self-organizing maps: Interpretation, **3**, no. 4, SAE59–SAE83, doi: [10.1190/INT-2015-0037.1](https://doi.org/10.1190/INT-2015-0037.1).
- Saraswat, P., and M. K. Sen, 2012, Artificial immune-based self-organizing maps for seismic-facies analysis: Geophysics, **77**, no. 4, O45–O53, doi: [10.1190/geo2011-0203.1](https://doi.org/10.1190/geo2011-0203.1).
- Shafiq, M. A., T. Alshawi, Z. Long, and G. AlRegib, 2018, The role of visual saliency in the automation of seismic interpretation: Geophysical Prospecting, **66**, 132–143, doi: [10.1111/1365-2478.12570](https://doi.org/10.1111/1365-2478.12570).
- Taner, M. T., J. D. Walls, M. Smith, G. Taylor, M. B. Carr, and D. Dumas, 2001, Reservoir characterization by calibration of self-organized map clusters: 71st Annual International Meeting, SEG, Expanded Abstracts, 1552–1555, doi: [10.1190/1.1816406](https://doi.org/10.1190/1.1816406).
- Von Luxburg, U., 2007, A tutorial on spectral clustering: Statistics and Computing, **17**, 395–416, doi: [10.1007/s11222-007-9033-z](https://doi.org/10.1007/s11222-007-9033-z).
- Wang, Z., A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, 2004, Image quality assessment: From error visibility to structural similarity: IEEE Transactions on Image Processing, **13**, 600–612, doi: [10.1109/TIP.2003.819861](https://doi.org/10.1109/TIP.2003.819861).
- Wang, Z., H. Di, M. A. Shafiq, Y. Alaudah, and G. AlRegib, 2018, Successful leveraging of image processing and machine learning in seismic structural interpretation: A review: The Leading Edge, **37**, 451–461, doi: [10.1190/tle37060451.1](https://doi.org/10.1190/tle37060451.1).
- Zujovic, J., T. N. Pappas, and D. L. Neuhoff, 2009, Structural similarity metrics for texture analysis and retrieval: 16th IEEE International Conference on Images Processing, 2225–2228.



Rodrigo S. Ferreira received a B.S. (2009) in electrical engineering from the Rio de Janeiro State University, Brazil, and an M.S. (2011) and a Ph.D. (2015) in electrical engineering from the Pontifical Catholic University of Rio de Janeiro. During this time, he worked on research projects related to knowledge-based image analysis for remote sensing applications in the Computer Vision Lab. In 2013, he was a visiting Ph.D. student at the University of Pavia, Italy. Since 2015, he has been working as a research staff member at IBM Research Brazil. His research interests include computer vision and machine learning applied to natural resources.



Emilio Vital Brazil received an M.S. and a Ph.D. in mathematics with specialization in computer graphics from the Instituto Nacional de Matemática Pura e Aplicada (IMPA). He is a scientist at IBM Research Labs in the Natural Resources Group, focusing his research on machine AI applied to the oil and gas industry. After the Ph.D., he worked with graph theory and computer graphics at

COPPE/UFRJ as a postdoc. He also spent four years as a postdoctoral fellow at the interactive reservoir modeling, visualization, and analytics research group at the University of Calgary. During this time, he worked in collaboration with practitioners in the oil and gas industry on the research and development of interactive visual computing software technologies and their application to a variety of problems in geoscience and reservoir engineering.



Reinaldo Mozart Silva received a B.S. (2017) in geophysics from the Federal Fluminense University, Brazil. Currently, he is an M.S. candidate in informatics at the Pontifical Catholic University of Rio de Janeiro. In 2016, he joined IBM Research Brazil as a research intern, working with projects related to deep learning applied to seismic data analysis and knowledge engineering applied

to the mining industry. Since 2018, he has been working as a geophysicist researcher at Tecgraf (PUC-Rio), working with deep learning applied to seismic data interpretation. His research interests include seismic data processing, deep neural networks, and seismic data analysis.



Renato Cerqueira is a senior research manager and senior technical staff member at IBM Research Brazil, where he leads the natural resources solutions group, aimed at investigating and developing new technologies to advance the ongoing digitalization of the oil and gas and mining industries. In recent years, he and his research group have been working together with different industry partners to create AI-based technologies for data and knowledge intensive processes, mainly in subsurface exploration and oil production.