Texture-Based Similarity Graph to Aid Seismic Interpretation*

Rodrigo S. Ferreira[1], Emilio Vital Brazil[1], Reinaldo Silva[1], and Renato Cerqueira[1]

Search and Discovery Article #70365 (2018)**
Posted October 8, 2018

*Adapted from extended abstract based on oral presentation given at AAPG 2018 Annual Convention & Exhibition, Salt Lake City, Utah, United States, May 20-23, 2018
**Datapages © 2018. Serial rights given by author. For all other rights contact author directly. DOI:10.1306/70365Ferreira2018

[1]IBM Research, Rio de Janeiro, Brazil (rosife@br.ibm.com)

Abstract

Seismic interpreters use their trained eyes to assess the similarity between seismic datasets. However, to find and evaluate all relevant parts of a seismic cube can be a time-consuming task. We have developed a method based on texture analysis and graph theory that can automatically compare seismic lines. Such a method has the potential to help experts in several tasks and, to the best of our knowledge, it is the first one to tackle seismic image similarity combining texture descriptors and graphs.

This work proposes a texture-based dissimilarity graph that represents the seismic survey as a graph whose vertices represent seismic lines and whose edges represent the cosine distance between texture descriptors computed for each line. By using this representation, it is possible to calculate the similarity between any two seismic lines. To demonstrate the capabilities of our technique, we apply it on a real-world dataset. Using our method, we suggest which seismic lines (*key lines*) should be considered in the interpretation process taking into consideration their distance to neighboring lines. This distance, ultimately, captures the visual dissimilarity between two lines.

Key lines computed with this method could be used to build a non-regular grid, which is more likely to capture the underlying structures present in a survey allowing for a faster and more precise interpretation. Another possible application would be to use dissimilarity information to highlight different areas within a seismic survey. Experiments conducted on the Netherlands F3 public dataset (OSR, 2018) indicate that the methodology has a great potential to aid in the early steps of the seismic interpretation process.

Introduction

Seismic reflection survey is the most used geophysical method in the O&G industry. In addition to providing high-quality structural images, the dense sampling of the seismic data can sometimes make it possible to map reservoir quality and the distribution of oil and gas in it (Bacon et al., 2007). For the interpretation of such data, geoscientists use their knowledge background to annotate parts of big datasets, segmenting critical slices that present important geometric changes. In other words, they create regular or non-regular grids, aiming to describe the structural context, envelope, continuity, and coherence of key subsurface strata, in a process called seismic facies analysis (Mitchum, 1977). However, finding and evaluating all relevant parts of a seismic cube can be a time-consuming task, especially with the ever-growing datasets.

In this sense, several methods have been proposed to assist geoscientists and optimize the interpretation process. West et al. (2002) combine image textural analysis with a Probabilistic Neural Network (PNN) classification to quantitatively map seismic facies in three-dimensional data. Furthermore, the authors extend their methodology to the interpretation of AVO attributes volumes, such as intercept and gradient.

Another example of a technique that assists the interpretation process is proposed in the work of Roden et al. (2015) which uses Principal Component Analysis (PCA) to understand which seismic attributes, or a combination of these, have interpretative significance. Besides that, the authors apply a Self-Organizing Maps (SOM) algorithm, a form of unsupervised neural networks, to reveal geological clusters and patterns in the attributes, possibly indicating sweet spots, fault trends and more.

More focused on salt body detections, the work of Guillen et al. (2015) presents a machine-learning algorithm, more precisely the Extremely Random Trees Ensemble, to train and automatically identify and classify salt regions. For validation purposes, the authors applied the proposed methodology in a complex synthetic seismic dataset that represents deep-water areas in the Gulf of Mexico.

The work of Mattos et al. (2017) presents an evaluation of the accuracy of textures descriptors such as Gabor Filters, GLCM, and LBP for seismic image retrieval. Moreover, for validation purposes, the authors used interpreted real seismic datasets to compare with the texture-based retrieval results, considering different parameters and similarity metrics.

Although there are several works that investigate the potential of machine learning in seismic interpretation and the role that texture descriptors may play in characterizing the subsurface, this is the first work, to the best of our knowledge, that approaches seismic image similarity combining texture descriptors and graphs. The adjacency matrix analysis performed in this work follows the works of Foote (1999), Foote and Coper (2001) and Dubnov and Apel (2004), which used self-similarity matrix for audio and music analysis.

Therefore, this work proposes a texture-based dissimilarity graph whose vertices represent seismic lines and whose edges represent the cosine distance between texture descriptors computed for each line. By using this representation of a seismic dataset, we can calculate the similarity between any two seismic lines, highlighting the significant differences present in the seismic dataset and drawing geoscientists’ attention to key regions presenting strata geometry alterations.

Proposed Method

The method we propose in this work is shown in Figure 1. Each step of the workflow is explained in detail in the following sections.

Texture Descriptor Computation

In the first step, each seismic line (inline and crossline) of a given seismic cube is divided into tiles of size **ts** (Figure 2). This procedure divides a seismic line into **n_rows x n_cols** tiles. For each tile, a texture descriptor is computed. The length **len** of the texture descriptor vector depends on the selected algorithm (GLCM (Haralick, 1979) and LBP (Ojala et al., 2002) were investigated in this work).

The final descriptor of a seismic line is a feature vector with **n_rows x n_cols x len** features. It is possible in this step to stack features from different descriptors, in which case a feature normalization procedure is required. Depending on the final length of the feature vector, a dimensionality reduction step may also be performed (ex. PCA). The selection of the parameter **ts** controls the scale in which texture features will be considered and it affects the length of the final descriptor.

Seismic Graph Construction

In this work, we represent a seismic cube as two complete weighted undirected graphs, which we will refer to henceforth as **inline/crossline seismic graphs**. The vertices of these graphs represent seismic lines and the weight of the edges represents the cosine distance between the texture descriptors of the vertices they connect. For most of the computations in this paper, we will be interested in the adjacency matrix, which in our case can be seen as a dissimilarity matrix (Figure 3).

Key Line Selection

The final step comprises a visual inspection of the adjacency matrix, which may give valuable information on the key regions of a seismic cube, highlighting their relationship with other regions. This procedure may provide a good guess on how many main clusters there are in the dataset, whose centers and boundaries could be used as initial key lines for the interpretation. Another available tool is the plots of the adjacency matrix rows. This method helps the interpreter to gain insights about the region being studied, allowing the identification of variations in the dataset that may be correlated with changes in the geology or with problems in the data (e.g. noise, missing data etc.).

Experimental Analysis

The dataset used in this work – Netherlands F3 – consists of a 3D seismic survey carried out in the North Sea comprising an area of 24×16 km^2. The range of inlines go from 100 to 750, and of crosslines go from 300 to 1250. The bin size is 25m in both directions with a Z range of 0-1848 with a sample interval of 4ms.

Texture Descriptor

For the texture descriptor computations performed in the experiments, we considered three different tile sizes: 50, 100, and 150 pixels. These tile sizes were defined after a visual inspection of some seismic lines and they represent respectively regions of 1.2km, 2.5km and 3.7km in the horizontal axis.

For our experiments, we considered the GLCM and LBP texture descriptors. The GLCM algorithm was used with the following parameters:

- features: contrast, correlation, energy, homogeneity
- angles: 0, 45, 90, 135
- distances: 1

This parametrization produces a feature vector with 16 elements. For LBP, although a classical parametrization uses 8 neighbors, we used 4 so that the feature vector has also 16 elements instead of 256:

- neighbors: 4
- distances: 1

The number of tiles for each tile size is shown in Table 1.

The seismic lines' images were rescaled, clipping the original intensities in 1% in both sides of the histogram, and were smoothed using a Gaussian window with size 2. Although the experiments were conducted with all tile sizes and texture descriptors, for brevity, the experimental analysis presented in this paper will focus on the tile size 150 for the LBP texture descriptor since this was the combination that produced the best results.

Seismic Graph Construction

The inline and crossline seismic graphs for the Netherlands F3 dataset are shown in Figure 4. They represent the cosine distance between LBP texture descriptors of pairs of seismic lines. The histogram of the images was equalized to enhance the visualization.

Key Line Selection

After the analysis of the adjacency matrices presented in Figure 4, we could delineate the main clusters that stood out for each direction. At a first glance, it is possible to see a strong similarity change at inline 700 and at crossline 661, which are possibly due to data acquisition or processing problems. The clusters are presented in Figure 5. Sometimes the clusters are not clearly defined in the adjacency matrix but a quick visual analysis may give a clue on how many main clusters there are in the dataset. This gives the interpreter a general idea of the dataset and guides the following analysis, which should help to refine the clustering.

To validate this analysis, we compared it to a description of the dataset that was carried out previously by a geoscientist. For the inlines (Figure 5 (a)), cluster 1 comprises a region with a single salt dome (lines 100-202). Cluster 2 matches a region in which the main salt dome splits into two smaller ones and one of them disappears from the seismic section (lines 203-296). The next cluster (3) includes lines 297 to 432, which corresponds to an area where the main salt dome moves to the left and starts to disappear. Cluster 4 captures the lines (433-495) in which the main salt dome finally disappears. In cluster 5, a salt dome rises on the right of the section (lines 496-573). In the next cluster (6), the salt dome grows bigger and generates intense normal faulting on the strata above (lines 574-650). Cluster 7 goes from line 651 to 700, in which a second salt dome appears. Finally, cluster 8 (lines 701-750) represents a region of the dataset in which there is a large missing data strip and the data is shifted to the left. The selected key inlines (mean line of each cluster) are shown in Figure 7. The plot of lines in the boundaries between clusters was omitted for conciseness.

For the crosslines (Figure 5 (b)), cluster 1 matches a region in the seismic cube (lines 300-661) with no significant change in the geometry of subsurface strata, except for a small strip of missing data on the right. Cluster 2 comprises the next lines, in which the missing data strip shortens and disappears and a salt dome rises in the middle of the section (lines 662-751). The next cluster (3) includes lines 752 to 942, which corresponds to the appearance of a second salt dome on the right, which disappears a few lines later. Cluster 4 matches the region in which the remaining salt dome moves to the left of the section until it disappears from the seismic section (lines 899-1110). Finally, cluster 5 represents the appearance of a salt dome on the right that rises, generating structural changes in the strata above (1111-1250). The last lines of this cluster present a missing data strip on the right. The selected key crosslines (mean line of each cluster) are shown in Figure 8.

Another visualization that may help the analysis at this point is the plot of the dissimilarity of specific lines (Figure 6). We selected the first line, the last and a seismic line in the middle of the cube for each direction but any lines that seem interesting could be used in this step. For the inlines, the selected lines present a fair agreement with the clusters visually selected previously. By analyzing the plot of the line 100, we see significant changes around lines 200, 300 and 430. These changes are consistent with the boundaries between clusters 1-2, 2-3 and 3-4. After that, the curve decreases around line 500 and show other clear changes around lines 570 and 650 that are consistent with clusters 5 and 6. The plot of line 475 corroborates this analysis and shows strong changes around lines 700 and 725. These changes match the boundaries between clusters 7 and 8, although we decided to not delineate a separate cluster for the region between slices 725 and 750, because they were already included in a cluster that clearly represented a region with data issues.

For the crosslines (Figure 6 (b)), the three selected lines show an agreement with the clusters visually selected in the previous step. All three show a significant change around inline 660, which matches the strong dissimilarity raise observed in the boundary between clusters 1 and 2. There is also a clear variation around line 1100, observed in all three curves, which matches the boundary between clusters 4 and 5. The plot of lines 300 and 775 also show a change around 920, which corresponds to the boundary between clusters 3 and 4. An interesting finding in this plot is the peaks around lines 415 and 1175, which indicate possible cluster boundaries that we did not mark in the previous analysis.

Conclusions

In this work, we have investigated the use of texture descriptors to represent seismic lines. These representations were modeled as vertices in a seismic graph whose edge weights are given by the dissimilarities between the seismic lines. A visual inspection of the adjacency matrix helped us to delineate key regions of the Netherlands F3 dataset, which were compared to a previous description of the dataset performed by a geoscientist. This comparison showed that the selected regions were highly correlated to important features of the geology of the area.

The centers and boundaries of the selected clusters for inlines and crosslines could be used as initial key lines for the seismic interpretation. This method may help the interpreter to gain insights about the region being studied, allowing the identification of variations in the dataset that may be associated with changes in the geology or with problems in the data (e.g. noise, missing data etc.).

As future works, we intend to further investigate the representation of a seismic dataset as an adjacency matrix by evaluating the insights that spectral analysis can bring to seismic interpretation. Additionally, we would like to evaluate the quality of clusters obtained from the adjacency matrix by directly applying clustering algorithms such as k-means and hierarchical clustering.

Acknowledgments

The authors thank dGB Earth Sciences for maintaining the Open Seismic Repository (OSR) and for making the dataset used in this work publicly available.

References Cited

Bacon, M., R. Simm, and T. Redshaw, 2007, 3-D seismic interpretation. Cambridge University Press.

Dubnov, S., and T. Apel, 2004, Audio Segmentation by Singular Value Clustering. In ICMC.

Foote, J., 1999, Visualizing music and audio using self-similarity: In Proceedings of the seventh ACM international conference on Multimedia (Part 1), p. 77-80.

Foote, J., and M.L. Cooper, 2001, Visualizing Musical Structure and Rhythm via Self-Similarity: In ICMC, v. 1, p. 423-430.

Guillen, P., G. Larrazabal, G. González, D. Boumber, and R. Vilalta, 2015, Supervised learning to detect salt body: In 2015 SEG Annual Meeting. Society of Exploration Geophysicists.

Haralick, R.M., 1979, Statistical and structural approaches to texture: in Proceedings of the IEEE, v. 65, p. 786804.

Mattos, A.B., R.S. Ferreira, R.M.D.G. de Silva, M. Riva, and E.V. Brazil, 2017, Assessing Texture Descriptors for Seismic Image Retrieval: In Graphics, Patterns and Images (SIBGRAPI), 2017 30th SIBGRAPI Conference, p. 292-299.

Mitchum, R.M., 1977, Glossary of terms used in seismic stratigraphy: AAPG Memoir 26, p. 205–212.

Ojala, T., M. Pietikainen, and T. Maenpaa, 2002, Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns: IEEE Trans. Pattern Analysis and Machine Intelligence, v. 24/7, p. 971-987.

OSR - Open seismic repository, https://opendtect.org/osr/index.php, accessed: 2018 March 29th.

Roden, R., T. Smith, and D. Sacrey, 2015, Geologic pattern recognition from seismic attributes: Principal component analysis and self-organizing maps: Interpretation, v. 3/4, p. SAE59-SAE83.

West, B.P., S.R. May, J.E. Eastwood, and C. Rossen, 2002, Interactive seismic facies classification using textural attributes and neural networks: Leading Edge, v. 21/10, p. 1042–1049.

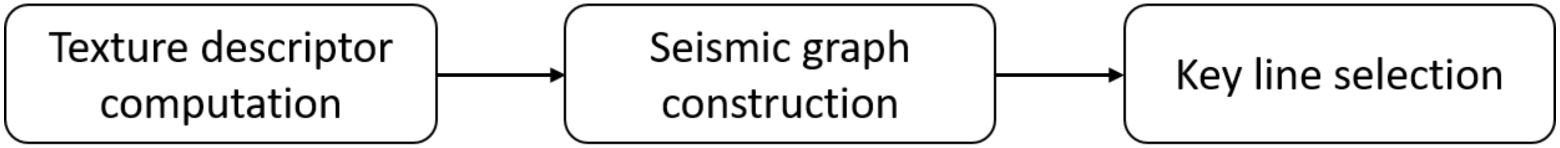


Figure 1. Proposed method.

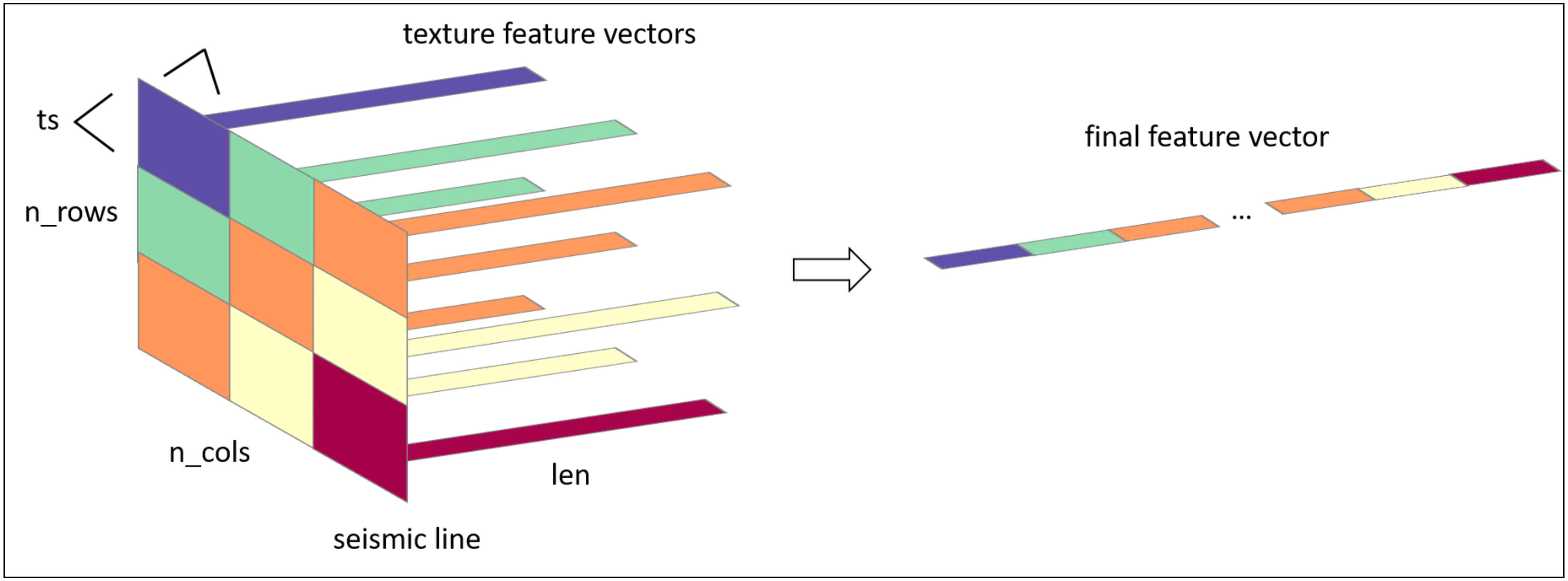
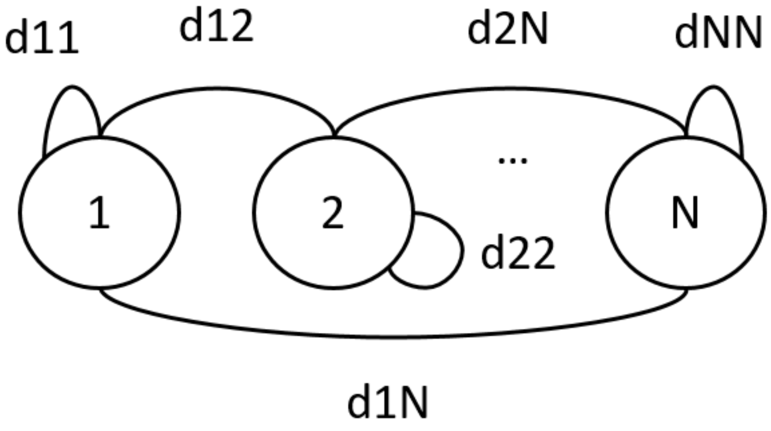
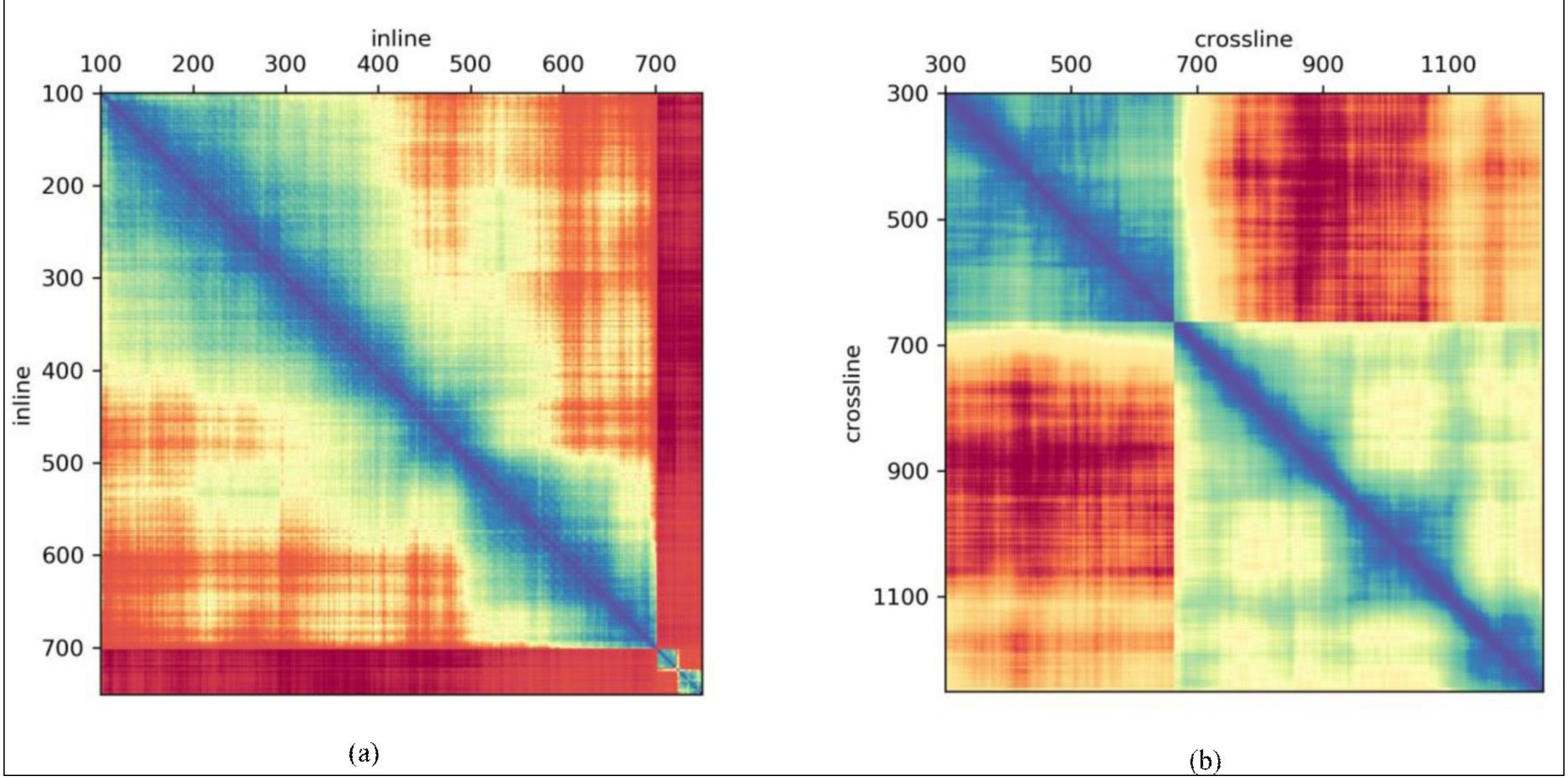


Figure 2. Texture descriptor computation.



	1	**2**	**...**	**N**
1	d11	d12	...	d1N
2	d12	d22	...	d2N
...	...	...	...	...
N	d1N	d2N	...	dNN

Figure 3. A seismic graph and its adjacency matrix.



(a)

(b)

Figure 4. (a) inline and (b) crossline adjacency matrices for tile size 150 with LBP descriptor.

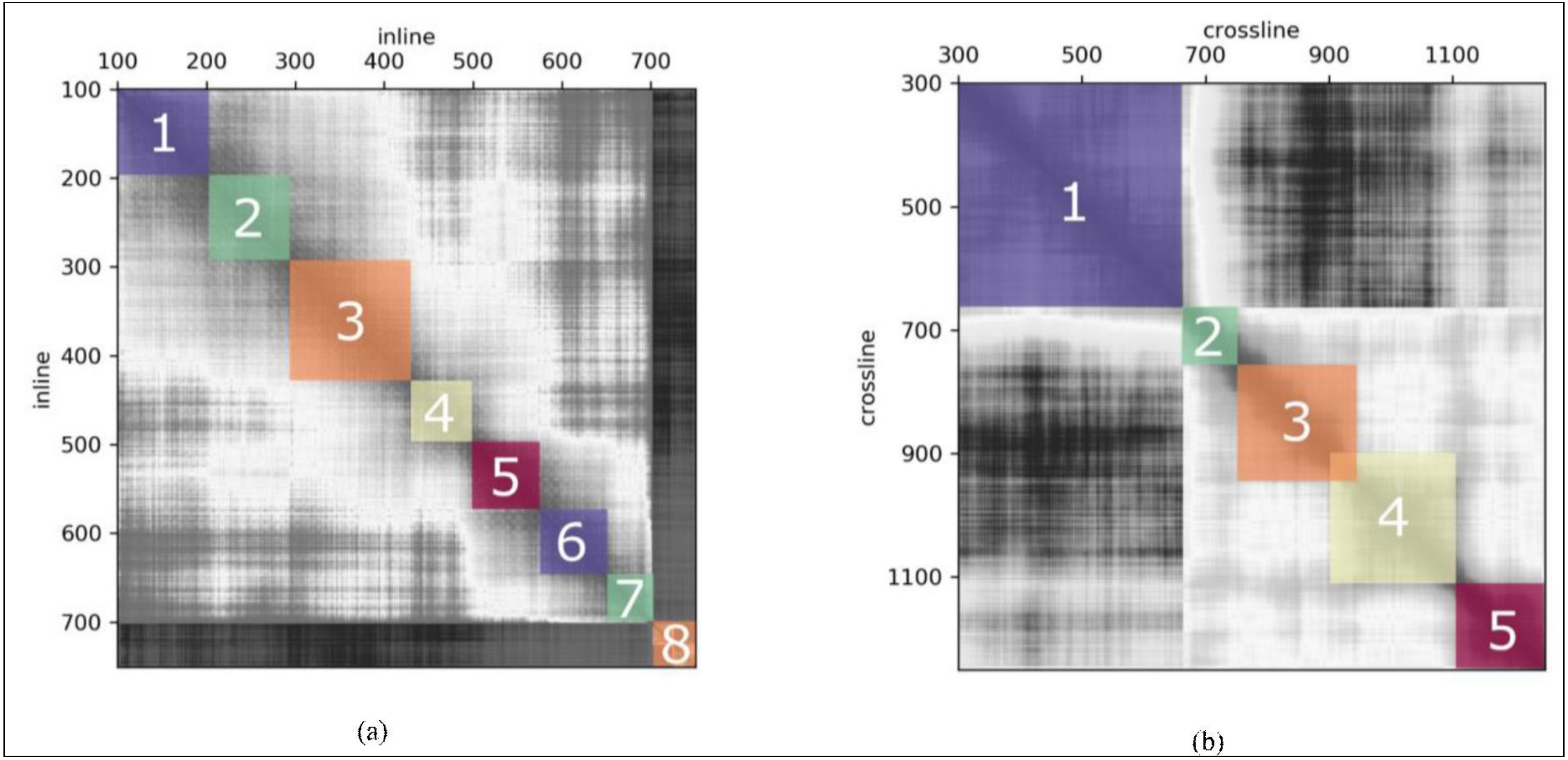


Figure 5. Visual clusters for (a) inline and (b) crossline adjacency matrices.

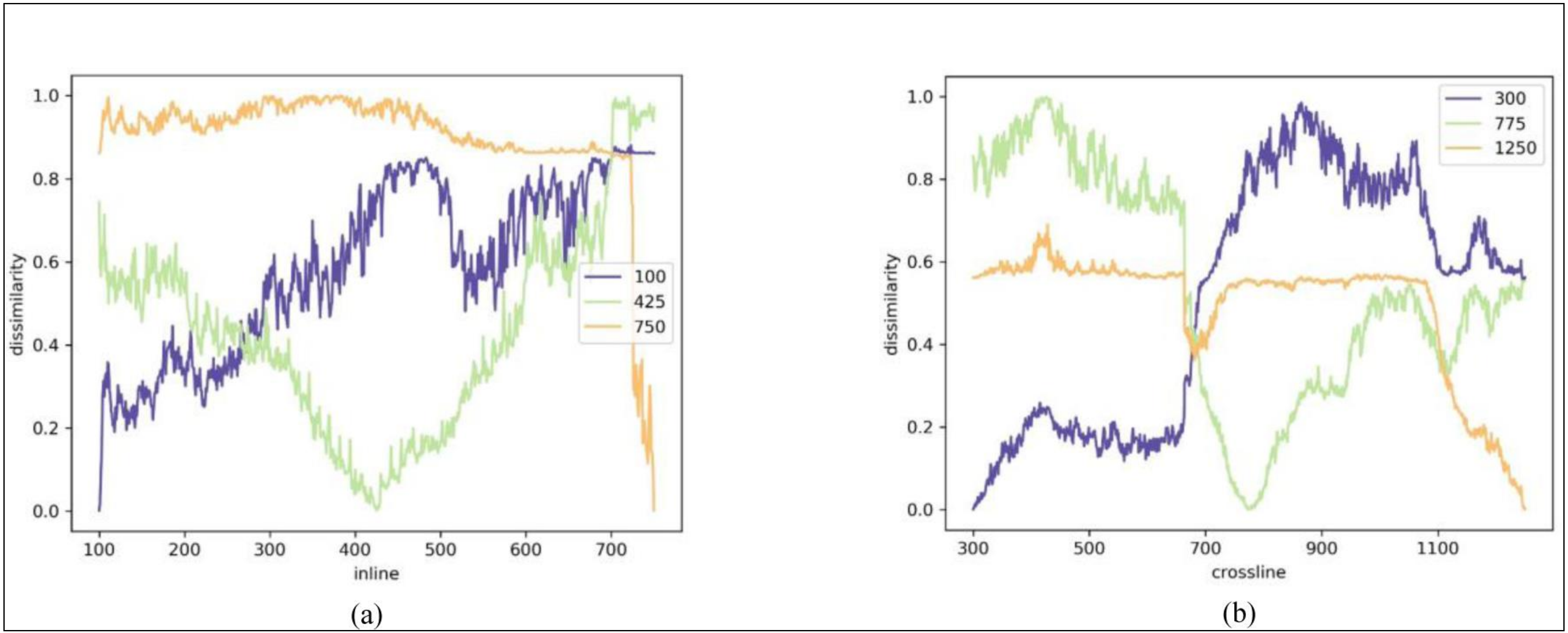


Figure 6. Dissimilarities for (a) inlines 100, 425 and 750 and (b) crosslines 300, 775 and 1250.

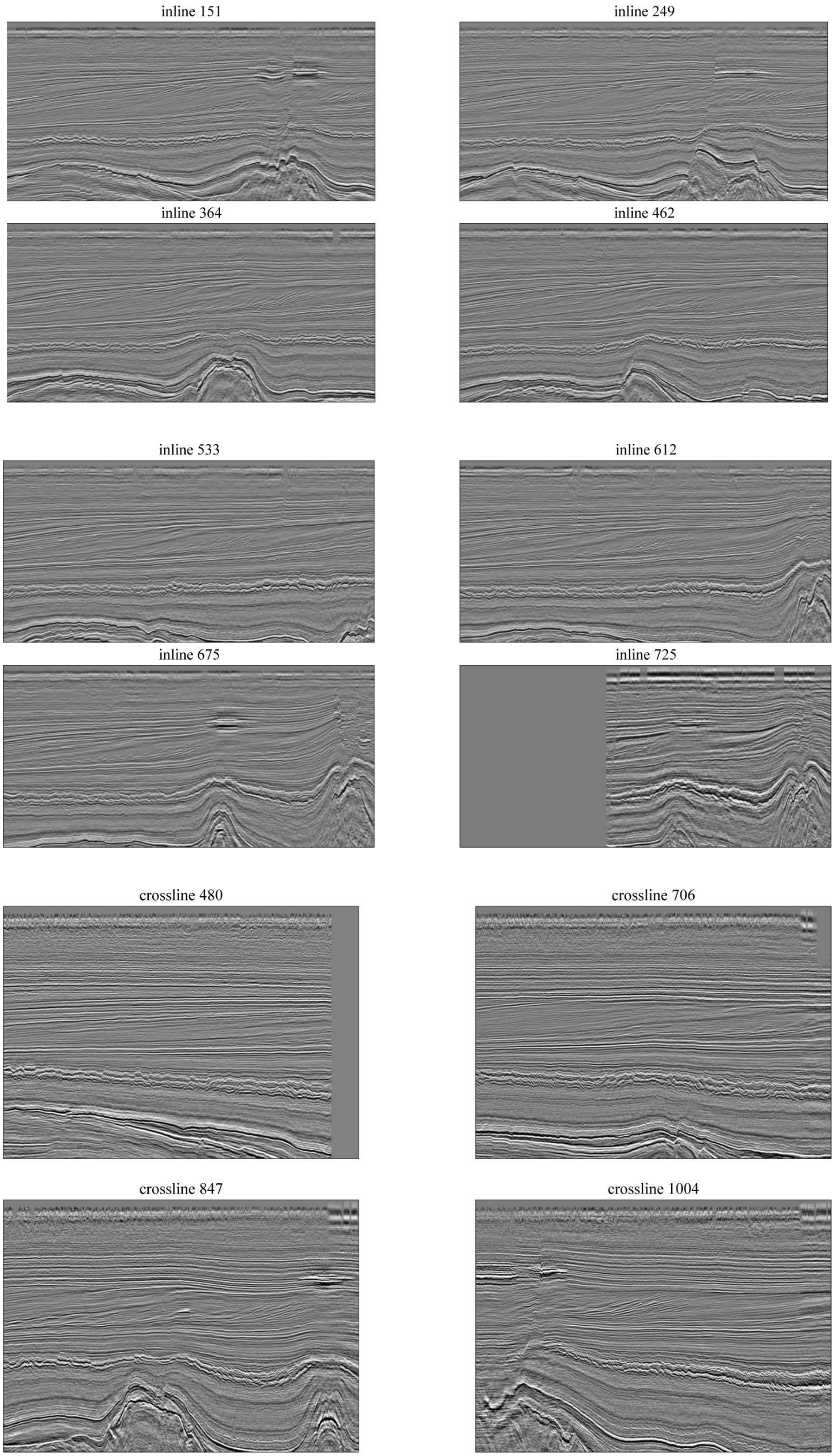
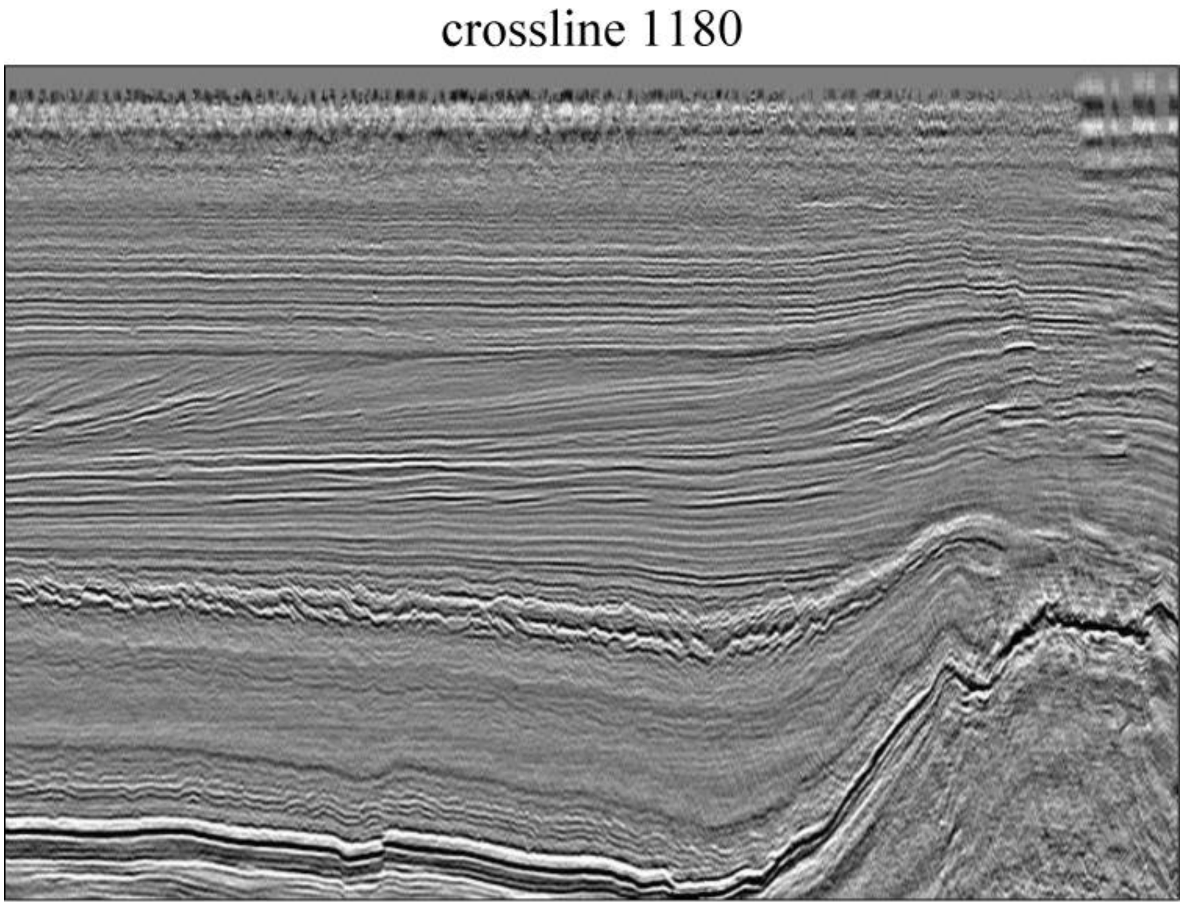


Figure 7. Selected key inlines.



crossline 1180

Figure 8. Selected key crosslines.

inlines			
tile size	50	100	150
number of tiles (n_rows × n_cols)	200	50	28
crosslines			
tile size	50	100	150
number of tiles (n_rows × n_cols)	140	35	20

Table 1. Number of tiles for inlines and crosslines.