

Detection and extraction of fault surfaces in 3D seismic data

Israel Cohen¹, Nicholas Coult², and Anthony A. Vassiliou²

ABSTRACT

We propose an efficient method for detecting and extracting fault surfaces in 3D-seismic volumes. The seismic data are transformed into a volume of local-fault-extraction (LFE) estimates that represents the likelihood that a given point lies on a fault surface. We partition the fault surfaces into relatively small linear portions, which are identified by analyzing tilted and rotated subvolumes throughout the region of interest. Directional filtering and thresholding further enhance the seismic discontinuities that are attributable to fault surfaces. Subsequently, the volume of LFE estimates is skeletonized, and individual fault surfaces are extracted and labeled in the order of decreasing size. The ultimate result obtained by the proposed procedure provides a visual and semantic representation of a set of well-defined, cleanly separated, one-pixel-thick, labeled fault surfaces that is readily usable for seismic interpretation.

INTRODUCTION

Fault surfaces are common subterranean structures that are associated with displacements or offsets of subsurface layers. Their consistent and reliable detection in 3D-seismic data provides an interpreter with very powerful means to quickly visualize and map complex geological structures.

A common tool for facilitating structural and stratigraphic interpretation is the coherency cube, originated by Bahorich and Farmer (1995, 1996). It is calculated from seismic data using a coherency measure that quantifies the seismic discontinuity at each point. Discontinuities that are attributable to fault surfaces include dip, azimuth, and offset changes of seismic reflectors, and waveform and amplitude variations caused by defocusing. Such discontinuities appear on coherence slices as incoherent linear or curved features (e.g., Marfurt et al., 1999; Gersztenkorn et al., 1999; Neff et al., 2000; Lees, 1999).

The most acceptable coherence measures are based on crosscorrelation (Bahorich and Farmer, 1995), semblance (Marfurt et al., 1998), or eigenstructure (Gersztenkorn and Marfurt, 1996a, 1996b; Kirlin, 1992) techniques. These methods typically suffer from a lack of robustness, especially when dealing with noisy data (Marfurt et al., 1999; Gersztenkorn and Marfurt, 1999). Recently, we introduced a multiscale analysis method for the estimation of seismic coherency that is both robust for noise and computationally efficient (Cohen and Coifman, 2002). It involves another measurement, namely, the local structural entropy (LSE), which evaluates the dissimilarity of subvolumes that enclose a given analysis point. Dealing with subvolumes, rather than individual traces, leads to robustness, yet avoids the expensive computations of eigenstructure-based large covariance matrices and eigenvalues.

A major drawback of coherency-based fault analysis is that seismic discontinuities also may be the result of geological features that are unrelated to faults. Furthermore, creating a consistent geological interpretation from large 3D-seismic-data volumes often requires manual intervention, which is time-consuming, tedious, and imprecise.

In this paper, we propose a robust and computationally efficient method for the extraction of fault surfaces in 3D-seismic volumes. The seismic data are transformed into a volume of local-fault-extraction (LFE) estimates, which provides the interpreter with a much clearer visual indication of the fault surfaces. The LFE estimate at a given analysis point is obtained by the following procedure. First, a 3D-analysis cube that is tilted and rotated about the analysis point is selected by the interpreter. The analysis cube moves throughout the seismic volume and outputs a measure of normalized differential entropy (NDE) for each point. The NDE value represents the likelihood of a fault surface whose dip and azimuth are similar to those of the analysis cube to intersect with the analysis point. Subsequently, the local average of the NDE is removed, and portions of fault surfaces, approximately aligned with the analysis cube, are extracted by directional filtering. The filtered NDE coefficients are thresholded and filtered back to produce directional LFE volumes. Next, the LFE attribute is given by the maximal directional LFE, over the presumably tested set of dips and azimuths. This approximately gathers the

Manuscript received by the Editor November 22, 2004; revised manuscript received July 1, 2005; published online July 12, 2006.

¹Technion — Israel Institute of Technology, Department of Electrical Engineering, Haifa, Israel 32000. E-mail: icohen@ee.technion.ac.il.

²GeoEnergy Incorporated, 3000 Wilcrest Drive, Houston, Texas 77042. E-mail: coult@coult.net; avassili@geoenergycorp.com.

© 2006 Society of Exploration Geophysicists. All rights reserved.

significant portions of the fault surfaces into smooth larger surfaces. Finally, the results are skeletonized, and individual faults surfaces are extracted and labeled in the order of decreasing size. A comparison of the LFE volume with the local-structural-entropy attribute (Cohen and Coifman, 2002) shows that the LFE method provides a more reliable and convenient tool for detecting and extracting fault surfaces.

In the following section, we present the fault-extraction algorithm. We begin with a local estimate of faults. Next, we enhance the contrast of these faults to facilitate the analysis of regions that contain dipping layers or are highly discontinuous. Then, we extract the portions of fault surfaces by directional filtering and construct fault surfaces out of these portions. Finally, we demonstrate the applicability of the proposed algorithm to real data.

THE LOCAL FAULT EXTRACTION

In this section we describe the basic components forming the proposed fault-extraction algorithm.

Normalized differential entropy

We begin by subdividing the original 3D-seismic volume into individual 3D data-analysis volumes of $N \times M \times P$ dimension. A

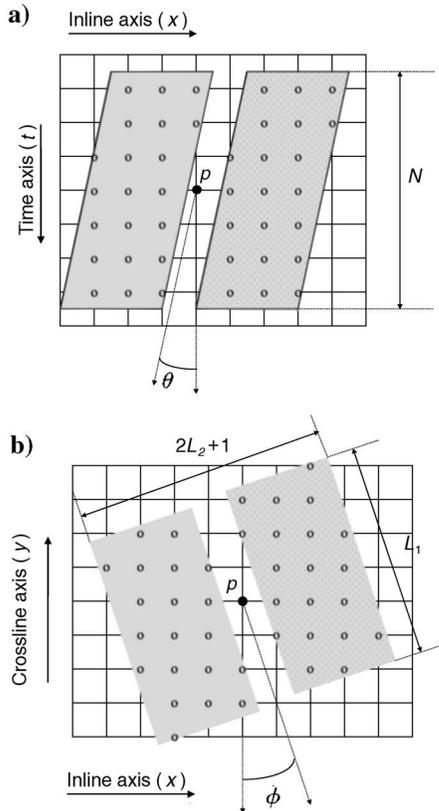


Figure 1. (a) A vertical cross section and (b) horizontal slice that illustrate the geometrical distribution of traces and samples used in the analysis cube. The analysis cube, consisting of two subvolumes, is centered about an analysis point $p = (t, x, y)$ and is defined by the length of major axis L_1 , the length of minor axis $2L_2 + 1$, the time duration N samples, the azimuth ϕ around the inline axis, and the tilt θ from the vertical axis.

typical analysis volume is $41 \times 11 \times 6$ values. The analysis volume is defined by the length L_1 along the major axis, the length $2L_2 + 1$ along the minor axis; the time duration of N samples, the azimuth ϕ (rotation angle around the inline axis), and the tilt θ from the vertical axis (see Figure 1).

The analysis volume is broken into two subvolumes (for instance, a $41 \times 11 \times 6$ analysis cube becomes two $41 \times 11 \times 3$ subvolumes, for a total of 1353 values), which are rotated and tilted around a central analysis point $p = (t, x, y)$. Subsequently, the samples within the respective subvolumes are rearranged consistently into two column vectors $\mathbf{v}_{1,p}(\theta, \phi)$ and $\mathbf{v}_{2,p}(\theta, \phi)$. Note that the subvolumes in Figure 1 have horizontal top and bottom surfaces, which is particularly preferable when the subsurface layers are horizontal or close to horizontal; however, in case there are dipping layers, the top and bottom surfaces can be elevated so that they are parallel to the dominant local seismic orientation within the analysis cube (Cohen et al. 2004).

Rearranging the seismic data within the analysis cube in such a way allows the resultant column vectors to be used directly for edge detection. That is, the fault-surface edges that are defined by a combination of difference of position of the central analysis point p , the analysis-cube size, the rotation ϕ , and the tilt θ are computed from the pairs of the corresponding two-column vectors. The computations, which are carried out through a normalized version of the Prewitt edge-detection filter (Jain, 1989; Lipkin and Rosenfeld, 1970; Luo et al., 1996), are designed to capture the edges and breaks, whether visible or not, that are generated by subterranean faults in the seismic data.

A measure $\mathcal{N}_p(\theta, \phi)$, referred to as normalized differential entropy (NDE), is computed at each analysis point p as a normalized version of the Prewitt filter,

$$\mathcal{N}_p(\theta, \phi) = \frac{\|\mathbf{v}_{1,p}(\theta, \phi) - \mathbf{v}_{2,p}(\theta, \phi)\|_p}{\|\mathbf{v}_{1,p}(\theta, \phi)\|_p + \|\mathbf{v}_{2,p}(\theta, \phi)\|_p}, \quad (1)$$

where $\|\cdot\|_p$ is the ℓ_p norm. Typically we use the ℓ_1 norm, and the metric defined in equation 1 can be referred to as a normalized Manhattan distance. We may observe the following: First, the range of $\mathcal{N}_p(\theta, \phi)$ is $[0, 1]$, where the minimal value (zero) is obtained when the two subvolumes are perfectly correlated without an offset of seismic layers [i.e., $\mathbf{v}_{1,p}(\theta, \phi) = \mathbf{v}_{2,p}(\theta, \phi)$], and the maximal value of $\mathcal{N}_p(\theta, \phi)$ is obtained for maximally offset correlated subvolumes [i.e., $\mathbf{v}_{1,p}(\theta, \phi) = -\mathbf{v}_{2,p}(\theta, \phi)$]. Second, if the subsurface layers have mostly horizontal orientation without significant lateral elastic-impedance contrasts, then the NDE measure produces an excellent edge detector and fault-surfaces indicator, but a very poor indicator of subsurface-layer interfaces. Third, even when the dominant layer orientation is not horizontal, by using top and bottom analysis-cube surfaces that are parallel to the dominant local 3D-seismic orientation (Cohen et al., 2004), we obtain a useful 3D-edge detector, which is a good indicator of fault surfaces and poor indicator of subsurface layer interfaces. Fourth, the data structure used for the NDE and its 3D-edge-detection computational architecture is completely different from the data structures and the computational setup in the 3D-seismic coherence and variance methods (e.g., Marfurt et al., 1999; Bahorich and Farmer, 1996; Gersztenkorn et al., 1999). Fifth, when the top and bottom analysis-cube surfaces are horizontal, there is a very significant overlapping of the analysis cubes for any dip and azimuth combination; accordingly, the computation of $\mathcal{N}_p(\theta, \phi)$ and $\mathcal{N}_{p'}(\theta', \phi')$ for nearby analysis points p and p' involves many re-

peatable difference computations. Consequently, all possible differences for the NDE computations are computed up front for all combinations of the dip and azimuth that are used, which yields significant acceleration in the NDE computations. We have found from practical experience that a set of 12 dips from -35° to 35° with a 5° increment excluding the dips of -5° , 0° , and 5° , and a set of 12 to 15 uniformly spaced azimuth values are sufficient for a $41 \times 6 \times 7$ NDE analysis cube. By using the up-front NDE local-difference computations for all combinations of dip and azimuth, we obtain a computational speedup of approximately a factor of 20, which is extremely important for efficient computation and extraction of the fault surfaces.

Contrast enhancement

The second step of the fault-extraction algorithm is contrast enhancement, which is applied to single NDE volumes, i.e., per postulated orientation. Patches of fault surfaces, which are approximately characterized by the dip and azimuth of the analysis cube, are identified by a local increase in the NDE values. Accordingly, we apply contrast-enhancement filtering to single NDE volumes, i.e., per postulated orientation, and set to zero negative values. Contrast enhancement facilitates the analysis of regions that contain dipping layers or are highly discontinuous.

Contrast enhancement can be efficiently implemented by using a discrete ‘‘Mexican hat’’ function, which corresponds to local-average subtraction:

$$f(n) = C(1 - n^2T^2)\exp(-n^2T^2/2), \quad (2)$$

where T is the sampling period of the Mexican-hat function, n is an integer, and C is a normalization constant, such that $\sum_{n=-\infty}^{\infty} |f(n)| = 2$. We use a finite-length filter ($-4.5 \leq nT \leq 4.5$) that contains an odd number of uniformly spaced coefficients. In general, filter length selection depends on the size of the analysis cube and the thickness of the fault surfaces. The filtered NDE by the Mexican-hat function is computed as

$$\bar{N}_p(\theta, \phi) = g_p(\theta, \phi) * \mathcal{N}_p(\theta, \phi) = \sum_{p'} g_{p-p'}(\theta, \phi) \mathcal{N}_{p'}(\theta, \phi), \quad (3)$$

where $g_p(\theta, \phi)$ is a 3D version of f that is rotated around a central analysis point $p = (t, x, y)$, such that its main axis is perpendicular to the slabs of the analysis cube (i.e., rotated by azimuth ϕ around the inline axis and tilted by θ from the vertical axis). The contrast-enhanced NDE is given by

$$\hat{\mathcal{N}}_p(\theta, \phi) = \max\{\bar{N}_p(\theta, \phi), 0\}. \quad (4)$$

Directional LFE

The third step of the fault-extraction algorithm uses 3D directional filtering on single contrast-enhanced NDE volumes, i.e., per postulated orientation. Directional filtering enhances the portions of fault surfaces that are approximately aligned with the analysis cube.

The directional filter, denoted by $h_p(\theta + \alpha, \phi)$, is a 3D ellipsoid that is tilted by $\theta + \alpha$ with respect to the time axis, rotated by ϕ with respect to the inline axis, and normalized such that $\sum_p h_p(\theta, \phi) = 1$. Its dimensions, selected by the interpreter, control the minimal dimensions of the detected subsurfaces. The maximum value of α is

determined by the tilt increment Δ_θ ($|\alpha| < \Delta_\theta/2$). The implementation uses a 3D pancakelike Hann window that is aligned so that it is parallel to the division in the analysis cube. A possible set of dimensional values for this 3D pancakelike window is 61 samples at its major axis and three samples at its minor axes. A possible value for the tilt increment is $\Delta_\theta = 5^\circ$, and the relative tilt of the directional filter α can be restricted to values in the set $\{-2^\circ, 0, 2^\circ\}$. Alternatively, a smaller tilt increment could be used and the relative tilt could be discarded (i.e., $\alpha \equiv 0$); however, the above formulation has been found to be computationally more efficient.

Directional filtering of the contrast-enhanced NDE yields

$$C_p(\theta + \alpha, \phi) = \sum_{p'} h_{p-p'}(\theta + \alpha, \phi) \hat{\mathcal{N}}_{p'}(\theta, \phi), \quad (5)$$

where the summation in equation 5 is over the set of points $p' = (t', x', y')$, which are in the proximity of $p = (t, x, y)$ [i.e., $p - p'$ is in the support of $h_{p-p'}(\theta + \alpha, \phi)$]. The resultant coefficients are thresholded by δ ($0 < \delta < 1$),

$$\tilde{C}_{p'}(\theta + \alpha, \phi) = \begin{cases} C_{p'}(\theta + \alpha, \phi), & \text{if } C_{p'}(\theta + \alpha, \phi) \geq \delta \\ 0, & \text{otherwise,} \end{cases} \quad (6)$$

and then filtered back to produce the directional LFE values, determined by

$$\mathcal{L}_p(\theta, \phi) = \sum_{p', \alpha} \tilde{C}_{p'}(\theta + \alpha, \phi) h_{p-p'}(\theta + \alpha, \phi). \quad (7)$$

The directional LFE volumes contain significant portions of fault surfaces that are characterized by roughly the same dip and azimuth orientations as those of the analysis cube. Subsequently, at each point of the 3D volume, we keep the maximum directional LFE value over the tested set of tilts and azimuths. Specifically, the LFE attribute at the analysis point p is obtained by

$$\hat{\mathcal{L}}_p = \max_{\theta, \phi} \{\mathcal{L}_p(\theta, \phi)\}. \quad (8)$$

Equipped with the tilt and azimuth arguments, $\theta(p)$ and $\phi(p)$, which yield the maximum in equation 8 for each particular analysis point p , the LFE volume is further enhanced by 3D skeletonization and 3D surface separation. The directional LFE performs best on 3D-seismic volumes that do not have fault shadows. Extensive presence of fault shadows can generate small fault surfaces that cross true faults in an X-pattern.

Skeletonization

The fourth step of the fault-extraction algorithm, skeletonization, is very important, both for filtering out very small faults or faultlike features and for separating the different fault surfaces. Skeletonization algorithms have been used in image processing for decades (e.g., Blum, 1967; Pavlidis, 1980; Stentiford and Mortimer, 1983; Zhang and Fu, 1984; Zhang and Suen, 1984). Here, we present a skeletonization algorithm for 3D data that is designed particularly for the results of the filtering step that is outlined above.

The skeletonization algorithm performs the following computations on the results of the filtering step:

- 1) For each horizontal slice t , it carries out a 2D binarization and skeletonization using the adaptive image-binarization-and-skeletonization algorithm given in Appendix A.
- 2) For each vertical slice x , it performs 2D skeletonization and

stretches the skeletons, as described in the adaptive image-binarization-and-skeletonization algorithm (A).

- 3) It repeats step 2 for vertical slice y , horizontal slice t , vertical slice x , and so on, until there is no change in the result or a pre-specified limit of iterations has been reached. Each iteration further stretches the skeletons, so that gaps in the fault surfaces are bridged and the resultant surfaces become more complete.

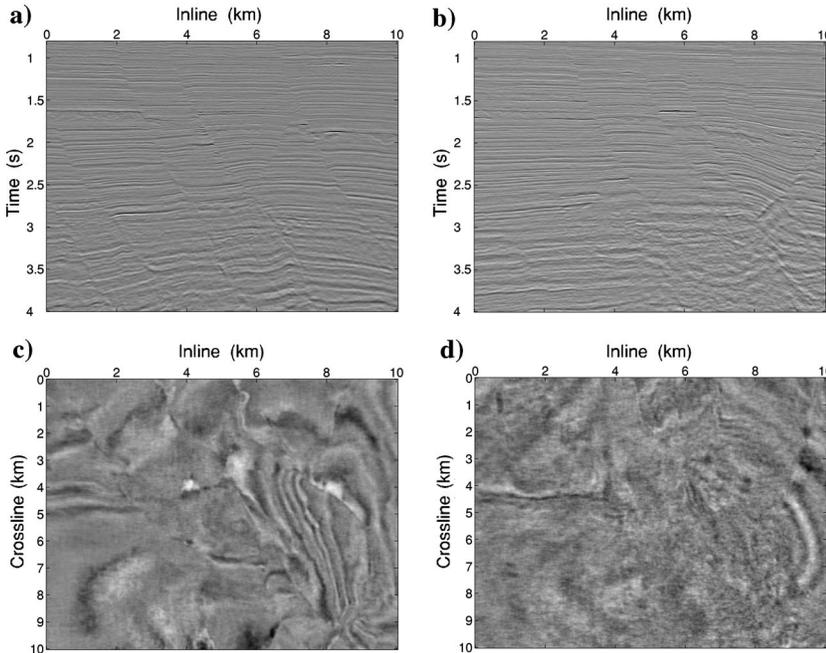


Figure 2. Vertical cross sections and horizontal slices through the seismic data: (a) vertical cross section at $y = 1.3$ km; (b) vertical cross section at $y = 8$ km; (c) horizontal slice at $t = 2.4$ s; and (d) horizontal slice at $t = 3.2$ s.

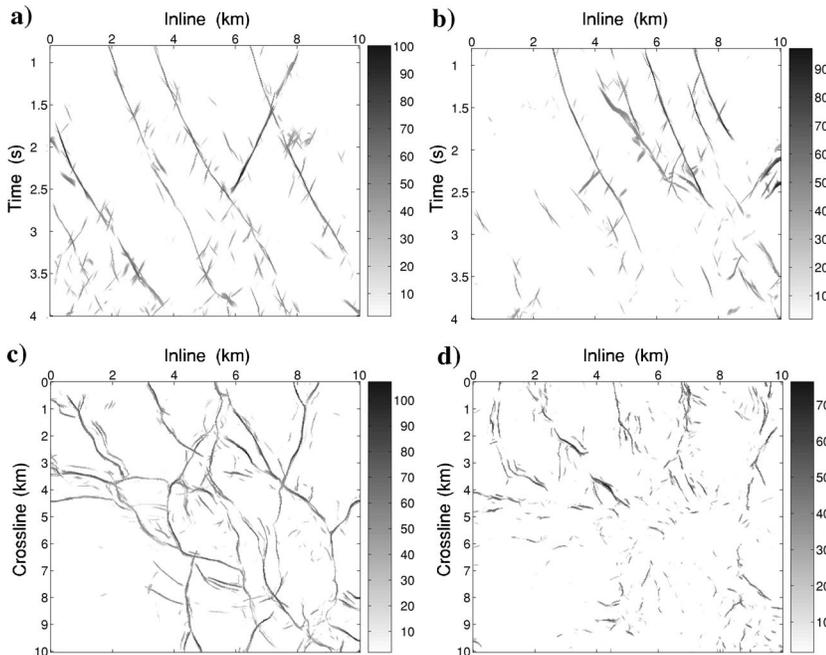


Figure 3. Vertical cross sections and horizontal slices through the skeletonized LFE volume: (a) vertical cross section at $y = 1.3$ km; (b) vertical cross section at $y = 6$ km; (c) horizontal slice at $t = 2.4$ s; and (d) horizontal slice at $t = 3.2$ s.

Labeling of individual fault surfaces

The final step of the fault-extraction algorithm is the separation of the individual fault surfaces and labeling of the resultant objects in order of decreasing size.

For each azimuth, we extract the data that are associated with that azimuth and concatenate them along the fourth dimension, ordering them according to the azimuth value, thus producing 4D data that contain the azimuth information, as well. Subsequently, we label the newly produced 4D data and re-create the 3D labeled data by taking the maximum along the fourth dimension. This creates a labeled version of the 3D data, such that the objects are distinguished by their azimuths, as well, and therefore are labeled differently if their azimuths are not close to one another. Next, we identify objects whose size is above a certain threshold, and relabel these objects in order of decreasing size. (The remaining objects are removed.)

RESULTS

In this section, we use a real-data example to demonstrate the applicability of the LFE algorithm and to illustrate its execution. The data are decimated in time and space. The time interval is 4 ms; inline and crossline trace spacing is 25 m. A small subvolume of 401×401 traces and 1001 samples is used for demonstration (10-km inline and crossline distances and 4-s duration).

Figure 2 shows vertical cross sections and horizontal slices through the seismic data. Figure 3 displays the corresponding vertical cross sections and horizontal slices through the skeletonized LFE volume. For the current example, we used a 3D pancakelike Hann filter whose dimensions are 61 samples at its major axis and three samples at its minor axes. The tilt increment is $\Delta_\theta = 5^\circ$, the azimuth increment is $\Delta_\phi = 45^\circ$, and the relative tilt of the directional filter α is restricted to three values $\{-2^\circ, 0, 2^\circ\}$. The filtered NDE coefficients are thresholded by $\delta = 0.12$ and filtered back to produce the directional LFE volumes. The skeletonized LFE volume (Figure 3) contains fault surfaces that are consistent with the presumed model (i.e., the dimensions of the analysis cube, set of dips and azimuths, directional filter, threshold,

etc.). The existence of fault shadows in the prestack-time-migrated volume causes small-scale X-pattern faults to be generated. Hence, it is generally preferable to apply LFE on prestack-migrated data that have minimal fault shadows. Figure 4 shows the final result of the fault-extraction logarithm which includes labeling of distinct skeletons in order of decreasing size.

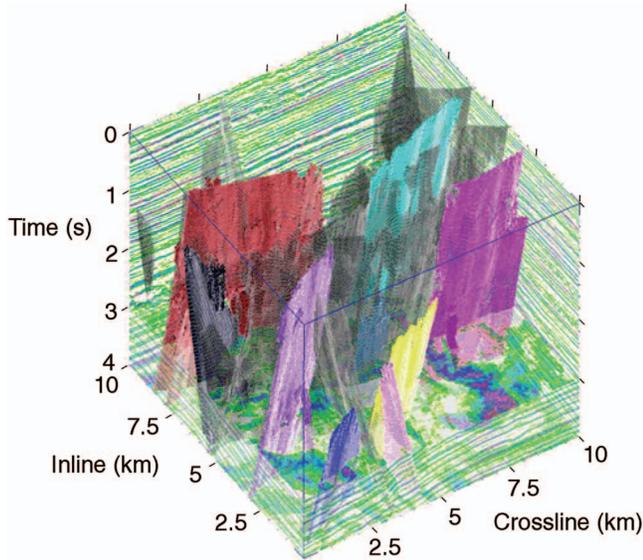


Figure 4. Three-dimensional visualization of the original seismic data and corresponding color-coded fault surfaces. Different colors correspond to different fault-surface numbers. The fault-surfaces are robustly extracted, separated, and labeled.

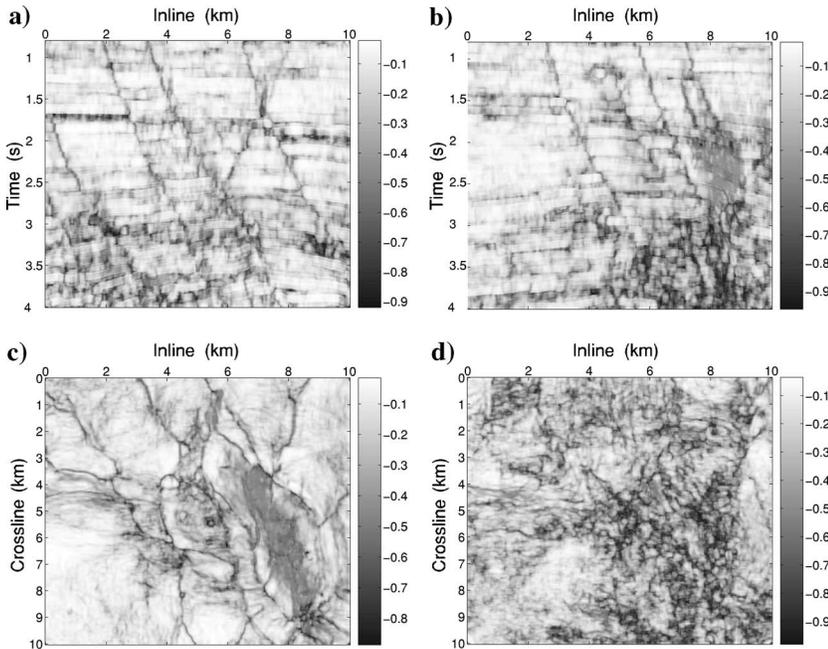


Figure 5. Vertical cross sections and horizontal slices through the LSE volume: (a) vertical cross section at $y = 1.3$ km; (b) vertical cross section at $y = 6$ km; (c) horizontal slice at $t = 2.4$ s; and (d) horizontal slice at $t = 3.2$ s.

For comparison with the LSE attribute (Cohen and Coifman, 2002), in Figure 5 we plot vertical cross sections and horizontal slices through the LSE volume that was obtained by using analysis cubes of 6×6 traces and 21 samples. Values are mapped to shades of gray, where darker shades indicate greater discontinuity. A comparison of Figures 5 and 3 shows that the proposed method provides a more reliable and convenient tool for the extraction and separation of fault surfaces. Furthermore, faults along strike that are difficult to see in the LSE volume are very clear in the skeletonized LFE volume.

CONCLUSION

A major component of structural interpretation is the identification, location, and extraction of individual fault surfaces. Fault surfaces are very important in hydrocarbon exploration because they are directly related to hydrocarbon accumulation and hydrocarbon flow paths. Extraction of individual fault surfaces from seismic data is a largely qualitative procedure and thus has been characterized by the need for careful human data interpretation.

In recent years, there has been progress in visualizing stratigraphic and structural discontinuities with the coherence or variance methods, which look at the similarity or dissimilarity of a small number of neighboring traces to determine these discontinuities. However, the existing methods have a limited precision and generally are inefficient in isolating fine details, such as fault surfaces or sedimentary-layer interfaces that may be only one pixel wide. The proposed fault-extraction algorithm facilitates the extraction, separation, and labeling of the fault surfaces by identifying portions of the surfaces and combining them into large, distinct fault surfaces. Specifically, the portions of the fault surfaces are identified by analyzing tilted and rotated subvolumes throughout the region of interest. Applying 3D contrast enhancement and directional filtering enables the analysis of regions that contain dipping layers or are highly discontinuous. Skeletonizing the results of the filtering operation, separating the individual fault surfaces, and finally, labeling the skeletons in the order of decreasing size yields a useful representation for a 3D visualization of subterranean features.

APPENDIX A

ADAPTIVE IMAGE BINARIZATION AND SKELETONIZATION

The steps of the adaptive image-binarization-and-skeletonization algorithm are:

- 1) Define two thresholds: low and high. Define the high threshold so that the high-threshold binarization will remove low-intensity objects yet will not completely erase objects that should remain in the image. Define the low threshold so that connectivity between two close objects will be gained, yet no unnecessary pixels will be marked as "1."
- 2) Binarize the original image using the high threshold.

- 3) Skeletonize the binary image as follows:
 - a) Using an eight-neighborhood notation as shown in Figure A-1a, let $N(p_1)$ denote the number of nonzero neighbors of the center point p_1 , and let $T(p_1)$ denote the number of 0-1 transitions in the ordered sequence $p_2, p_3, \dots, p_8, p_9, p_2$. Compute $N(p)$ and $T(p)$ for all points p in the image.
 - b) Flag type-I border points in the image, where a point p_1 is flagged if the following conditions are satisfied:
 - i) $2 \leq N(p_1) \leq 6$.
 - ii) $T(p_1) = 1$.
 - iii) $p_5 = 0$ or $p_7 = 0$ or $(p_3 = 0$ and $p_9 = 0)$.
 - c) Delete all type-I border points.
 - d) Flag type-II border points in the image, where a point p_1 is flagged if the following conditions are satisfied:
 - i) $2 \leq N(p_1) \leq 6$.
 - ii) $T(p_1) = 1$.
 - iii) $p_3 = 0$ or $p_9 = 0$ or $(p_5 = 0$ and $p_7 = 0)$.
 - e) Delete all type-II border points.
 - f) Iterate deletion of type-I and type-II points using steps 3b–e until the image contains no more such border points.
- 4) For each pixel in the skeletonized image that is marked as “1” and has no neighbors marked as “1” or has only one neighbor marked as “1” (edge points), do the following:
 - a) For a pixel that has no neighbors marked as “1,” find the pixel with the maximum value among the eight-connected neighborhood of the corresponding pixel in the origi-

nal image. For a pixel that has only one neighbor marked as “1,” find the pixel with the maximum value among the pixels in the original image that belong to the eight-connected neighborhood of the pixel and are not too close to the neighbor pixel that is marked as “1.” Figures A-1b and A-1c show the locations of the not-too-close pixels. Figure A-1b refers to the possibility that the neighbor marked as “1” belongs to the four-connected neighborhood, and Figure A-1c refers to the possibility that the neighbor marked as “1” is on the corner of the eight-neighborhood.

- b) If this maximum value is above the low threshold, mark the corresponding pixel in the skeletonized image as “1.” Otherwise, look for a pixel that is above the low threshold in the 5×5 neighborhood in a similar manner. If such a pixel exists, mark the corresponding pixel in the skeletonized image as “1,” and mark the appropriate pixel in the 3×3 neighborhood so that connectivity will be retained.
- c) If the newly marked pixel (the outer pixel in the case of 5×5 neighborhood) has only one neighbor marked as “1,” go to step 4a in the algorithm for the newly marked pixel.

It is important to note that the skeletonized image is updated as necessary at the various steps of the algorithm, and that every step in the algorithm uses the previously updated skeletonized image. Furthermore, steps 4a–c in the algorithm can be implemented as a recursive function, which is recursively called as necessary in step 4c of the algorithm.

ACKNOWLEDGMENTS

The authors thank Ronald R. Coifman of Yale University for valuable discussions and helpful suggestions. We also thank associate editor Kurt J. Marfurt and the three anonymous reviewers who provided valuable comments that considerably improved the quality of this paper.

REFERENCES

Bahorich, M. S., and S. L. Farmer, 1995, 3-D seismic discontinuity for faults and stratigraphic features: The Leading Edge, **14**, 1053–1058.
 ———, 1996, Methods of seismic signal processing and exploration: U. S. Patent 5 563 949.
 Blum, H., 1967, A transformation for extracting new descriptors of shape, in Weiant Whaten-Dunn, ed., Symposium models for speech and visual form: MIT Press.
 Cohen, I., and R. R. Coifman, 2002, Local discontinuity measures for 3-D seismic data: Geophysics, **67**, 1933–1945.
 Cohen, I., A. Vassiliou, and N. Coult, 2004, Method and apparatus for seismic feature extraction: U. S. Patent WO2004044615.
 Gersztenkorn, A., and K. J. Marfurt, 1996a, Coherence computations with eigenstructure: 58th Conference and Technical Exhibition, EAGE, Extended Abstracts, X031.
 ———, 1996b, Eigenstructure based coherence computations: 66th Annual International Meeting, SEG, Expanded Abstracts, 328–331.
 ———, 1999, Eigenstructure-based coherence computations as an aid to 3-D structural and stratigraphic mapping: Geophysics, **64**, 1468–1479.
 Gersztenkorn, A., J. Sharp, and K. J. Marfurt, 1999, Delineation of tectonic features offshore trinidad using 3-D seismic coherence: The Leading Edge, **18**, 1000–1008.
 Jain, A. K., 1989, Fundamentals of digital image processing: Prentice Hall, Inc.
 Kirlin, R. L., 1992, The relationship between semblance and eigenstructure velocity estimators: Geophysics, **57**, 1027–1033.
 Lees, J. A., 1999, Constructing faults from seed picks by voxel tracking: The Leading Edge, **18**, 338–340.
 Lipkin, B. S., and A. Rosenfeld, eds., 1970, Picture processing and psychopictorics: Academic Press, Inc.

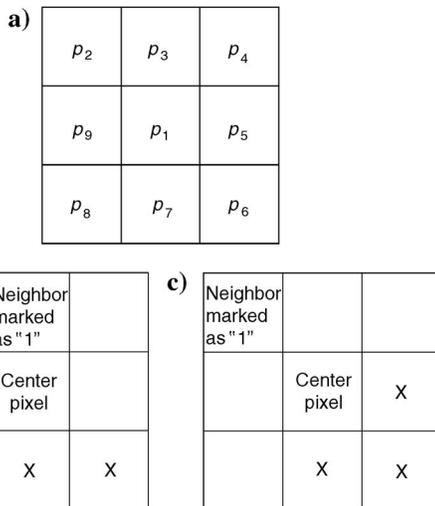


Figure A-1. (a) Eight-neighborhood notation that is used for binary image skeletonization. (b) and (c) Eight-neighborhood for a pixel that has only one neighbor marked as “1.” The neighbor marked as “1” either belongs to the four-neighborhood (b) or is located on the corner of the eight-neighborhood (c). The locations of pixels that are considered not too close to the neighbor pixel marked as “1” are depicted by “X.”

- Luo, Y., W. G. Higgs, and W. S. Kowalik, 1996, Edge detection and stratigraphic analysis using 3D seismic data: 66th Annual International Meeting, SEG, Expanded Abstracts, 324–327.
- Marfurt, K. J., R. L. Kirlin, S. L. Farmer, and M. S. Bahorich, 1998, 3-D seismic attributes using a semblance-based coherency algorithm: *Geophysics*, **63**, 1150–1165.
- Marfurt, K. J., V. Sudhaker, A. Gersztenkorn, K. D. Crawford, and S. E. Nissen, 1999, Coherency calculations in the presence of structural dip: *Geophysics*, **64**, 104–111.
- Neff, D. B., J. R. Grismore, and W. A. Lucas, 2000, Automated seismic fault detection and picking: US Patent 6 018 498.
- Pavlidis, T., 1980, A thinning algorithm for discrete binary images: *Computer Graphics and Image Processing*, **13**, 142–157.
- Stentiford, F. W. M., and R. G. Mortimer, 1983, Some new heuristics for thinning binary handprinted characters for OCR: *IEEE Transactions on Systems, Man and Cybernetics*, **13**, 81–84.
- Zhang, S., and K. S. Fu, 1984, A thinning algorithm for discrete binary images: First International Conference on Computers and Applications, Institute of Electrical and Electronics Engineers Inc., Computer Society Press.
- Zhang, T. Y., and C. Y. Suen, 1984, A fast parallel algorithm for thinning digital patterns: *Communications of the ACM*, **27**, 236–239.