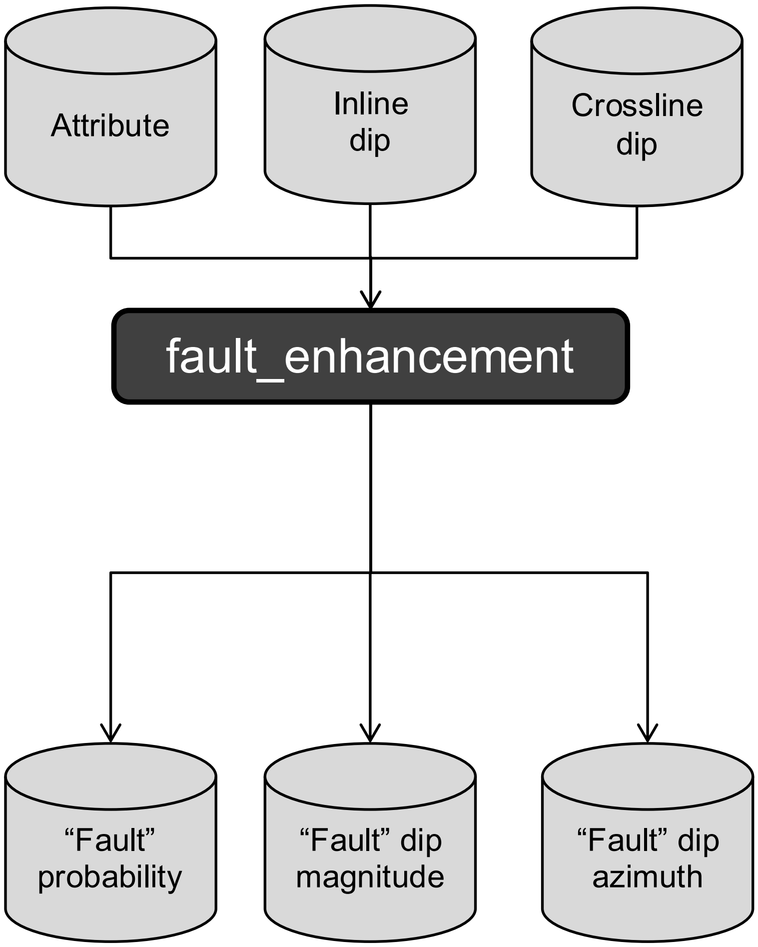
Enhancing faults and axial planes – Program fault_enhancement

Overview

The fault enhancement attribute is a post-stack attribute which enhances locally planar features within a seismic attribute volume. We suspect the most common application will be to improve fault images previously approximated by a similarity attribute. However, through proper choice of parameters, one can also enhance unconformities and other discontinuities parallel or subparallel to reflector dip. This algorithm will also enhance axial planes delineated by most-positive and most-negative curvature volumes. In addition to sharpening hypothesized faults, fault_enhancement also generates ancillary fault dip magnitude and fault dip azimuth volumes.

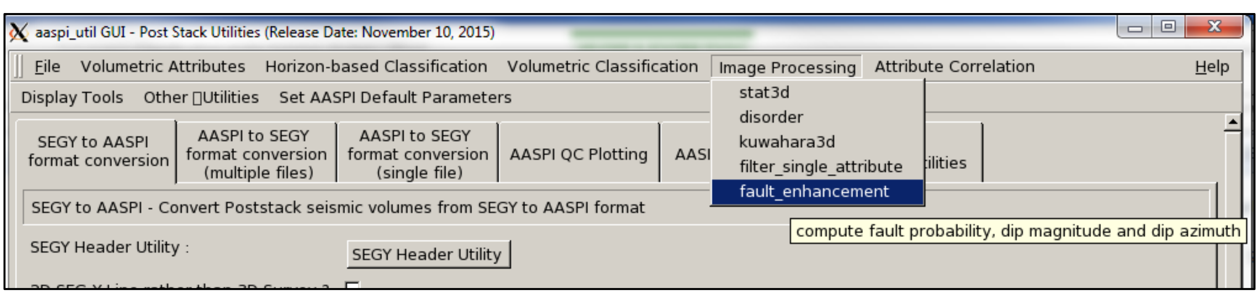
Flow Chart

The input to program fault_enhancement include a primary attribute that approximates the faults or axial planes that you wish to enhance. For faults, this will usually be one of the similarity/coherence attributes computed using program similarity3d. The fault enhancement is done in a 3D window, For axial planes, this will usually be the most-positive or most-negative principal curvatures. Program fault_enhancement will allow the user to suppress or enhance attribute features with respect to reflector dip. For this reason, the inline and crossline components of reflector dip are additional input volumes.

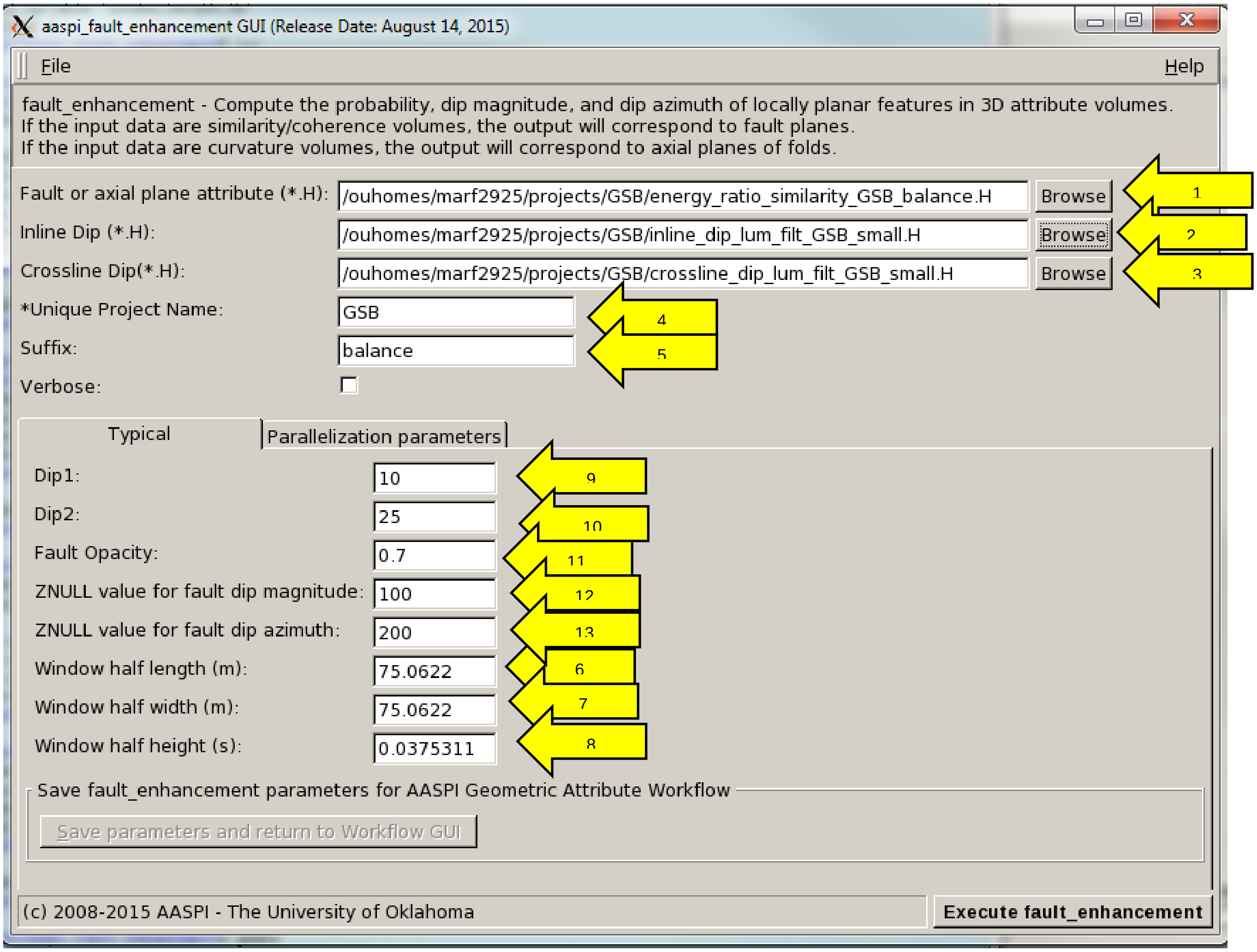


How to run

The fault enhancement program is located under the Image Processing tab -> filter_single_attribute of the main aaspi_util window:



and will appear as:



The user needs to specify (1) the input seismic attribute volume, (2) the inline dip, (3) the crossline dip, (4) a unique project name, and (5) a suffix. Once the inline dip component is browsed, the lower three parameters (window length, width, and height) will be automatically filled and can be modified. For time domain data, the value of velocity used in program dip3d is used to estimate a vertical window of comparable size to the window length and width. In this example from the Great South Basin (courtesy of NZPM for public use), the cdp spacing, dcdp=12.5 m and the line spacing dline=25 m. The current default is to use a window whose radius is 3 times the greater of these two spacings. In the case above, the (6) window half length and (7) half width are approximately 75 m while the (8) window half height is 37 ms.

The parameters (9) Dip 1 and (10) Dip 2 define a Tukey filter that rejects fault attributes that fall beyond Dip1 and retains fault features beyond Dip2.

If the numerical value of Dip1 > Dip2, then fault features subparallel to reflector dip are retained rather than suppressed. For example, if Dip 1= 25 and Dip 2= 35^{o} then all the discontinuities with a dip greater than 35^{o} will be rejected, while if Dip1=65^{o} and Dip2= 75^{o} then anything with an apparent dip lower than 65^{o} will be rejected. In general, every voxel in the volume will have a valid fault dip magnitude and azimuth. If the fault attribute (probability) is small, these values are meaningless. For this reason we define a cutoff (11) Fault Opacity value, below which the fault dip magnitude and fault dip azimuth are set to be user defined (12 and 13) znull values.These znull values may depend on your interpretation workstation software. The znull value for each volume will also be stored in the output fault dip magnitude and fault dip azimuth *.H files.

The parallelization parameters are identical to those in all other AASPI programs running under MPI.

Clicking the Execute fault_enhancement button on the lower right submits the program.

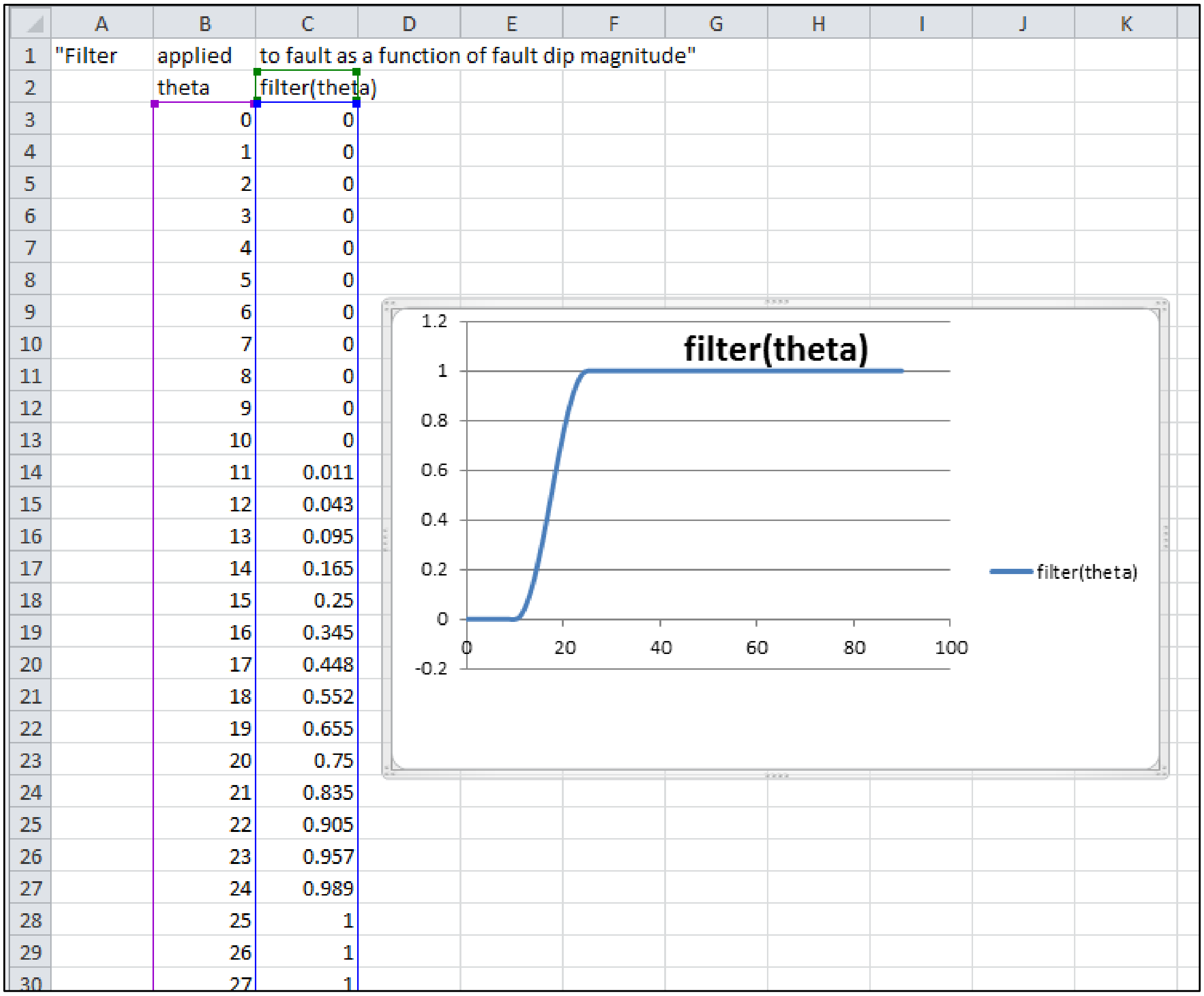
The following files were generated for the parameters chosen above:

```
-rw-r--r--. 1 marf2925 faculty  837 Aug 17 12:51 fault_enhancement.parms
-rw-r--r--. 1 marf2925 faculty 2915 Aug 17 12:51 fault_dip_filter.txt
-rw-r--r--. 1 marf2925 faculty 2732 Aug 17 12:51 fault_probability_GSB_balance.H@@
-rw-r--r--. 1 marf2925 faculty 6895 Aug 17 12:51 fault_probability_GSB_balance.H
-rw-r--r--. 1 marf2925 faculty 2734 Aug 17 12:51 fault_dip_magnitude_GSB_balance.H@@
-rw-r--r--. 1 marf2925 faculty 6946 Aug 17 12:51 fault_dip_magnitude_GSB_balance.H
-rw-r--r--. 1 marf2925 faculty 2732 Aug 17 12:51 fault_dip_azimuth_GSB_balance.H@@
-rw-r--r--. 1 marf2925 faculty 6938 Aug 17 12:51 fault_dip_azimuth_GSB_balance.H
```

The fault_enhancement.parms file simply provides parameters to the python script and reads as follows:

```
marf2925@ediacaran:~/projects/GSB$ cat fault_enhancement.parms
use_mpi=y
processors_per_node=40
node_list="localhost"
build_lsf_script=n
max_run_time=10
batch_queue=""
unique_project_name="GSB"
suffix="balance"
verbose=n
inline_dip_fn="/ouhomes/marf2925/projects/GSB/inline_dip_lum_filt_GSB_small.H"
crossline_dip_fn="/ouhomes/marf2925/projects/GSB/crossline_dip_lum_filt_GSB_small.H"
attribute_fn="/ouhomes/marf2925/projects/GSB/energy_ratio_similarity_GSB_balance.H"
fault_probability_fn="fault_probability_GSB_balance.H"
fault_dip_azimuth_fn="fault_dip_azimuth_GSB_balance.H"
fault_dip_magnitude_fn="fault_dip_magnitude_GSB_balance.H"
output_fn="aaspi_fault_enhancement_GSB_balance.out"
error_fn="aaspi_fault_enhancement_GSB_balance.err"
dip1=10
dip2=25
fault_opacity=0.7
fault_dip_magnitude_znull=100
fault_dip_azimuth_znull=200
window_length=75.0622
window_width=75.0622
window_height=0.0375311
```

The definition of the files is fairly obvious and represent the fault probability, fault dip magnitude, and fault dip azimuth. The fault_dip_filter.txt file is an ASCII-format file that can be plotted using excel. In this case it appears as follows:



	A	B	C
1	"Filter	applied	to fault as a function of fault dip magnitude"
2		theta	filter(theta)
3		0	0
4		1	0
5		2	0
6		3	0
7		4	0
8		5	0
9		6	0
10		7	0
11		8	0
12		9	0
13		10	0
14		11	0.011
15		12	0.043
16		13	0.095
17		14	0.165
18		15	0.25
19		16	0.345
20		17	0.448
21		18	0.552
22		19	0.655
23		20	0.75
24		21	0.835
25		22	0.905
26		23	0.957
27		24	0.989
28		25	1
29		26	1
30		27	1

Which indicates that the fault probability of features with dip magnitude less than 10^0 will be set to zero and those with dip magnitude greater than 20^0 will be unchanged.

Theory

Eigenvector estimation of fault dip and azimuth

Our work is based on Barnes' (2006) contribution to edge detection methods, where he constructed a covariance-like matrix using an edge attribute, α_m, with an m-voxel analysis

$$C_{ij} = \frac{\sum_{m=1}^{M} x_{im} x_{jm} \alpha_m}{\sum_{m=1}^{M} \alpha_m}, \qquad (1)$$

The variables x_{im} *and* x_{jm} are the distances from the center of the analysis window along axis *i* and *j* of the m^{th} data point respectively. In general, the covariance matrix should be computed from a variable, α_m that has a mean of zero. If we assume that the great majority of our data is highly coherent, such that c_m=1.0, we define α_m =1-c_m, (where c_m is coherence). In a three-dimensional setting the covariance-like matrix **C** has three eigenvalues, λ_j, and eigenvectors, $\mathbf{v_j}$. By construction:

$$\lambda_1 \ge \lambda_2 \ge \lambda_3 \ . \qquad (2)$$

The values of λ_3 and $\mathbf{v_3}$ are key to our subsequent analysis. If $\lambda_1 \approx \lambda_2 >> \lambda_3$, the edge attribute defines a plane that is normal to the third eigenvector, $\mathbf{v_3}$. . If $\lambda_1 \approx \lambda_2 \approx \lambda_3$then the coherence data exhibit no orientation, and instead represents either chaotic (λ_3 large) or homogeneous (λ_3 small) seismic facies. In such cases, the orientation of the geological feature becomes randomized and cannot be used for further interpretation.

The size of the analysis window M used in equation 1 is also important since it will affect the resulting eigenvalues and filtering applications. The eigenvectors v1 and v2 define a plane that least-squares fits the cloud of edge attributes, αm . To avoid biasing this estimate along any given axis, we convert our time axis to depth using an approximate conversion velocity, and then define an oblique rectangle whose length and width are equal to its height. The top and bottom faces of the analysis window are parallel to reflector dip (Figure 4).

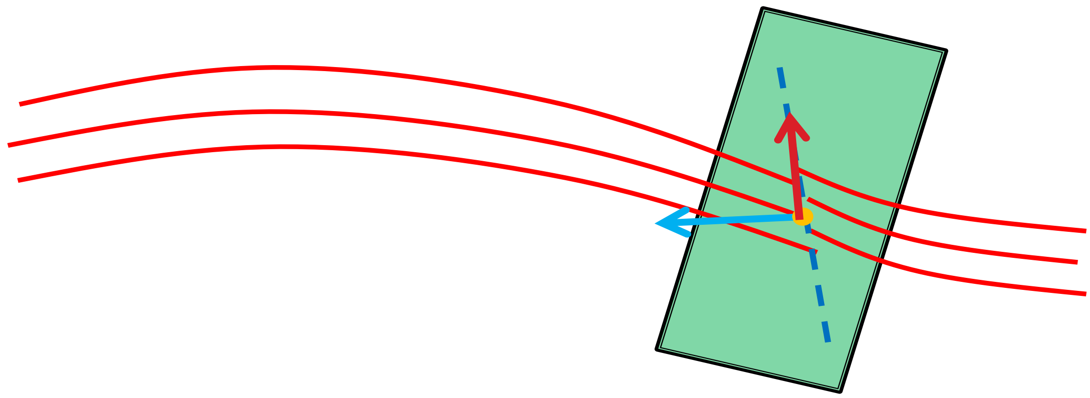


Figure 4. Cartoon showing reflectors (as solid red lines) and a coherence anomaly (as blue dotted line). The analysis window (in green) is an oblique rectangular prism whose top and bottom are parallel to the reflector at the analysis point (in orange). The eigenvector $\mathbf{v}_3$ is perpendicular to the fault plane reflector (blue arrow), while the normal vector **n** is perpendicular to the reflector (red arrow). The Gaussian smoother (in gray) is short in the direction parallel to $\mathbf{v}_3$ and long in the direction parallel to $\mathbf{v}_1$ and $\mathbf{v}_2$.

In order to display the orientation of a planar feature, we define the "fault" dip magnitude, θ, to be

$$\theta = \mathrm{ACOS}(v_{33}) \,, \tag{3}$$

and the "fault" dip azimuth, ψ, to be

$$\theta = \mathrm{A}TAN2(v_{32}, v_{31}) \,, \tag{4}$$

with the three components of eigenvector $\mathbf{v_3}$ are defined as

$$\mathbf{v_3} \equiv \hat{\mathbf{x}}_1 v_{31} + \hat{\mathbf{x}}_2 v_{32} + \hat{\mathbf{x}}_3 v_{33} \tag{5}$$

where the x_1-axis is oriented positive to the North, the x_2-axis positive to the East, and the x_3-axis positive downward. Here we use the word "fault" in quotes; while we are interested in mapping and enhancing faults, this method works similarly for mapping any discontinuity, such as angular unconformities. The word "fault" will help us differentiate these dips from those of the reflectors dip magnitude and azimuth which we will discuss later.

Fault smoothing and edge enhancement using the Laplacian of a Gaussian operator

Laplacian operators are commonly used in sharpening photographic images (Millan and Valencia, 2005). Unfortunately, such sharpening can exacerbate short wavelength noise. In contrast, Gaussian operators are used to smooth such images. The "Laplacian of a Gaussian" or LoG operator avoids some of the artifacts of the Laplacian operator itself by smoothing high frequency artifacts prior to sharpening. Using the associative law when creating the operator, one finds that

$$\mathbf{L}(\mathbf{G}\boldsymbol{\alpha}) = (\mathbf{LG})\,\boldsymbol{\alpha} \;. \tag{6}$$

The composite operator LoG will have the general form:

$$\mathbf{LG}\,\boldsymbol{\alpha} = -\sum_{m=1}^{M} \frac{1}{\pi\sigma^4}\left(1 - \frac{x_{1m}^2 + x_{2m}^2 + x_{3m}^2}{3\sigma^2}\right)\exp\left(-\frac{x_{1m}^2 + x_{2m}^2 + x_{3m}^2}{2\sigma^2}\right)\alpha_m \,, \tag{7}$$

where σ^2 defines the variance of the Gaussian smoother.

Such a mathematical implementation has two advantages. First, one can precompute the LoG operator, rather than cascade two separate operations, resulting in a more efficient algorithm. Second, one is no longer restricted to orienting the Laplacian operator along the seismic acquisition axes, allowing one to implement a directional filter.

Directional smoothing and sharpening

We will modify the LoG operator to be directional: smoothing along the direction perpendicular to the planar discontinuity defined by the eigenvectors $\mathbf{v_1}$ and $\mathbf{v_2}$. We define the Gaussian to be elongated along the planar axes:

$$G_{mn} = exp[-x'^T_m \Sigma^{-1} x'_n] , \tag{8}$$

where $\boldsymbol{\Sigma}$ is defined as:

$$\boldsymbol{\Sigma} = \begin{pmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{pmatrix} = \begin{pmatrix} \lambda_x & 0 & 0 \\ 0 & \lambda_y & 0 \\ 0 & 0 & \lambda_z \end{pmatrix}, \tag{9}$$

and where $\mathbf{x}$’ indicates the coordinates of the voxels in the analysis window within the rotated coordinate system, aligned with the hypothesized fault. In our original (unprimed system) the Gaussian then becomes:

$$G_{mn} = exp[-x^T_m R^T \Sigma^{-1} R x_n] , \tag{10}$$

where $\mathbf{R}$ is the rotation matrix that aligns the new $\mathbf{x'}$-axis with $\mathbf{v_3}$ given by:

$$\mathbf{R} = \begin{pmatrix} v_{11} & v_{21} & v_{31} \\ v_{12} & v_{22} & v_{32} \\ v_{13} & v_{23} & v_{33} \end{pmatrix}. \tag{11}$$

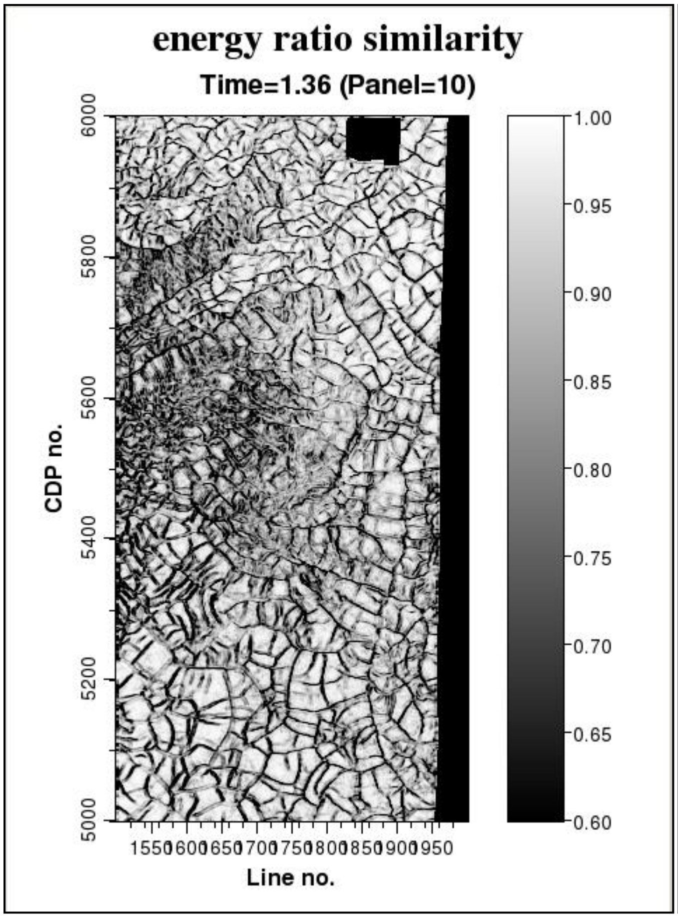
The second derivative of the Gaussian in the $\mathbf{x_3}$‘direction can be written as:

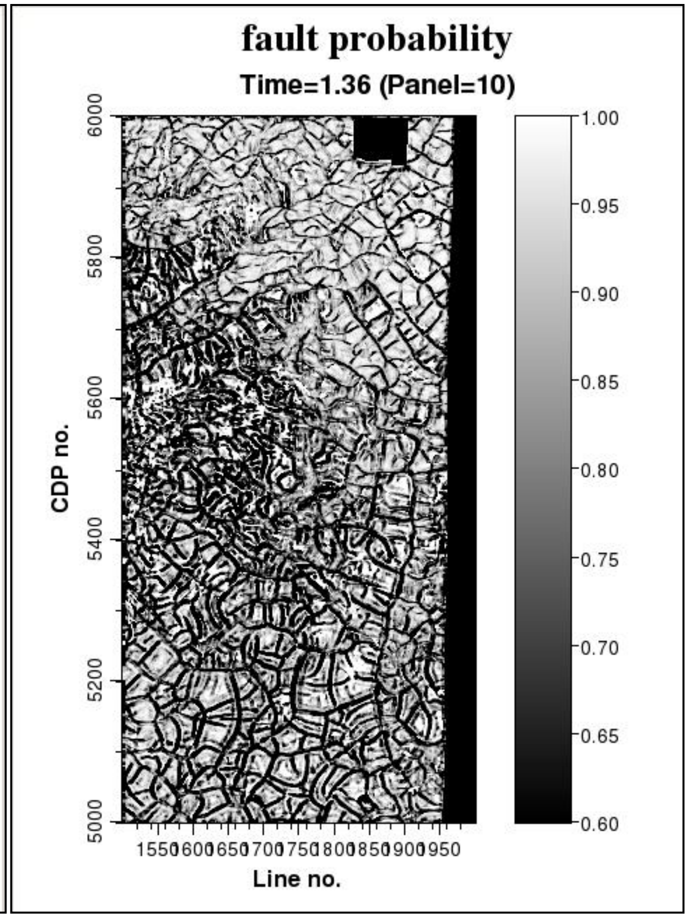
$$\frac{d^2G}{dx'^2_3} = \gamma \left[\frac{-2}{\lambda_3} + 4x'^2_3\right] \exp\left[-\left(\frac{x'^2_1}{\lambda_1} + \frac{x'^2_2}{\lambda_2} + \frac{x'^2_3}{\lambda_3}\right)\right] , \tag{12}$$

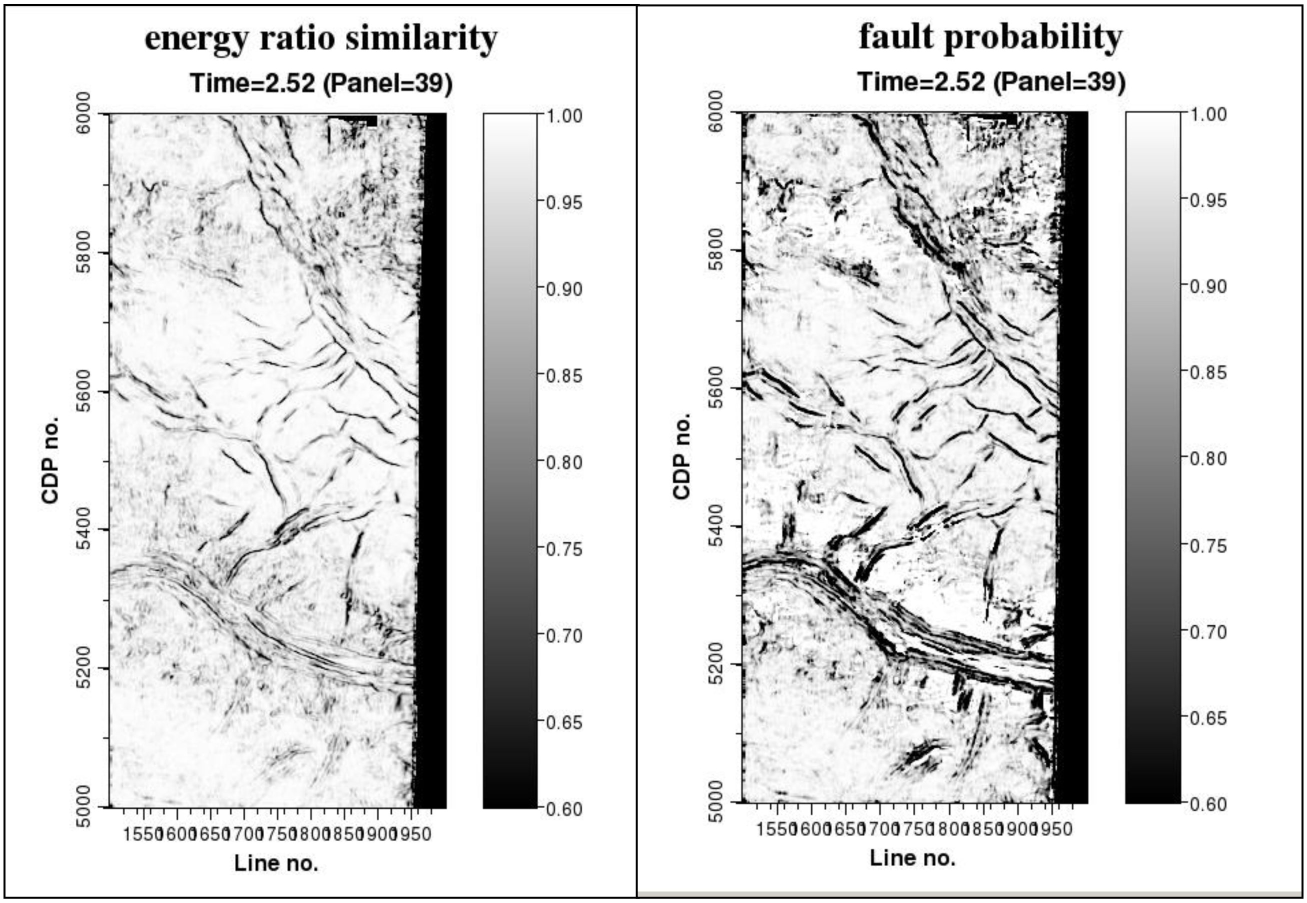
where γ represents a normalization term.

Example: Great South Basin (New Zealand dataset)

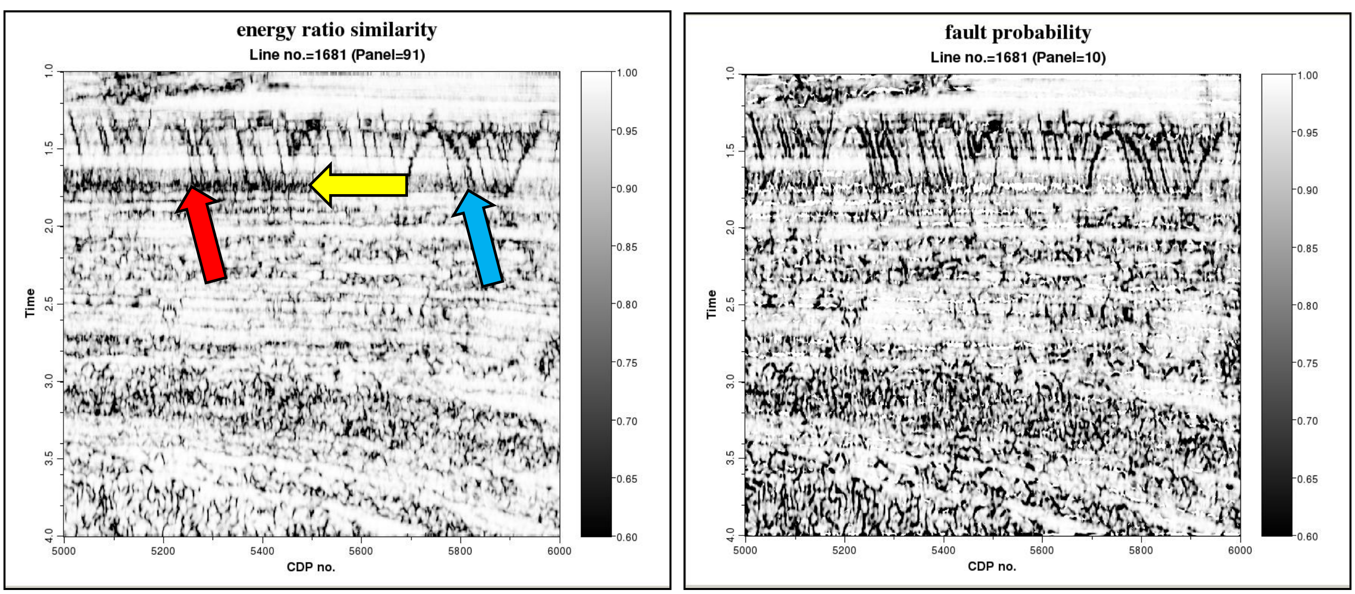
The following data from the Great South Basin are publically available and are provided courtesy of the New Zealand Petroleum and Minerals (NZP&M). The input to fault_enhancement is an energy ratio similarity volume. At present, the amount of sharpening is not significant, suggesting that we may wish to apply more aggressive filters or follow this process by skeletonization.


energy ratio similarity
Time=1.36 (Panel=10)
CDP no.
6000
5800
5600
5400
5200
5000
1550
1600
1650
1700
1750
1800
1850
1900
1950
Line no.
1.00
0.95
0.90
0.85
0.80
0.75
0.70
0.65
0.60


fault probability
Time=1.36 (Panel=10)
CDP no.
6000
5800
5600
5400
5200
5000
1550
1600
1650
1700
1750
1800
1850
1900
1950
Line no.
1.00
0.95
0.90
0.85
0.80
0.75
0.70
0.65
0.60

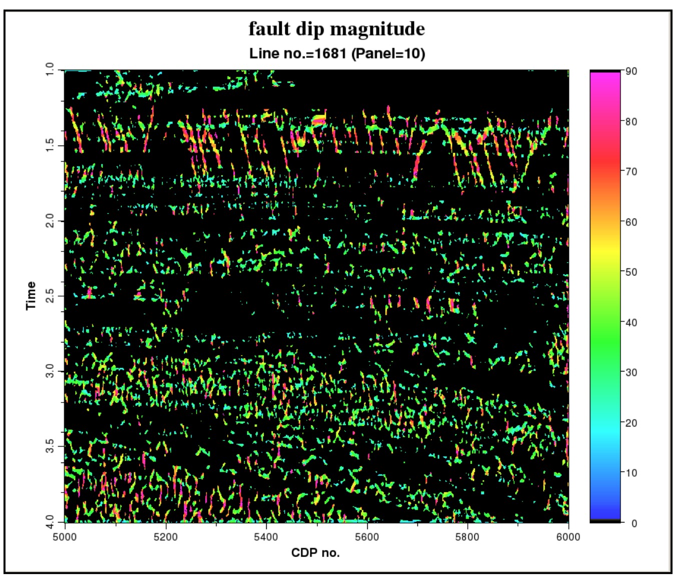


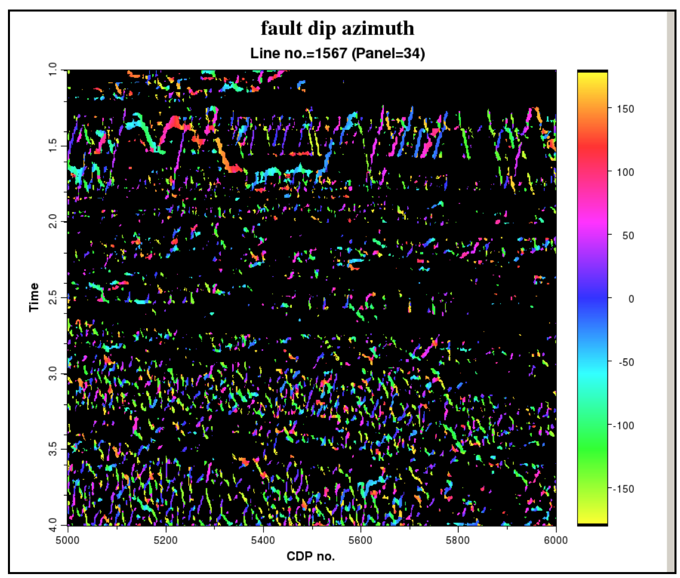
Examination of vertical slices shows some desired improvements:



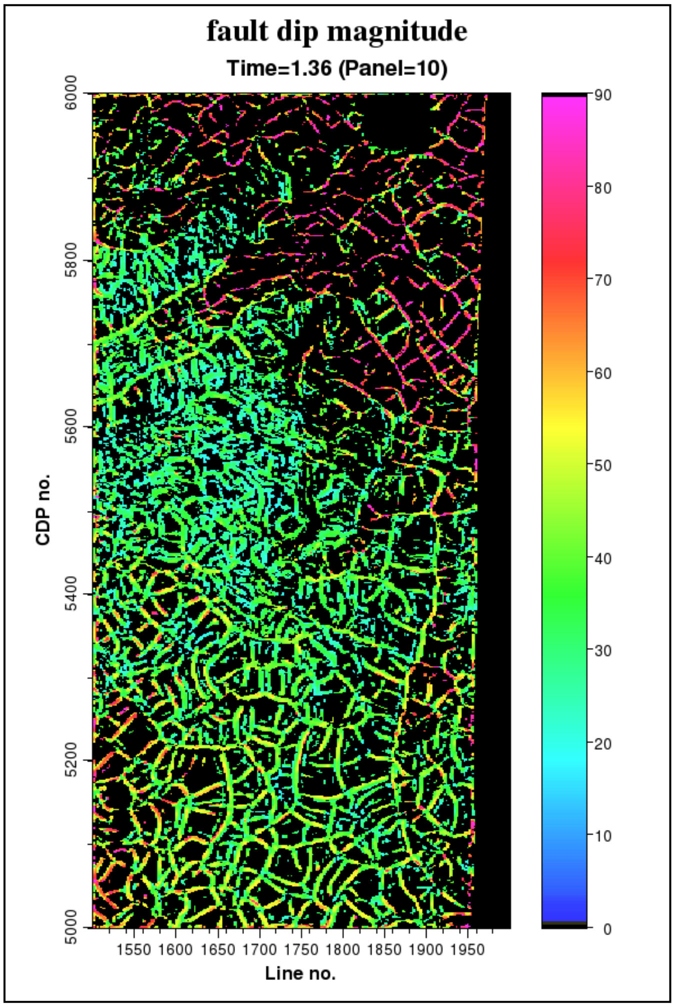
First, the horizontal, low-coherence feature indicated by the yellow arrow, which we hypothesize may be a shale layer, is suppressed through the use of the Dip1 and Dip2 options discussed above. Also note that after smoothing along fault dip magnitude and dip azimuth, several of the faults (red arrow) are more continuous, while others lack the “stair-step” artifacts common in most coherence computations. These improvements are similar to those found by Barnes (2006), although we have not yet attempted skeletonization.

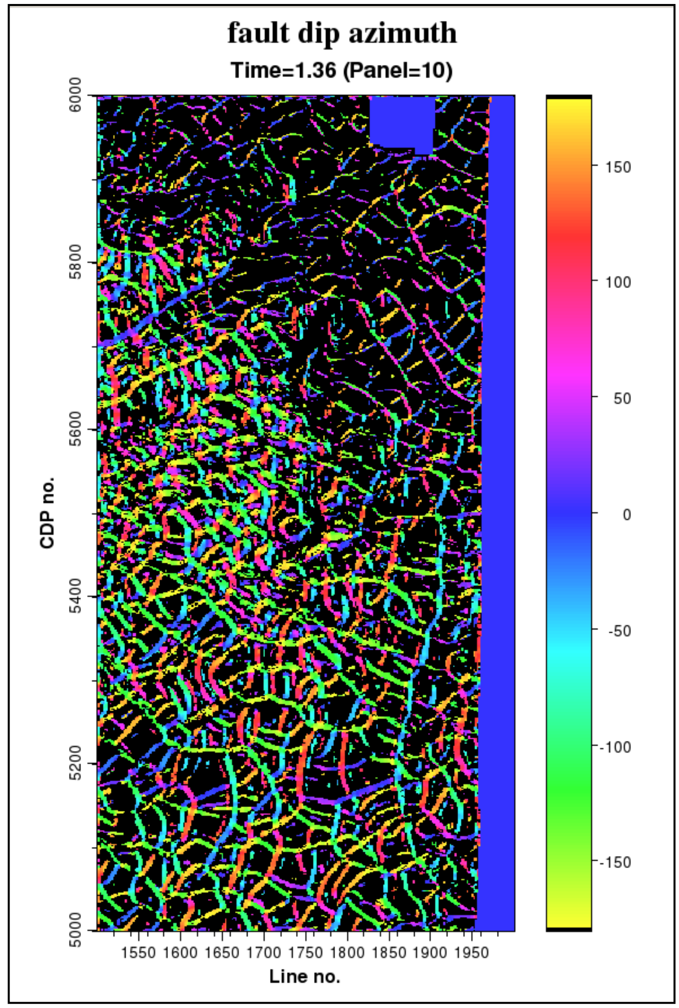
Perhaps more important than sharpening is the computation of fault dip magnitude and fault dip azimuth. Several commercial vendors provide such capability (Dorn et al., 2011), but the details of their implementation in unknown. Our preferred workflow is use an HLS 3D color bar to co-render fault dip azimuth against Hue, fault dip magnitude against Saturation, and fault probability against Lightness, not all of our sponsors have access to such interpretation software. We therefore use the “Fault opacity” option in the GUI above to set fault dip magnitude and fault dip azimuth to a ZNULL value when the fault probability exceeds 0.7 . Then using program aaspi_plot, we set these extreme values to black.





The time slices appear:





The “thickness” of the faults is a function of the opacity chosen. Recall that every voxel has a value of fault probability, fault dip magnitude, and fault dip azimuth. If we set the opacity value to be lower, the faults would be thinner. Obviously, this is more easily done in an interactive environment, where one can use opacity to co-render the three attributes together.

While we prefer co-rendering the fault probability, fault dip magnitude against saturation advantage of fault_enhancement

Figure 5 the result file from running the fault enhancement attribute. Figure 5, 6, and 7 are the output files, with the dip azimuth, dip magnitude and fault enhancement. The latter one shows a better image of the faults in the seismic data, while the first two must first be correndered together to make an appropriate interpretation. Figures 8 and 9 show such correndering. The taper used for each one of the images is displayed, in the same way that was discussed in the "how to run" part of this document.

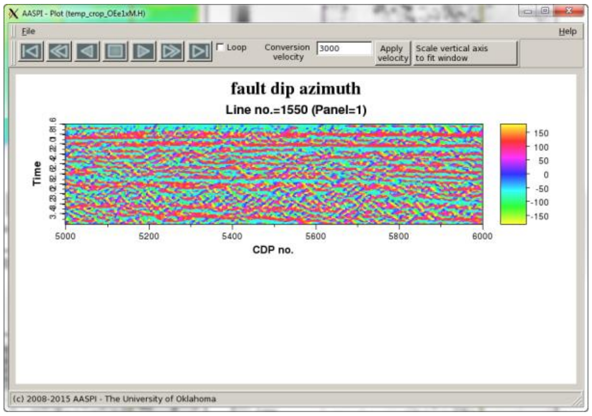


Figure 5. **"**Fault_dip_azimuth_..." file generated.

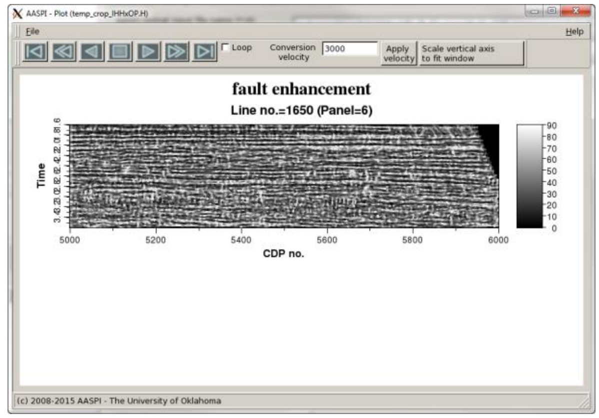


Figure 6. Fault_dip_magnitude_..." file, generated.

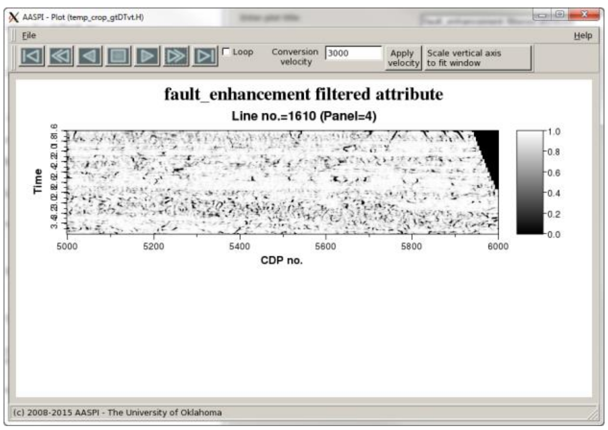


Figure 7. **"**Fault_enhancement_..." file with the improved fault images.

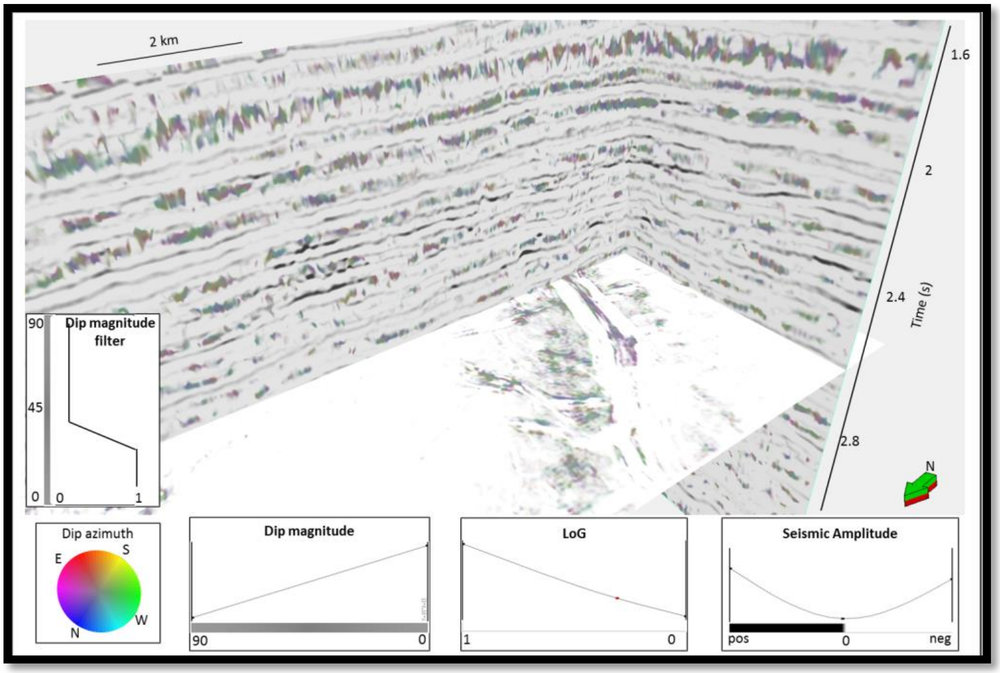


Figure 8. Dip magnitude, azimuth and fault enhancement attribute corendered together, with a taper on the dip angles displayed. Only relatively vertical discontinuous events are displayed with the tapering function.

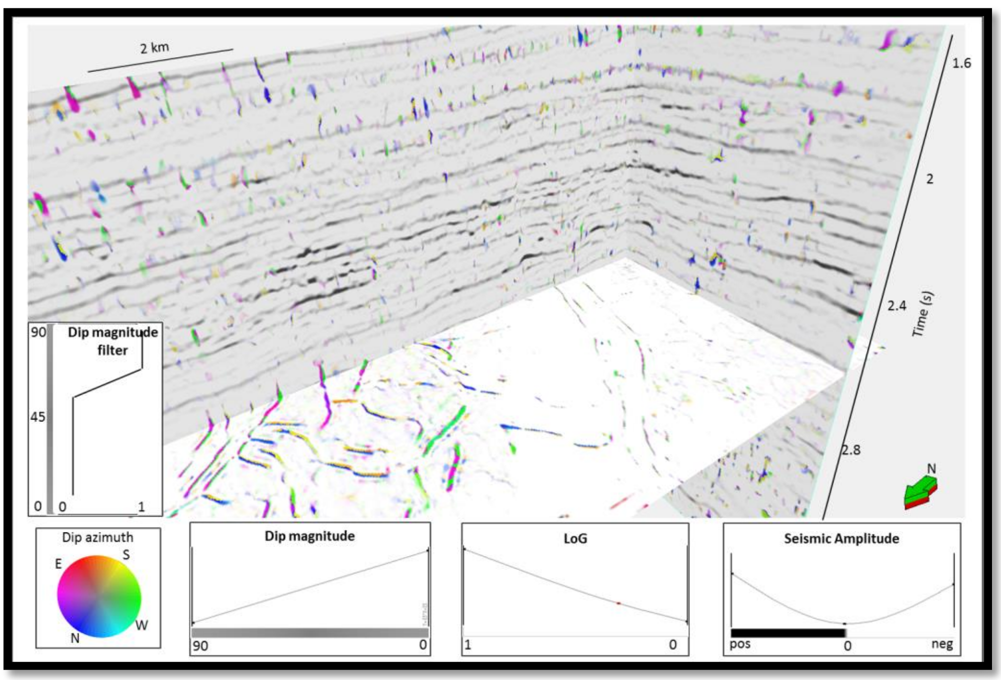


Figure 9. Dip magnitude, azimuth and fault enhancement attribute corendered together, with a taper on the dip angles displayed. Only

relatively gorizontal discontinuous events are displayed with the tapering function.

References

Barnes, A. E., 2006, A filter to improve seismic discontinuity data for fault interpretation,: Geophysics, **71**, P1-P4.

Machado, G., A. Alali, B. Hutchinson, O. Olorunsola, and K. J. Marfurt, 2016, Display and enhancement of volumetric fault images, Interpretation, **4**, xx-yy.

Dorn, G. and B. Kadlec, 2011, Automatic Fault Extraction in Hard and Soft - Rock Environments: The 31stAnnual Bob F. Perkins Research Conference31stAnnual Bob F. Perkins Research Conference on Seismic attributes –New views on seismic imaging: Their use in exploration and production, 587-619.