3D Seismic Attributes to Define Structure and Stratigraphy – A Hands-On Short Course
Part 2: Geometric attributes and multiattribute display

AASPI Lab Exercises by Kurt J. Marfurt and Thang Ha

The University of Oklahoma
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table of Contents</td>
<td>1</td>
</tr>
<tr>
<td>Introduction</td>
<td>2</td>
</tr>
<tr>
<td>Computing volumetric estimates of dip - Program dip3d</td>
<td>4</td>
</tr>
<tr>
<td>Documentation and the Help tab</td>
<td>6</td>
</tr>
<tr>
<td>Computing coherence and energy gradients – Program similarity3d</td>
<td>8</td>
</tr>
<tr>
<td>Multiattribute Display</td>
<td>10</td>
</tr>
<tr>
<td>Displaying inline and cross-line dip in the AASPI QC Plotting tab</td>
<td>10</td>
</tr>
<tr>
<td>Co-rendering dip azimuth and dip magnitude using program hplot</td>
<td>14</td>
</tr>
<tr>
<td>Co-rendering dip azimuth and dip magnitude using program corender</td>
<td>16</td>
</tr>
<tr>
<td>Co-rendering dip-azimuth, dip-magnitude, and seismic amplitude volumes</td>
<td>18</td>
</tr>
<tr>
<td>Calculating apparent dip at any azimuth</td>
<td>20</td>
</tr>
<tr>
<td>Calculating similarity (coherence), energy, and coherent energy gradient attributes using program similarity3d</td>
<td>22</td>
</tr>
<tr>
<td>Energy and energy gradients</td>
<td>22</td>
</tr>
<tr>
<td>Coherence (Similarity)</td>
<td>26</td>
</tr>
<tr>
<td>Computing structural curvature, reflector shapes, reflector rotation, reflector convergence, and aberrancy using program curvature3d</td>
<td>33</td>
</tr>
<tr>
<td>Curvature</td>
<td>35</td>
</tr>
<tr>
<td>Aberrancy</td>
<td>41</td>
</tr>
<tr>
<td>Reflector Convergence</td>
<td>44</td>
</tr>
<tr>
<td>Shape Index and Curvedness</td>
<td>47</td>
</tr>
<tr>
<td>Computing amplitude curvature with program curvature3d</td>
<td>50</td>
</tr>
<tr>
<td>Computing gray level co-occurrence matrix textures using program glcm3d</td>
<td>53</td>
</tr>
</tbody>
</table>

---

# Introduction

Attribute-Assisted Seismic Processing and Interpretation – February 18, 2020
Attributes computed from migrated, stacked seismic volumes can be broken into several broad categories: single trace attributes, geometric attributes, spectral attributes, and attributes that measure the seismic behavior about a given horizon or within a formation. The AASPI software also provides other useful applications in data conditioning, image processing, correlation, and multiattribute display. In addition to multiattribute display, multiple attributes can be analyzed using a rich collection of machine learning dimension reduction and classification tools.

Part 2 of this short course examines a suite of geometric attributes that measure the orientation, continuity, and texture of 3D seismic data that aid in mapping faults, folds, and flexures, as stratigraphic features such as channels, mass transport complexes, and carbonate buildups as well as drilling hazards. We begin with computing volumetric dip and azimuth. All other geometric attributes and most of the image processing algorithms are computed along structural dip, including coherence, amplitude gradients, curvature, aberrancy, and gray level co-occurrence matrix textures. Co-rendering each of these attributes with the original amplitude volume and each other is critical to understanding what geologic features (or noise!) they are measuring. For this reason, we will also cover the several simple multiattribute display options provided in the AASPI software. We will postpone the discussion of multispectral dip and multispectral coherence until we have defined spectral decomposition in Part 3.

The geometric attributes are found under the Geometric Attributes tab of the aaspi_util GUI:

In this exercise we will investigate programs dip3d, similarity3d, curvature3d, and similarity3d. We will discuss filter_dip_components and sof3d under Part 5 on data conditioning. Programs disorder and nonparallelism are useful in machine learning, but provide only limited interactive interpretation value. Program similarity_multiple_input provides a flexible interface to compute coherence from an arbitrary mixture of user-defined input volumes, and is typically used to analyze prestack data volumes.
Computing volumetric estimates of dip - Program dip3d

In general, the first step in computing geometric attributes and structure-oriented filtering is to compute estimates of volumetric dip using AASPI program dip3d. These dip components will be input to filter_dip_components, similarity3d, curvature3d, glcm3d, sof3d, and several other programs. To run dip3d, go to Volumetric Attributes → dip3d (7A). Browse to the AASPI-format dataset that you converted from SEGY (7B). Usually the unique project name is automatically read in after you browse a file. Also, please fill in the suffix (usually a number to distinguish this run from subsequent runs using different parameters). The unique project name and suffix are used in constructing the output file names.

In the Primary parameters tab, make sure to place a check mark in front of the Dip Magnitude and Dip Azimuth results (7C). In the Parallelization parameters tab, you can specify the number of processors to run dip3d (7D). If you already set up this parameter in the previous section, then you don’t have to modify anything. Finally, click Execute dip3d (7E). If you see a Windows Firewall message, hit allow access (7F). This is important if you want to run AASPI MPI-related programs across multiple processors. The MPI process manager needs to access the internet to be able to communicate with other processors on the same desktop. In principle, MPI can run across more than one Windows machine, as it is does on Linux; in practice, the security on Windows machine is insufficiently strong such that almost IT installations prohibit such communication.
Attribute-Assisted Seismic Processing and Interpretation – February 18, 2020
In your black window (on Linux, mine is gray), you will see the following kind of information roll by, showing the progress of the computation.

Upon completion, the following pop-up window will appear:

All is as it should be.

**Documentation and the Help tab**

Because the goal of our students is to publish and get their name out there to find a job, the documentation in the AASPI software is much more extensive than for most commercial interpretation software. The documentation can be found on our web site [http://mcee.ou.edu/aaspi/documentation.html](http://mcee.ou.edu/aaspi/documentation.html) or by invoking the Help tab on each application.
In addition to directions similar to this tutorial exercise providing a description of each parameter followed by a suite of examples, there is extensive documentation of the theory of each algorithm in a suite of gray boxes like this one:

Each documentation also contains a suite of references providing additional detail:
Computing coherence and energy gradients – Program similarity3d

To better use our class time, we will initiate the next program, similarity3d, before plotting our results from dip3d. (In general, it is always best to QC your result before proceeding to the next task). After dip3d finishes, the next step is to calculate coherence-related attributes. Go to Volumetric Attributes → similarity3d (8A). Browse to the seismic amplitude, inline dip, and cross-line dip datasets (8B). Inline dip and cross-line dip were computed by program dip3d. Under primary parameters tab, place a check mark next to the Total energy and Coherent energy attribute names (8C). The default values of (8D) window height and (8E) length and width are reasonable. Again, under parallelization tab, make sure the processors per node is the number you set up in the previous sections. Click Execute (8E).
If it appears the similarity3d job will take some time, let it run and jump to the next section – Displaying dip azimuth and dip magnitude using program hsplit and return to this step later.
Multiattribute Display

Displaying inline and cross-line dip in the AASPI QC Plotting tab

Revisit the “Display data using the AASPI QC Plotting Tab” section to display the results of dip3d: inline dip and cross-line dip attributes. Simply browse to the desired attribute (20B) to select it. The default color bar (20B) and title (20C) are reasonable but can be changed. The vertical images aren’t very impressive (though quite accurate) in vertical slices. Instead, set the 3rd axis to be time (20E). Then click Execute.
AASPI Hands-on short course – Part 2: Geometric attributes and multiattribute display
If you selected the GST option of computing the reflector dip vector components, you will also have generated a chaos volume. Let’s plot this one up as well:
The values of chaos range between -1.0 and +1.0, so your image will appear to be faded out. To address this issue, right click on File -> Settings and then select the Data Ranges tab. Select Histogram Range and let the data range from 0.5 to 99.5 percentile. You will note in the display that all the values are negative, with a value of -1.0 representing “coherent” or “not chaotic” parts of the data.
Co-rendering dip azimuth and dip magnitude using program hsplit

One way to interpret two attributes together is to co-render them using a 2D hue-lightness color bar. Go to Display Tools → hsplit (9A). Browse and select dip azimuth as the attribute to plot against hue axis (9B). Browse and select dip magnitude as the attribute to plot against the lightness axis (9C). Scaling is a key component of seismic data and seismic attribute display. Click Execute (9D). The 2D color bar, wheel, and histograms (9E-9H) will be displayed along with the co-rendered time-slice of the two attributes.

If you wish to view the co-rendered result in inline mode or cross-line mode, you can always follow the previous section “Display data using AASPI QC Plotting Tab” and browse to the file.
with the name in box 9l (in this case, magnitude_vs_azimuth_GSB.H), and set the 3rd axis to be line or cdp no, accordingly.

A time slice and inline view of the co-rendered result using hspolot are shown in (10A) and (10B).
Co-rendering dip azimuth and dip magnitude using program corender

Another way to co-render two attributes is to go under Display Tools and invoke (11A) program corender. This application constructs co-rendered volumes by adding layers using opacity. The (11B) first, or base layer, is always opaque. For most applications (such as this one) it will use an attribute (11C) a polychromatic color bar, in this case, a cyclic color bar cyclic.alut. For dip azimuth, we will want to fix the scaling to fall between -180° and +180°.

To emulate saturation, we will plot the (11E) second layer using (11F) dip magnitude, plotted against a monochrome gray color bar. We will set the opacity to set low values of dip magnitude to be opaque, rendering a gray image. High values will be more transparent, allowing us to “see through” this 2nd layer into the polychromatic layer below it. In order to optimally span the range of the data, we use (11H) statistical scaling with a 90-percentile clip. The values of dip magnitude are always non-negative.
In a February, 2015 Interpretation paper, Marfurt shows how to effectively blend multiple attributes in Petrel (http://mcee.ou.edu/aaspi/publications/2015/Marfurt_2015_Techniques_and_best_practices_in_multiattribute_display.pdf). If your software allows opacity, these same “tricks” can be used there. If you don’t have access to such 3D visualization software, then you can load the 2D color bar from program hlplot into your workstation. Instructions on how to do this are in the AASPI documentation. You will almost certainly be limited to use no more than 256 (230 for Kingdom) colors and will need to turn the default voxel color interpolation option off.

Choosing the 3rd axis to be time provides the following time slice:

![Co-rendered GST dip azimuth GST dip magnitude](image)

Since we are using 256 colors for the base layer attribute and another 256 colors for the 2nd layer attribute, this image exhibits 256*256=65,536 colors, and thus provides somewhat better resolution in the upper right corner than the 256-color image generated using program hsplit. Images with these many colors cannot be loaded into most commercial workstation packages, although several can generate similar images.

**Co-rendering dip-azimuth, dip-magnitude, and seismic amplitude volumes**

To quality control our estimates of dip, we wish to compare it to the 3D seismic amplitude volume from which it was calculated.
Let’s add (12A) the seismic amplitude volume as (12B) a third layer in the corender GUI:

Here the color bar and transparency are not intuitive. Read the paper by Marfurt (2015) on color display for further details. For now, use a binary black and white color bar, and (12C) set this 3rd layer to be transparent (i.e. see through) for small positive and negative values of amplitude and opaque (black for positive, white for negative) values of amplitude. We’ll use (12D) statistical scaling, and clip the data at 90-percentile, i.e. letting it range between 5 and 95 percentile of the data range. We obtain the following image:
If we wish to look at a crossline, we need to rerun the `corender` program with a different order of axes:


To obtain the following image:

Note the progradation dipping to the north, displayed as yellow.

**Calculating apparent dip at any azimuth**

In some case, you may know that structural features of interest trend along a given azimuth. If this is the case, the inline and crossline dip components may not give you the optimum image. To look at a specific direction (i.e. apparent dip go to Volumetric Attributes → *apparent_cmpt* (13A). Browse to the inline dip attribute (13B). The cross-line dip would be automatically loaded after you browse the inline dip. Then click Execute (13C). After the program finishes, refer to the previous section of “Display data using AASPI QC Plotting Tab” to plot different apparent dip components (13D-13G).
In commercial interpretation workstation software, you would do this using an “attribute calculator” tool. You would then write an equation that looks like

\[ \text{apparent}_\phi = \cos(\phi) \times \text{inline}_\phi + \sin(\phi) \times \text{crossline}_\phi \]

Now, if the inline azimuth and crossline azimuth are not North and East, you need to fool with this equation. If they are oriented counterclockwise you will need to fool a little more. Program \text{apparent}_\phi \text{cmpt} does this fooling for you.
Calculating similarity (coherence), energy, and coherent energy gradient attributes using program similarity3d

Coherence was first developed at Amoco in the mid-1990s. In the AASPI software, we have several versions using generalization of semblance, eigenstructure analysis and the Sobel filter (the one you have in your camera software). Our implementation is done with complex (or analytic) traces, which avoids artifacts that might occur near zero crossings of an otherwise high amplitude wavelet. All calculations are done along the previously calculated vector dip.

Go to Volumetric Attributes → similarity3d and then load in your original seismic amplitude volume as well as the inline dip, and crossline dip volumes previously computed in program dip3d. The defaults are reasonable, with an implicit assumption that the dominant period of your data is approximately ±5 samples. If you have high quality data, you can make the vertical window ±1 samples. Making the vertical window much larger than your dominant period may “stack” vertical faults (which you might like), but smear dipping faults, and mix channels or other stratigraphic features on neighboring horizons (which you won’t like). You can also choose smaller or larger analysis windows, with smaller window running faster and providing better lateral resolution. Be sure to click the optional Want Total Energy Attribute? and Want Coherent Energy Attribute? options towards the bottom.

Energy and energy gradients

Once the program has finished running (in my case it took a while) plot time slices of the total energy and coherent energy. The total energy is the energy of complex trace within in your analysis window (in the case above, an elliptical, 5-trace 12.5 by 25 m radii by ±20 ms vertically oriented oblique elliptical cylinder window whose top and bottom are parallel to structural dip. The coherent energy is computed from a Karhunen-Loeve (KL) filtered version of the data. The KL-filter is another word for principal component filtering, which we will discuss when we address structure-oriented filtering.
At present, these images look nearly identical. We will return to their differences shortly. But first, note that we can also take derivatives in the inline and crossline direction (but always along structure) of the coherent component of the data. We call these the inline and crossline coherent energy gradients. Please find these files and plot them to obtain the following images:
inline component of coherent energy gradient
Time=1.72 (Panel=14)

crossline component of coherent energy gradient
Time=1.72 (Panel=14)
Let’s co-render the base layer of coherent energy plotted against a fire color bar with the second layer of crossline coherent energy gradient plotted against a binary black-white color bar, making the values close to zero transparent, as shown in the two images of the GUI below:

The result is the following image:
This image looks a little like a shaded relief map, except that it is of energy, not of structure! We see some channels incising the shelf edge.

**Coherence (Similarity)**

Returning to the two energy images above, we combine them by taking the ratio of *coherent energy/total energy*. While these two images look very similar, the result is the *energy ratio similarity*, which we plot below:
Blending this image with coherent energy where the second (similarity) image is plotted against a monochrome black colorbar as shown in the GUI resulting in the following image
We note that some of these channels are high energy (sand filled?) while others are low energy (shale filled?).

There are two other similarity volumes which computationally are different but yield similar images – the outer product similarity, which is a generalization of semblance, and the Sobel filter similarity, which is a generalization of the filter you find in photoshop. Their images look like this:
Which one is best? It depends on the data quality and it depends on the geology. We can identify some end members. If the waveform changes but the amplitude stays the same, energy ratio
coherence is the best (this often occurs in faulted areas). If the waveform stays the same, but the amplitude changes (such as below thin bed tuning), Sobel filter similarity is the best. In this example, they all give excellent images of faulting, channel incisement, and some FLTs to the top center of the image. Now look at the white part of the similarity images where there is little detail and compare these areas to the inline and crossline energy gradients. In this case, the inline and crossline gradients show the most detail where the coherence shows the least! These two attributes are thus quite complimentary.

Co-rendering dip azimuth, dip magnitude, and coherence using program corender

If your corender GUI is still open, let’s add a third attribute, energy ratio coherence, to the previously plotted dip azimuth and dip magnitude. Let’s plot it against a monochrome black color bar, where voxels exhibiting high coherence are set to be transparent, while those exhibiting low coherence are set to be opaque, and thus black.
The resulting image looks like this:
where faults with significant displacement appear to be black, while those that are relatively continuous exhibit a colored dip. A little deeper we can identify syneresis features:
A vertical slice through these three data volumes looks like this:

**Computing structural curvature, reflector shapes, reflector rotation, reflector convergence, and aberrancy using program curvature3d**

Along with coherence, curvature is one of the most popular geometric attributes. Depending on the geology, curvature may show conjugate faulting about larger faults, the axial planes of folds,
karst collapse, carbonate builds ups, and differential compaction of sediments over incised channels. Mathematically, curvature asks the question: “in which directions is a local surface most and least deformed?”. For a quadratic surface, these directions will be orthogonal in directions along local structural dip and will be “eigenvectors” of the quadratic surface. The direction of most deformation will be the direction of maximum curvature, while that of least deformation will be the direction of minimum curvature. If we have syncline, the direction of maximum curvature will be perpendicular to the syncline and the (eigen) value of maximum curvature will be negative. Yes! The direction of minimum curvature will be parallel to the syncline and the (eigen) value of minimum curvature will be zero. Many folks are uncomfortable with a “maximum” anything be less than a “minimum” anything, which has led to the generation of alternative definitions and a great deal of confusion.

In the AASPI software, we avoid that confusing by working almost exclusively with the most positive curvature, $k_1$, and the most negative curvature, $k_2$, where $k_2 \leq k_1$. Furthermore, we use the eigenvectors to define the strikes, $\psi_1$ and $\psi_2$ of these features. With this preamble, let’s invoke the `curvature3d` GUI by going to Volumetric Attributes → `curvature3d` and selecting it. Leave the structural curvature (17A) radio button as is. We’ll return here when we compute amplitude curvature. Structural curvature is computed from inline and crossline derivatives of inline and crossline structural dip (including the cross derivatives!). So go enter them (17B and C). Leave the “long wavelength” curvature radio button (17D) as is for now. If you click it, you will get the short wavelength curvature. Then select various attributes as indicated (17E, 17F, and 17G). Click `Execute curvature3d` and go to lunch or take a long coffee break!
Curvature

All of the parameters are well described in the AASPI documentation and publications. After the program has completed, let’s plot the values of the most positive and negative structural curvature, $k_1$ and $k_2$ (not the files that begin with k1_strike and k2_strike which we will address later) to obtain the following images.
As you might expect, most positive curvature exhibits mostly positive (in red) values while most negative curvature exhibits most negative values (in blue). Planar (but perhaps dipping!) features appear as white. Blue features in most positive curvature indicate a negative value for $k_1$. Since $k_2$ is always less than or equal to $k_1$, this means both principal curvatures are negative, defining a bowl shape. In contrast, red features in most negative curvature indicate a positive value for $k_2$. 
Since $k_1$ is always greater than or equal to $k_2$, both principal curvatures are positive, defining a dome shape.

Next, let’s co-render $k_1$, $k_2$, and coherence on time slices, and $k_1$, $k_2$, and seismic amplitude (plotted against a gray scale) on vertical slices. The paper by Marfurt (http://mcee.ou.edu/aaspi/publications/2015/Marfurt_2015_Techniques_and_best_practices_in_multiattribute_display.pdf) shows you how to do this in Petrel. Here we will use AASPI program corender. The files, colorbars, opacity, and scaling parameters for the three layers look as follows.
and provides the following co-rendered result:
Note that the faults in the SE corner of the survey are well delineated by both curvature and coherence, while smaller faults to the E-NE show up only on curvature. Also note that many of the channels cutting through the shelf edge exhibit a positive curvature anomaly (in red) showing structural highs that suggest differential compaction over sand-filled channels cutting through a shale shelf margin.

Next, let’s plot the strike of k1 as the base layer using a cyclic color bar, the value of k1 as layer 2 plotted against monochrome gray and with the low positive values being opaque, and energy ratio similarity as layer 3 against monochrome black with low values opaque. (You can do the same thing in Petrel with a little practice):
We end up with the following image at time slice t=1.72 s:
We can generate the same kind of image using program hlsplot, which provides a useful colorbar to go with it:

The (eigen) value of $k_1$ plotted against its strike and coherence is shown in the middle image. Note that the color wheel replicates after 180 degrees – EW strike is the same as WE strike. Co-rendering on vertical slices will show the lineaments are on one (foot wall) side of a fault seen in coherence in $k_1$ and on the other (hanging wall) side in $k_2$.

**Aberrancy**

A more recent attribute that’s not yet in any commercial software is aberrancy. Aberrancy is a vector that measures the magnitude and orientation of flexures and is the 3rd derivative of an implicit depth structure map.
Use program corender to plot the azimuth of aberrancy against a cyclic color bar as the base layer, the magnitude of aberrancy against a monochrome gray colorbar as the 2\textsuperscript{nd} layer, setting the low values to be opaque, and coherence against a monochrome black colorbar as the 3\textsuperscript{rd} layers, setting the low values to be opaque. The result is the following figure:

![Co-rendered vector aberrancy vs. coherence](image)

Note that the faults in the ENE exhibit aberrancy anomalies but not coherence anomalies. The displacement is simply too small for coherence to see.

Do better understand what is going on, let’s change the 3\textsuperscript{rd} layer to be the original seismic amplitude volume. Here, I will use statistical scaling (90\%) and a binary black-white color bar with extreme values transparent and values near zero opaque:
Plotting an inline slice, I obtain the following image, where the colors indicate the direction (from North) of downward flexures.

![Image of inline slice]

Faults seen with coherence and aberrancy  
Faults seen only with aberrancy

Note there are many through-going flexures on the right-hand side of this image at the yellow (t=1.72 s) time slice. To better understand what’s going on, let’s co-render seismic amplitude (against a black-gray-white) colorbar and coherence (against a monochrome yellow colorbar with low values opaque):
A chaotic area, perhaps associate with syneresis lies above the red time slice at \( t=1.72 \) s, but there is only one through-going fault seen on coherence. Details on the aberrancy theory can be found under the Help tab and in a recent paper by Qi and Marfurt (2017) http://mcee.ou.edu/aaspi/publications/2018/Qi_and_Marfurt_2018-Volumetric_aberrancy_to_map_subtle_faults_and_flexures.pdf.

**Reflector Convergence**

Now let’s do a similar plot of the azimuth and magnitude of reflector convergence with coherence. The azimuth of convergence is the direction in which layers thin or are truncated. If the magnitude of convergence is zero, the layers are constant thickness. In this image these zones will appear as gray. Zones with strong convergences, such as pinchouts and unconformities will appear as brighter colors. Mathematically the convergence is the \( x \) and \( z \) components of the curl of vector dip, \( \nabla \times \mathbf{p} \).

Let’s plot vector convergence and coherence together using program corender. The three layers look as follows:
The result is the following image:
The 3-component cylindrical colorbar can be generated in program **hlsploit** and looks like this:

The yellow area indicates a convergence to the South. To examine this, let’s plot a crossline through this anomaly at CDP 4100, showing the seismic data where the white line indicates the time slice at t=1.80 s:
Note the yellow anomaly delineates the downlapping sediments coming from the shelf on the right of the image. Slightly above the yellow anomaly, the blue anomaly maps convergence to the North, corresponding to a toplap surface.

**Shape Index and Curvedness**

Shape index and curvedness make other useful structural pairs. The mathematical description of this attribute can be found in the AASPI documentation. In the following display, domes will be red, ridges yellow, saddles green, valleys cyan, and bowls blue. Planar features will be gray.

The time slice at t=1.44 s exhibits several syneresis features (the submarine analogue to the shrinkage seen in desiccated mud cracks). Again, co-render using program `corender` using the following settings:
The 2D color bar for high coherence reflectors is gray for planar events (Curvedness=0) and a fully saturated color for highly deformed events:

Lower values of coherence result in darker colors of the same hue.
Examining a vertical slice along line 3181 shows these features to be bowl-shape, corresponding to a shape index of -1, or a bowl:
These kinds of images are useful in mapping karst collapse, carbonate buildups, and other features. Earlier, we noted a positive curvature anomaly along some of the channels incising the shelf edge at t=1.72 s. Using the shape index and curvature we obtain the following image:

Note that these same channels now appear as ridges (in yellow) again suggesting differential compaction over sand.

**Computing amplitude curvature with program curvature3d**

If you are comfortable with thinking of the principal structural curvatures as directions (eigenvectors) of maximum and minimum time-structure change, then the extension to
amplitude curvature is simple — the direction (again eigenvectors) of maximum and minimum lateral amplitude variation. Whereas the input to structural curvature was the inline and crossline derivatives of time structure (the inline and crossline apparent dip components), the input to amplitude curvature will be the inline and crossline derivatives of amplitude, in our case the inline and crossline energy-weighted amplitude gradients. Click the Curvature Type tab until you see Type 2: AMPLITUDE CURVATURE (e) (17L). The GUI is smart enough to look only energy gradient components (17M and N). The output of amplitude curvature will not be rotationally invariant, since the vertical axis is time (or depth) and the horizontal axes are in units of energy per km or energy per kft. For this reason, we generate a somewhat simpler negative and positive curvature values \( e_{\text{pos}} \) and \( e_{\text{neg}} \) which map the crests and troughs of the lateral variation in amplitude rather than a rotated axes of maximum deformation (discussed in the documentation). Each of these curvatures also have a strike. Click Execute, let it cook, and then plot the results. As you see below, amplitude curvature is quite different from structural curvature. First, the features have much more rapid lateral variation in amplitude curvature than in structural curvature. Second, while they can be coupled to the same geologic features (incised channels on the shelf in these images) they do not seem to illuminate the faults seen on structural curvatures. Amplitude curvature can also be computed from input P- and S- impedance volumes. In these cases, the “impedance” curvature highlights lineaments of anomalously low or high changes in P- and S-impedance.
amplitude most-positive curvature (e_pos)

Time=1.72 (Panel=14)

Computing gray level co-occurrence matrix textures using program glcm3d
In contrast to the previous geometric attributes, the gray level co-occurrence matrix textures provide statistical rather than geometric variations in seismic amplitude volumes. The classic definition of texture is to take your thumb (an analysis window!) and place it over a surface. As you feel the topographic asperities in this surface your brain may interpret the surface to be rough, smooth, slick, ribbed, or waffled. GLCM works on lateral “amplitude” asperities and will be sensitive to diagenetic alteration, chaotic or smooth deformation, period fractures, or other features. By themselves, GLCM textures are less useful than the structural or spectral component attributes. However, modern resource plays often have 100s of wells within a relatively small 3D seismic survey. We have found that including such textures to augment P- and S-impedance, spectral component, or geometric attributes in either neural network or unsupervised classification techniques provides an additional measure that facilitates seismic facies classification. See some recent work by Jie Qi as an example (http://mcee.ou.edu/aaspi/publications/2016/Qi_and_Marfurt_2016_Semisupervised_facies_classification.pdf).

To run glcm3d, go to Volumetric Attributes → glcm3d

and fill out parameters on the following GUI:
In this case, I’ve generated all the GLCM attributes. In general, I find homogeneity and entropy differentiate facies that are otherwise difficult to separate. Program glcm3d is computationally intensive, so don’t plan on using your laptop for a day. You may wish to crop your data and run it on a target area, or run it on a larger machine. Here are the resulting images at time slice t=1.720 s:
AASPI Hands-on short course – Part 2: Geometric attributes and multiattribute display

**avg GLCM energy**

*Time (s)=1.36 (Panel=5)*

**avg GLCM mean**

*Time (s)=1.24 (Panel=2)*
avg GLCM variance
Time (s)=1.72 (Panel=14)

avg GLCM entropy
Time (s)=1.72 (Panel=14)
avg GLCM homogeneity
Time (s)=1.72 (Panel=14)

avg GLCM contrast
Time (s)=1.72 (Panel=14)
avg GLCM correlation

Time (s)=1.72 (Panel=14)