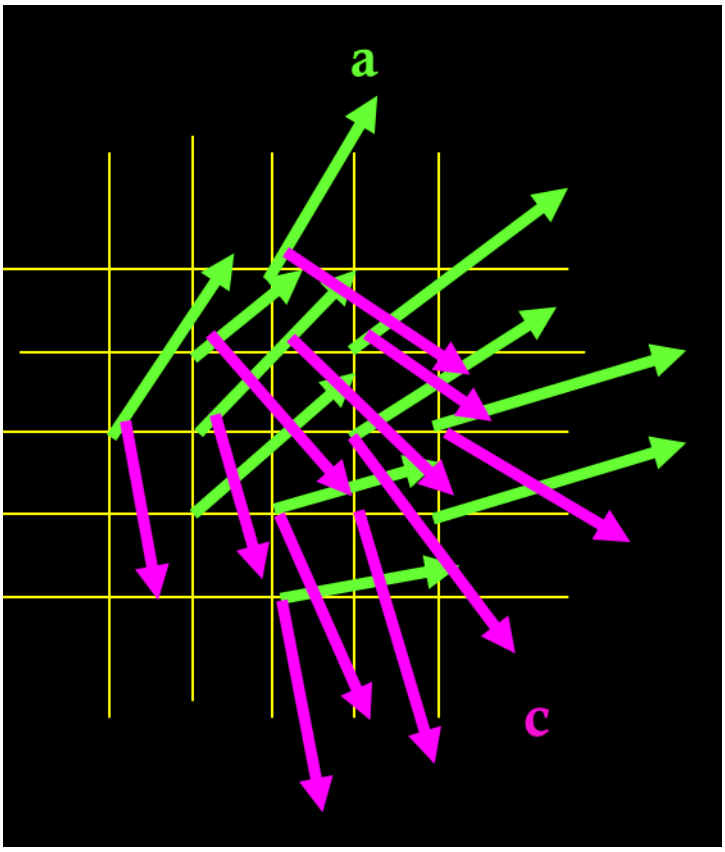
VECTOR CORRELATION OF TWO ATTRIBUTES - PROGRAM vector_correlate3d

Contents

Vector correlation

Some attributes are better represented as a vector (they have intensity as well as direction) e.g. curvature. Suppose you have attribute vectors ***a*** and vector ***c***, we can create vector correlation between these two attributes:



$$\text{colinear component}: \mathbf{a}\cdot\mathbf{c} = a_x c_x + a_y c_y$$

$$\text{orthogonal component}: \mathbf{a}\times\mathbf{c} = a_x c_y - a_y c_x$$

$$\mathrm{r}_{\text{colinear}} = \frac{\sum_{j=1}^{J}\left(a_x^j c_x^j + a_y^j c_y^j\right)}{\left[\sum_{j=1}^{J} a_x^j a_x^j + a_y^j a_y^j\right]^{1/2}\left[\sum_{j=1}^{J} c_x^j c_x^j + c_y^j c_y^j\right]^{1/2}}$$

$$\mathrm{r}_{\text{orthogonal}} = \frac{\sum_{j=1}^{J}\left(a_x^j c_y^j - a_y^j c_x^j\right)}{\left[\sum_{j=1}^{J} a_x^j a_x^j + a_y^j a_y^j\right]^{1/2}\left[\sum_{j=1}^{J} c_x^j c_x^j + c_y^j c_y^j\right]^{1/2}}$$

$$\|\mathbf{r}\| = \left(\mathrm{r}_{\text{colinear}}^2 + \mathrm{r}_{\text{orthogonal}}^2\right)^{1/2}$$

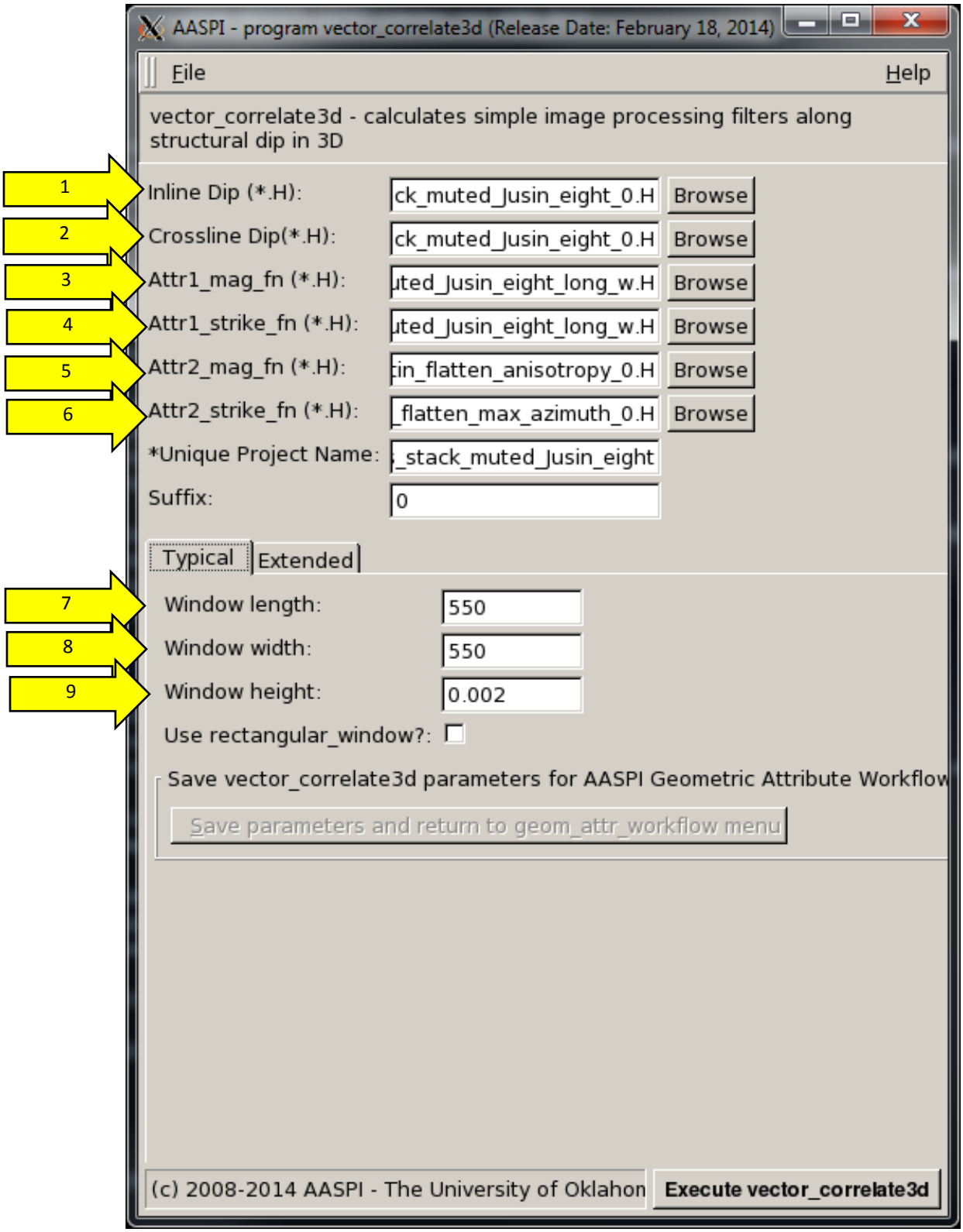
$$\psi = \arg(\mathbf{r}) = \mathrm{ATAN2}\left(\mathrm{r}_{\text{orthogonal}}, \mathrm{r}_{\text{colinear}}\right)$$

Launching the Graphical User Interface (GUI) - aaspi_vector_correlate3d

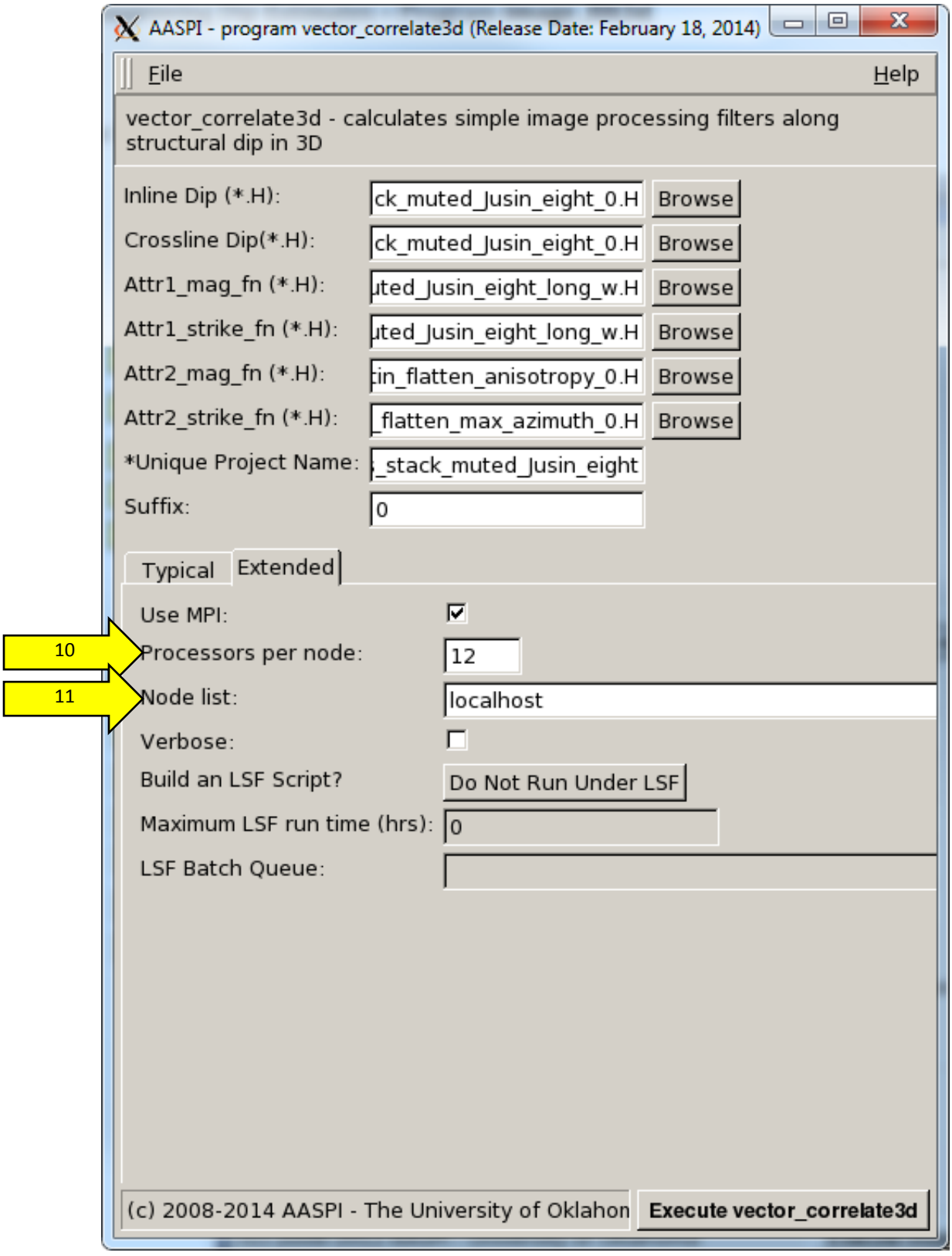
There are two ways to invoke **aaspi_vector_correlate3d** GUI: (1) either by typing it in on the command line, or (2) by choosing it on the upper right hand corner of the (poststack analysis) **aaspi_util GUI**.

In either manner, the following GUI appears.

Attribute Correlation: Program **vector_correlate3d**



Attribute Correlation: Program **vector_correlate3d**



Attribute Correlation: Program **vector_correlate3d**

Vector correlation3d has six input files. Click (1) browse inline dip file and similarly browse (2) crossline dip file, (3) magnitude file of first attribute (for example **k2_yourprojectname.H**), (4) strike of first attribute (for example **k2_strike_yourprojectname.H**), (5) magnitude of second attribute (intensity of AVAz anisotropy), (6) strike of second attribute (azimuth of AVAz anisotropy). You can also specify project name and suffix as you want. Specify the window length in (7) inline direction and (8) in cross line in ft or meter to calculate the vector correlation. I have used a square window of 550 ft by 550 ft, which is 5 times of inline and crossline space. (9) I have kept time window length (window height) as 0.002s or 2 milliseconds.

If you click the extended option in the middle of the menu, you will obtain the window above. We will wish to use MPI so place a checkmark in front of it. I am logged onto a machine called 'tripolite.ou.edu' and wish to run locally so (11) I type localhost as my node list. You may have to type your server name. I used (10) 12 processors on tripolite. Nodes typically will have 1, 2, 4, 8, 16, 32, or even 64 processors in a single box.

```
 2:all results sent back to master for line         196
 3:all results sent back to master for line         196
 4:all results sent back to master for line         196
 5:all results sent back to master for line         196
 6:all results sent back to master for line         196
 7:all results sent back to master for line         196
 8:all results sent back to master for line         196
 9:all results sent back to master for line         196
10:all results sent back to master for line         196
11:all results sent back to master for line         196
12:all results sent back to master for line         196
13:all results sent back to master for line         196
14:all results sent back to master for line         196
```

Attribute Correlation: Program **vector_correlate3d**

```
process                                  task       time (hr)  time/trace (s)
 9:                              read  data           0.000           0.000
 9:                         send data via MPI          0.000           0.000
 9:                      receive data via MPI          0.000           0.000
 9:                      send results via MPI          0.000           0.000
 9:                   receive results via MPI          0.000           0.000
 9:                      calculate attributes          0.000           0.000
 9:                     write results to disk          0.000           0.000
 9:                              total time         233.722         607.069
         6 :end loop over lines
         6 number of traces processed:         1386
process                                  task       time (hr)  time/trace (s)
 6:                              read  data           0.000           0.000
 6:                         send data via MPI          0.000           0.000
 6:                      receive data via MPI          0.000           0.000
 6:                      send results via MPI          0.000           0.000
 6:                   receive results via MPI          0.000           0.000
 6:                      calculate attributes          0.000           0.000
 6:                     write results to disk          0.000           0.000
 6:                              total time         233.722         607.069
11:                              read  data           0.000           0.000
11:                         send data via MPI          0.000           0.000
11:                      receive data via MPI          0.000           0.000
11:                      send results via MPI          0.000           0.000
11:                   receive results via MPI          0.000           0.000
11:                      calculate attributes          0.000           0.000
11:                     write results to disk          0.000           0.000
11:                              total time         233.722         607.069
process                                  task       time (hr)  time/trace (s)
 0:                              read  data         233.721      841393.939
 0:                         send data via MPI        233.699      841315.126
 0:                      receive data via MPI          0.000           0.001
 0:                      send results via MPI          0.000           0.001
 0:                   receive results via MPI        233.722      841397.626
 0:                      calculate attributes          0.000           0.001
        25 :end loop over lines
 0:                     write results to disk        233.722      841397.626
 0:                              total time         233.722      841397.626
```

The picture above shows running log, and the picture below show the program is finished.

```
 0 :normal completion. program crosscorrelate
10 :normal completion. program crosscorrelate
 3 :normal completion. program crosscorrelate
 9 :normal completion. program crosscorrelate
12 :normal completion. program crosscorrelate
 6 :normal completion. program crosscorrelate
 1 :normal completion. program crosscorrelate
13 :normal completion. program crosscorrelate
 2 :normal completion. program crosscorrelate
 8 :normal completion. program crosscorrelate
 7 :normal completion. program crosscorrelate
11 :normal completion. program crosscorrelate
 4 :normal completion. program crosscorrelate
22 :normal completion. program crosscorrelate
31 :normal completion. program crosscorrelate
30 :normal completion. program crosscorrelate
15 :normal completion. program crosscorrelate
25 :normal completion. program crosscorrelate
29 :normal completion. program crosscorrelate
27 :normal completion. program crosscorrelate
26 :normal completion. program crosscorrelate
28 :normal completion. program crosscorrelate
19 :normal completion. program crosscorrelate
```

At last you can generate four files, they are correlation intensity value and correlation strike.

```
-rw-r--r--. 1 shiguang aaspi      2023 Aug 24 19:06 vector_correlate3d_anisotropy_k2_stack_muted_Jusin_eight_flatten_550.H@@
-rw-r--r--. 1 shiguang aaspi      8367 Aug 24 19:06 vector_correlate3d_anisotropy_k2_stack_muted_Jusin_eight_flatten_550.H
-rw-r--r--. 1 shiguang aaspi      2030 Aug 24 19:06 vector_correlate3d_strike_anisotropy_k2_stack_muted_Jusin_eight_flatten_550.H@@
-rw-r--r--. 1 shiguang aaspi      8381 Aug 24 19:06 vector_correlate3d_strike_anisotropy_k2_stack_muted_Jusin_eight_flatten_550.H
```

Examples

Here I show the one example regarding this program:

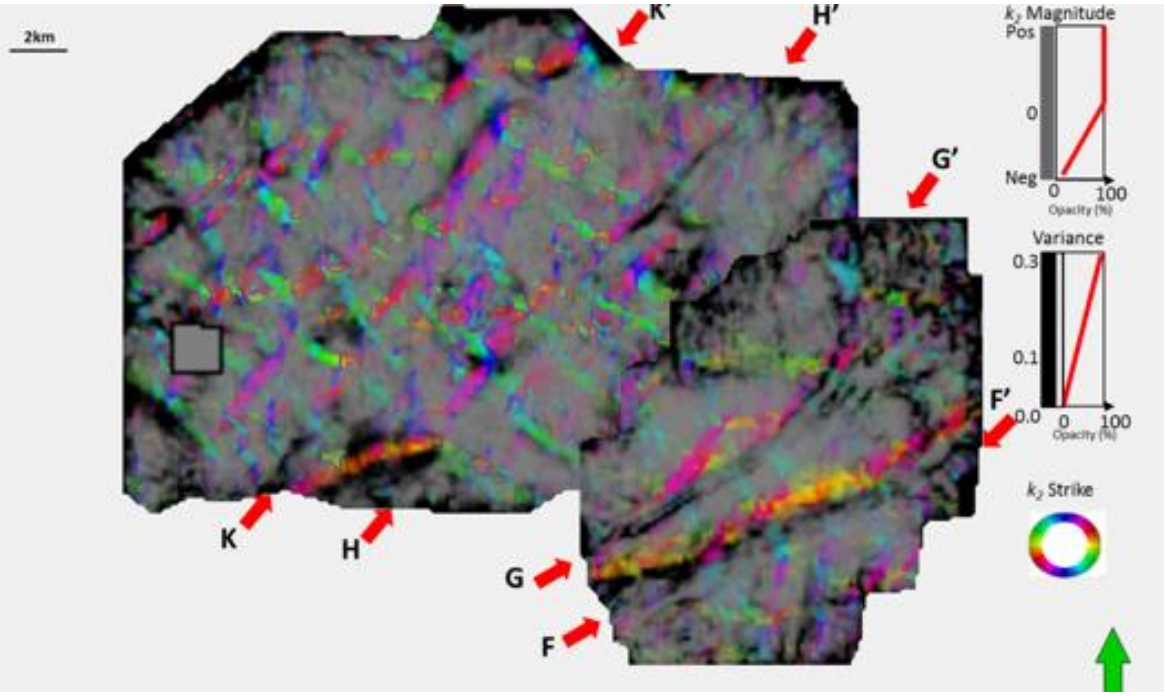


Figure 1: Phantom Horizon slices 20 ms above the top Viola limestone through strike of most positive curvature k_2 modulated by its value co-rendered by variance.

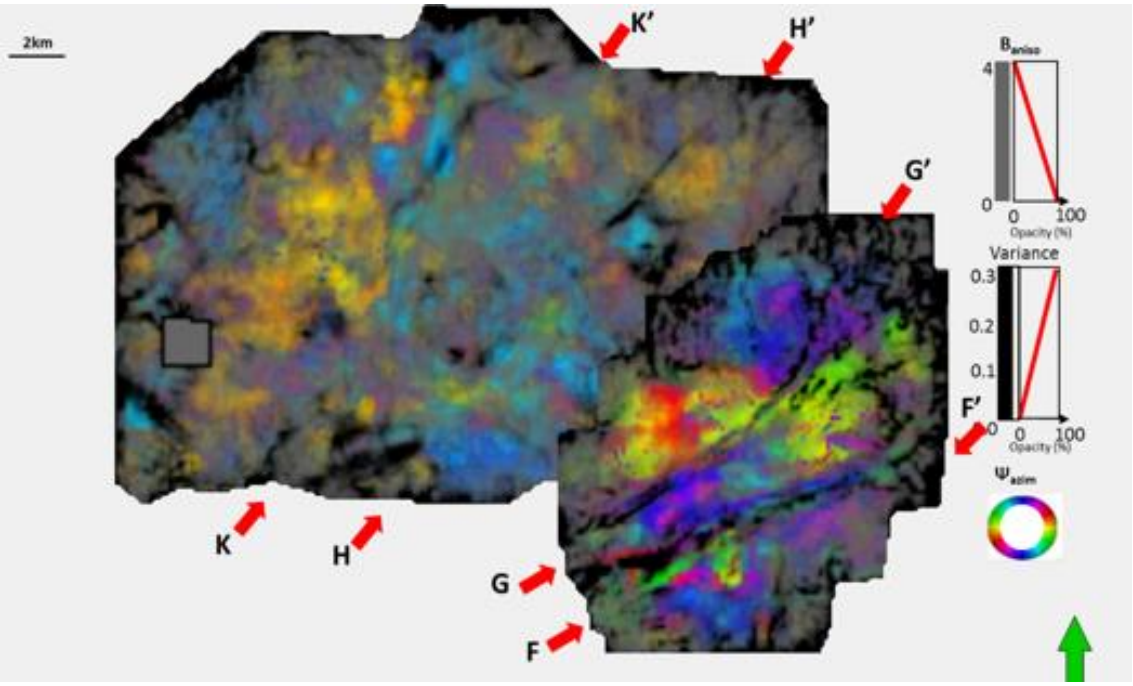


Figure 2: Phantom Horizon slices 20 ms above the top Viola limestone through anisotropy strike Ψ_{azim} modulated by its value B_{aniso} co-rendered by (a) variance

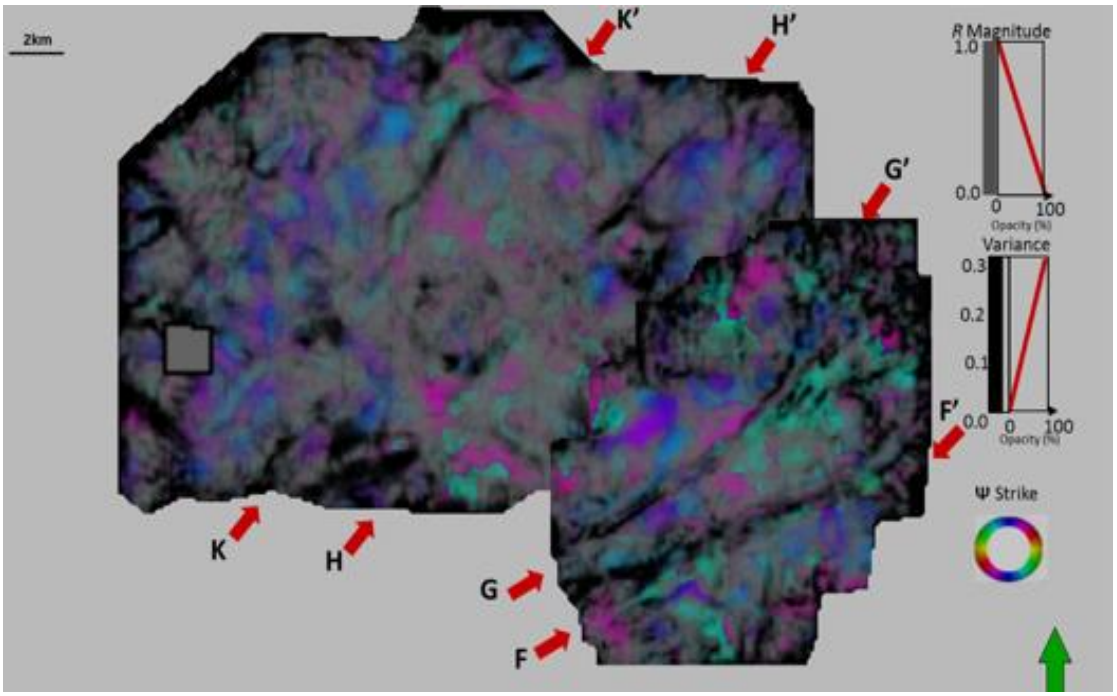


Figure 3: Phantom Horizon slices 20 ms above the top Viola limestone through strike of new vector correlation (AVAz and k_2) modulated by its value co-rendered by variance.