Steering Benchmark tests

A. Huck
dGB Earth Sciences

Introduction

- Recently, a new and fast way of producing a steering cube has been added to OpendTect.

- The final output quality is not only dependent on the steering algorithm, but also on 3 other additional decisions:
  - Size of the calculation cube
  - Presence and value of a dip limit
  - Presence and nature of an additional median dip filtering

Overview

- Input and naming conventions.
- Theoretical and real calculation speed.
- Comparisons.
- Recommendations.

Input data

- The input cube as 817811 traces with each 1551 samples, step 4 ms so 6200 ms.

- For the processing, a cluster of 6 PC’s was used.

Nomenclature

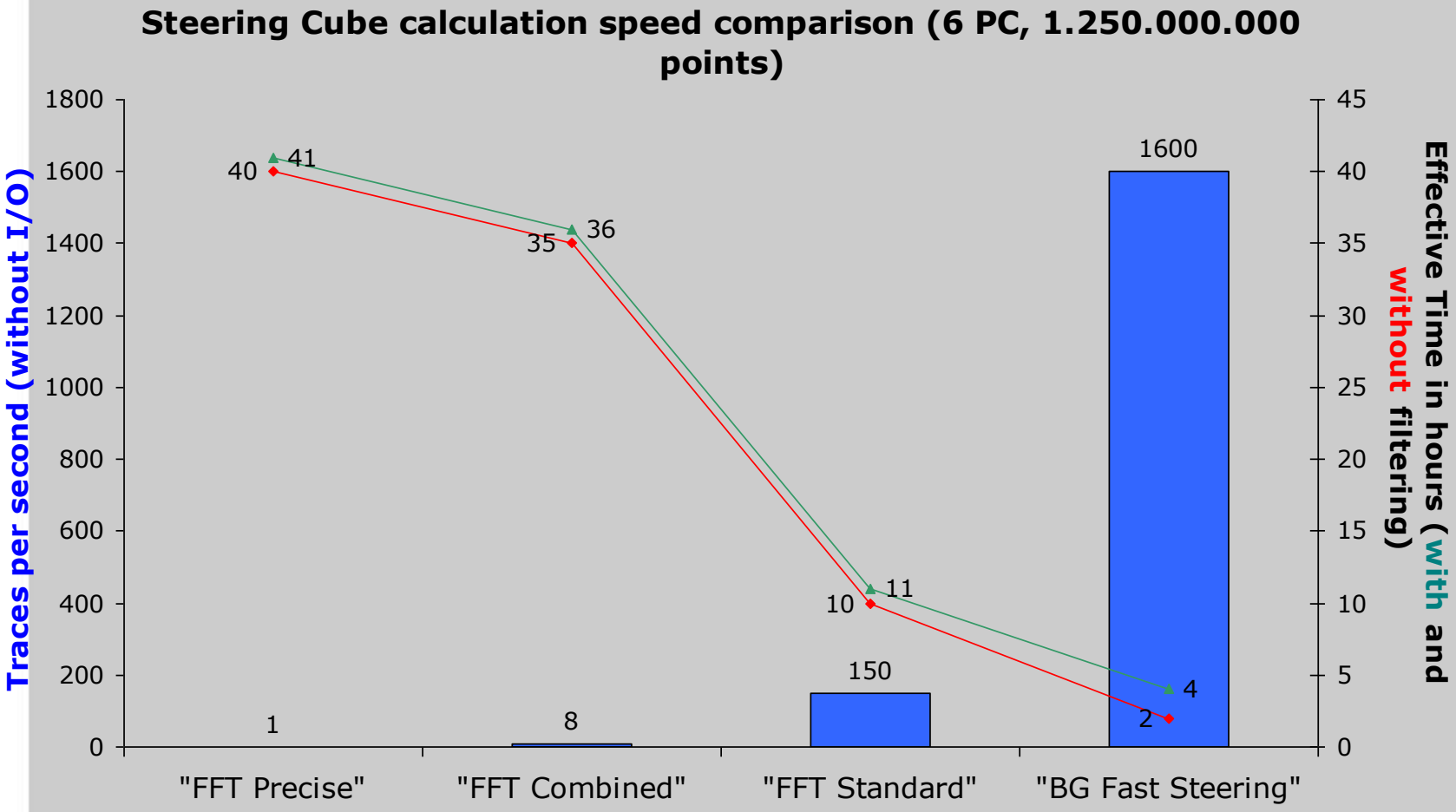
In this analysis, we will use the following conventions:

- The standard steering algorithm will be called FFT+++. (p=+ in the filenames)
- The FFT combined will be called FFT++.
- The FFT precise will be called FFT+.
- The calculation cube size will be specified after the algorithm (e.g. FFT7, BG5, etc.).
- The limit of dip will be called maxdipXXX, where XXX is the limit in µs/m.
- Additional filtering are called medXYZ, where X is the inline stepout, Y the crossline stepout and Z the sample stepout.

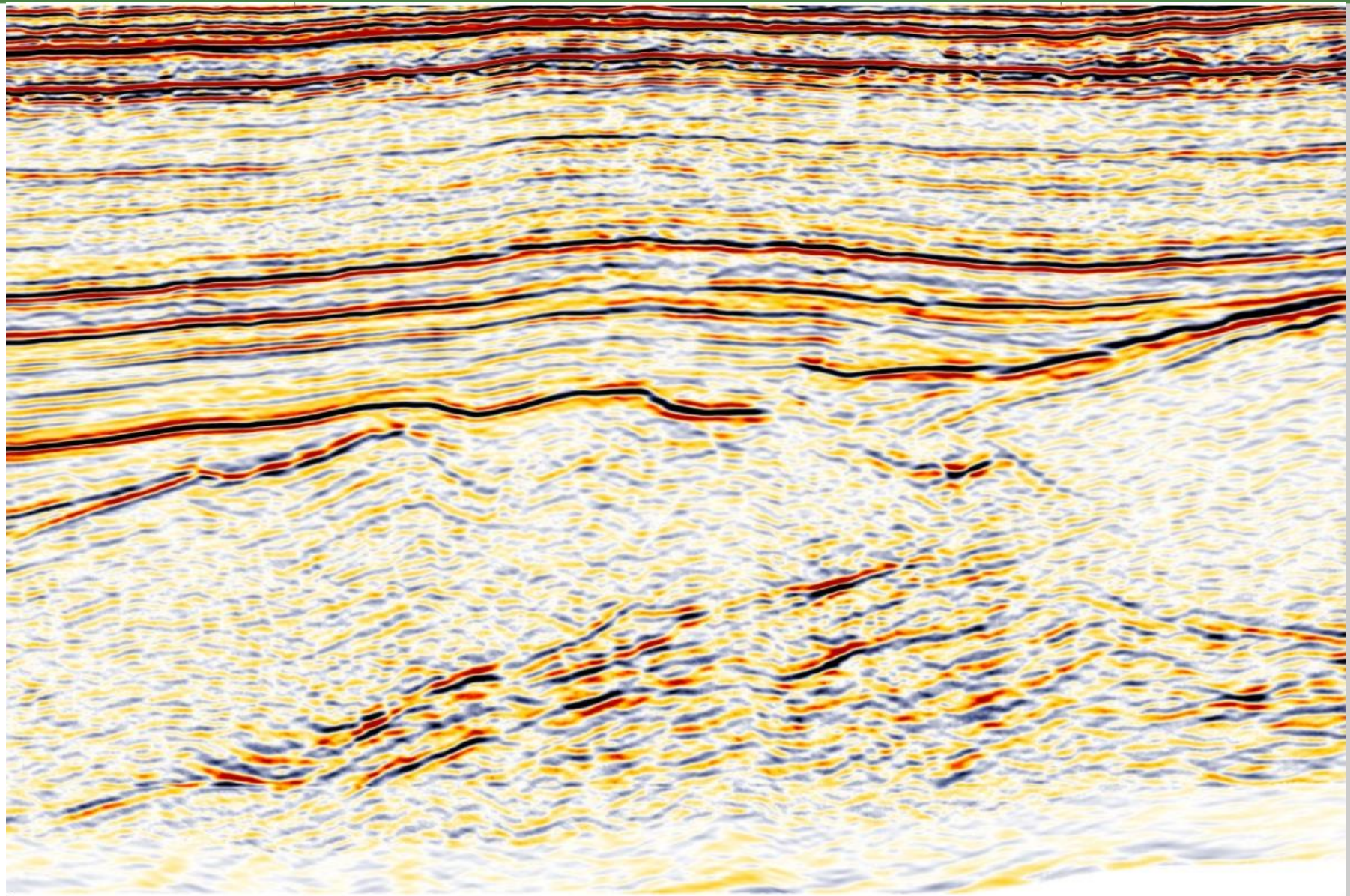
For example FFT7ppp_maxdip_300_med112.jpg refers to a steering using the FFT standard algorithm, with a dip limit of 300 µs/m and a median filtering of lateral stepout 1 and sample stepout 2.

This names are consistent with the ones you will find in OpendTect.

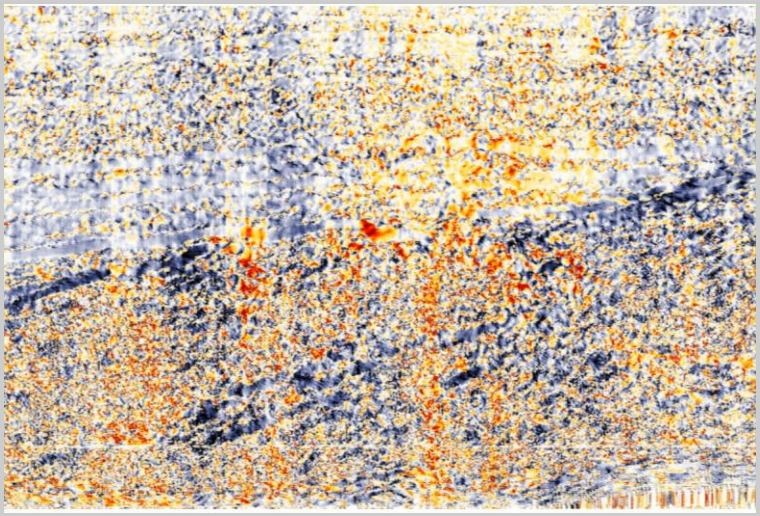
Comparison of relative speeds of steering cube algorithms



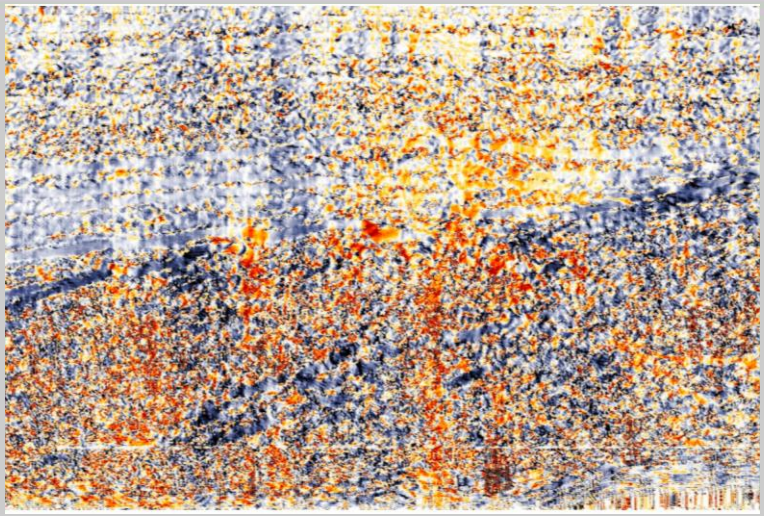
Input data along an inline



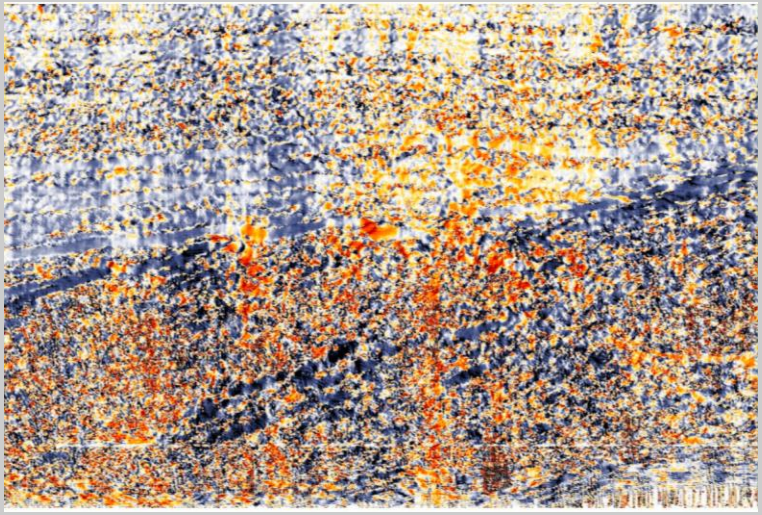
Raw Crossline Dip, Cube size of 3



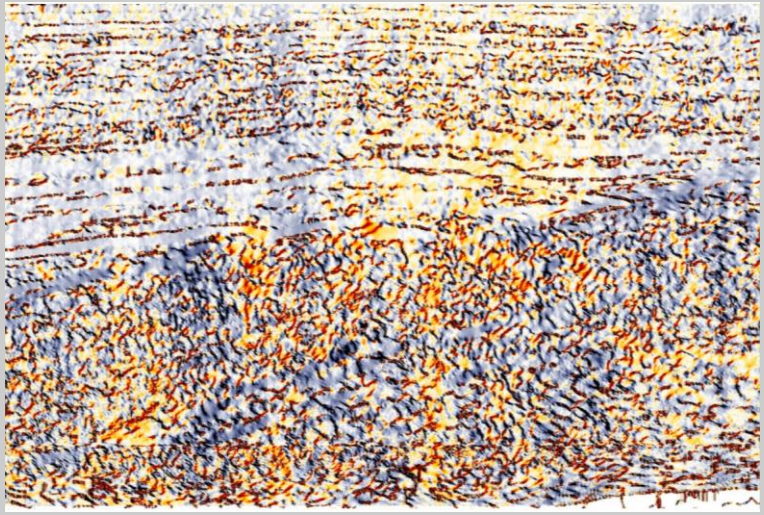
FFT3+++



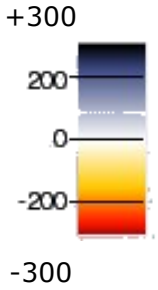
FFT3++



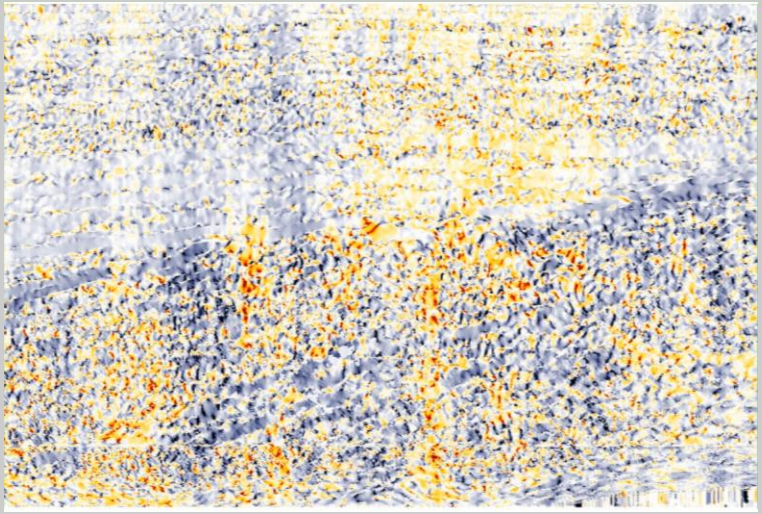
FFT3+



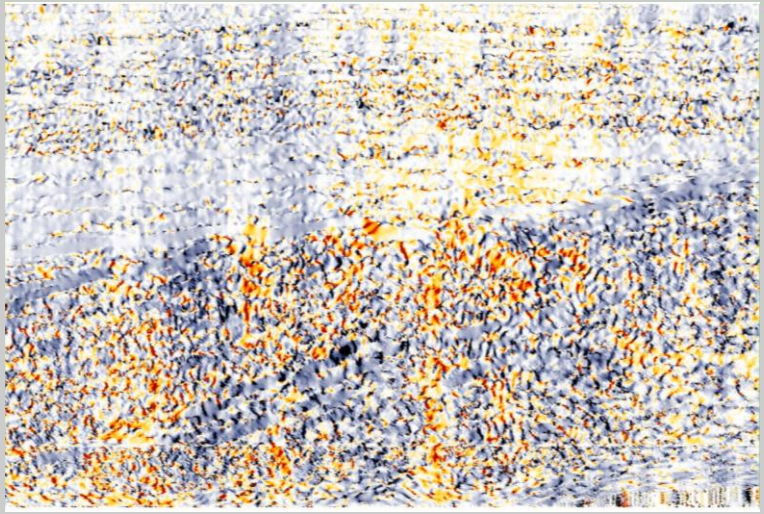
BG3



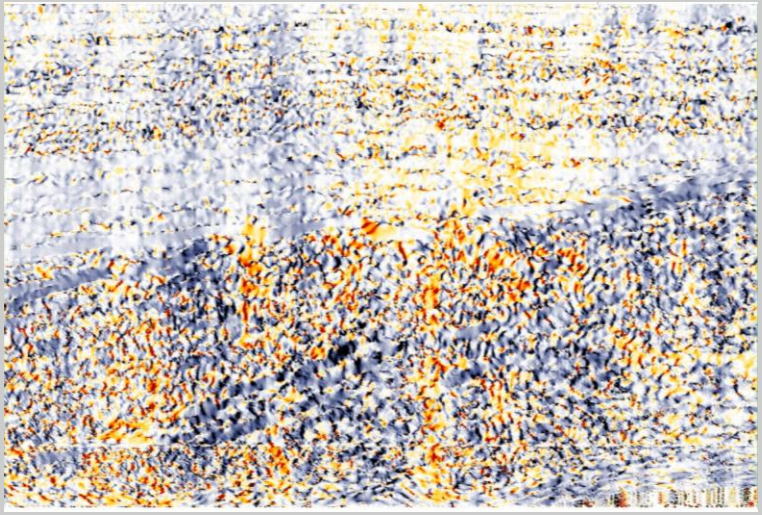
Raw Crossline Dip, Cube size of 5



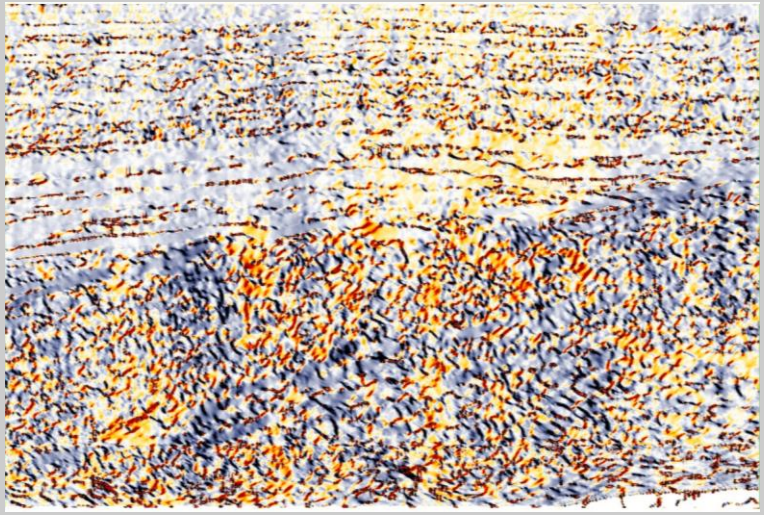
FFT5+++



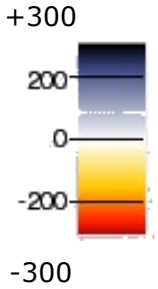
FFT5++



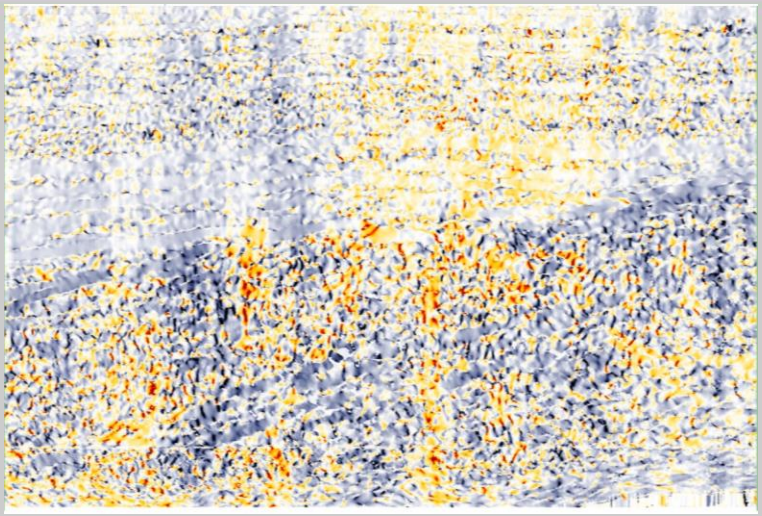
FFT5+



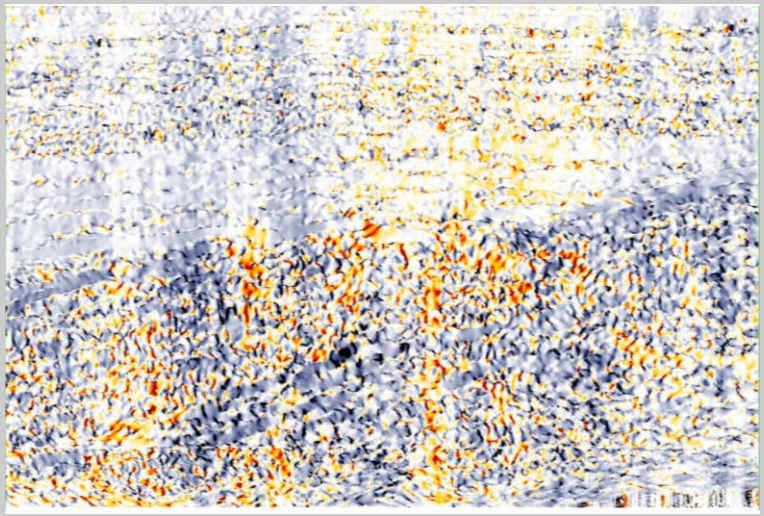
BG5



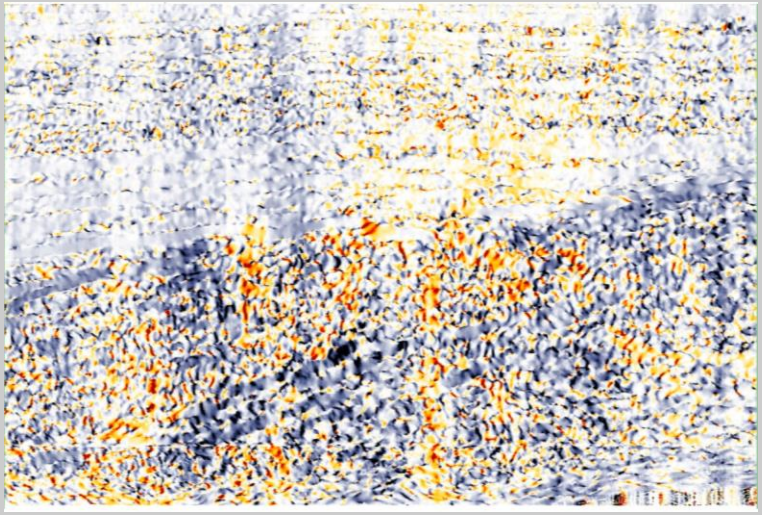
Raw Crossline Dip, Cube size of 7



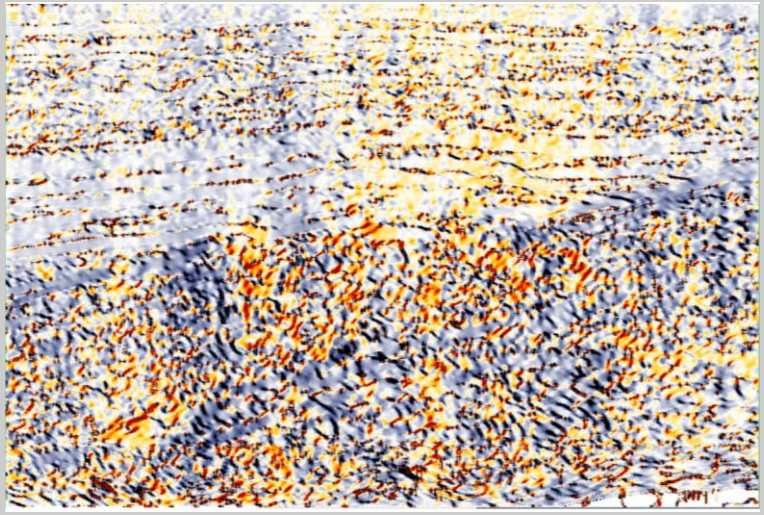
FFT7+++



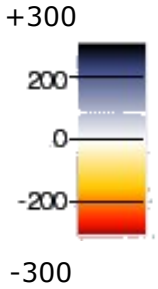
FFT7++



FFT7+



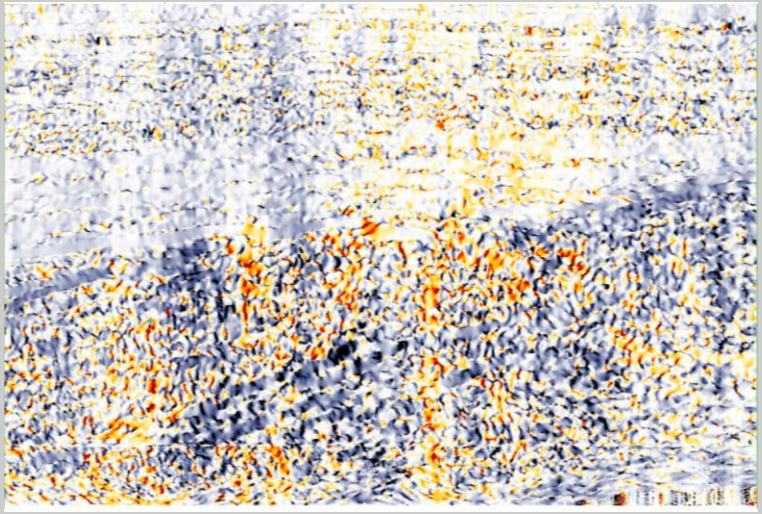
BG7



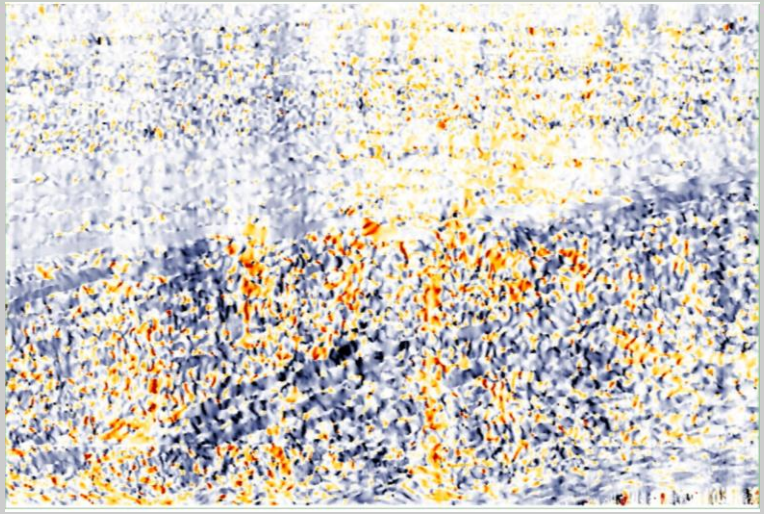
Conclusions #1

- The FFT+++ algorithm is fast but apparently often produces erroneous results in high dip areas.
- There is not much differences between FFT++ and FFT+, both in speed and in accuracy.
- To avoid getting a noisy steering cube, the calculation cube size of the FFT algorithms has to be at least 5 or 7. The latter is our default setting.
- The BG algorithm has a different behaviour:
  - A cube size of 3 looks enough.
  - A raw steering cube is unusable, a median filtering appears to be mandatory.

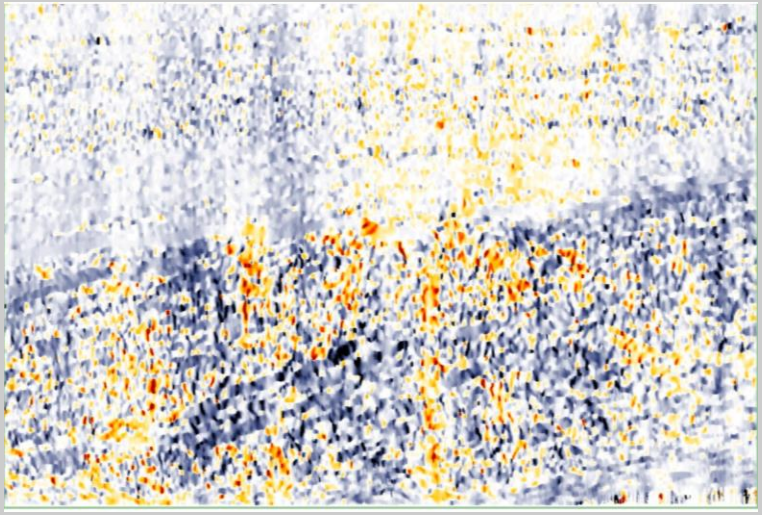
Raw Crossline Dip, Filtering of FFT+



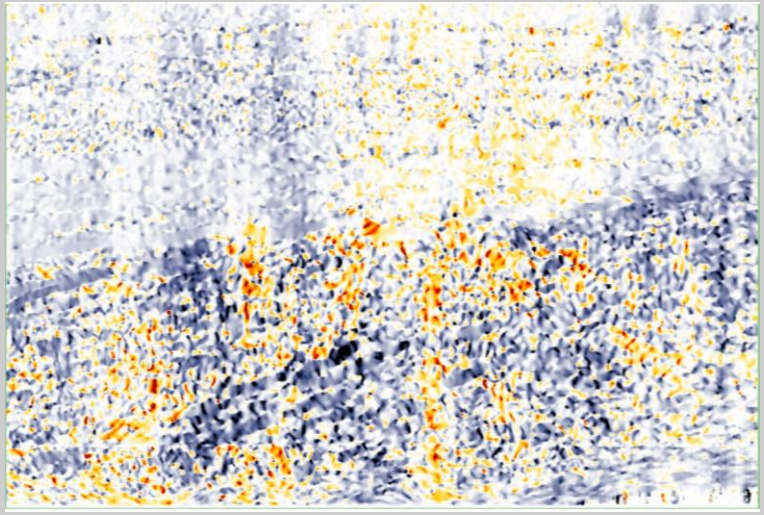
FFT7+ raw



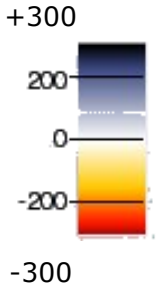
FFT7+ med002



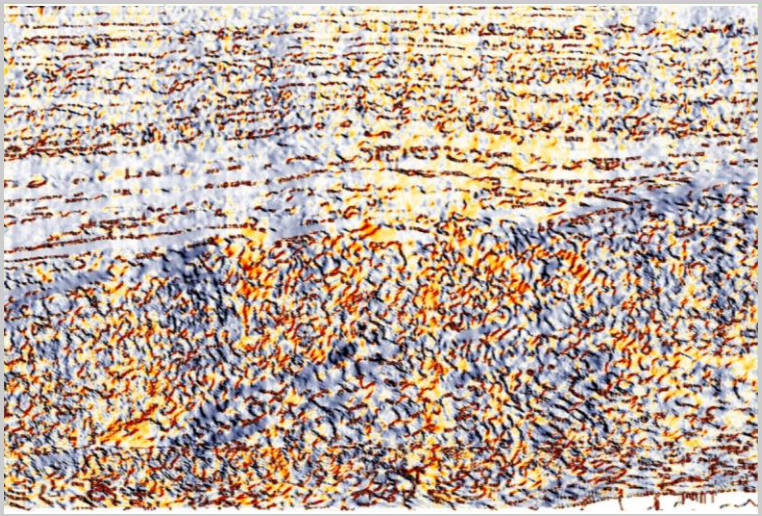
FFT7+ med004



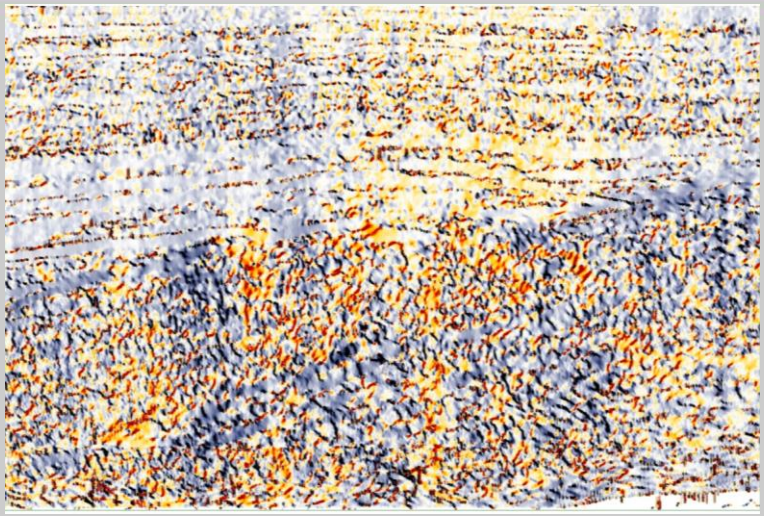
FFT7+ med112



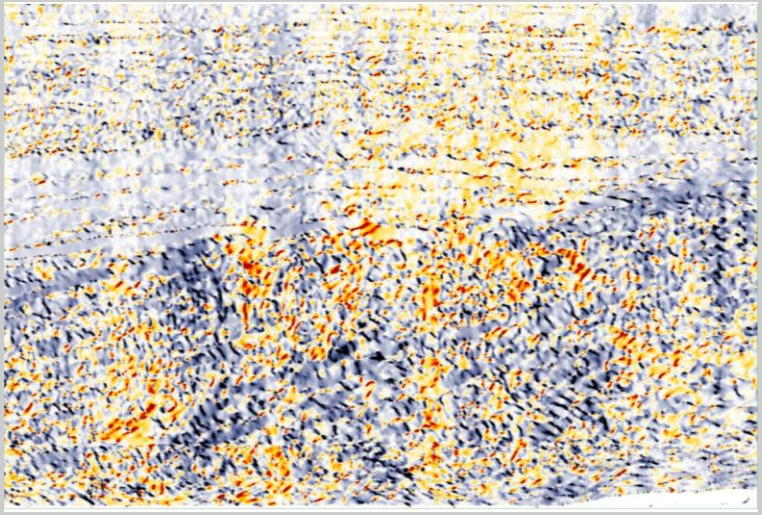
Raw Crossline Dip, Filtering results of BG



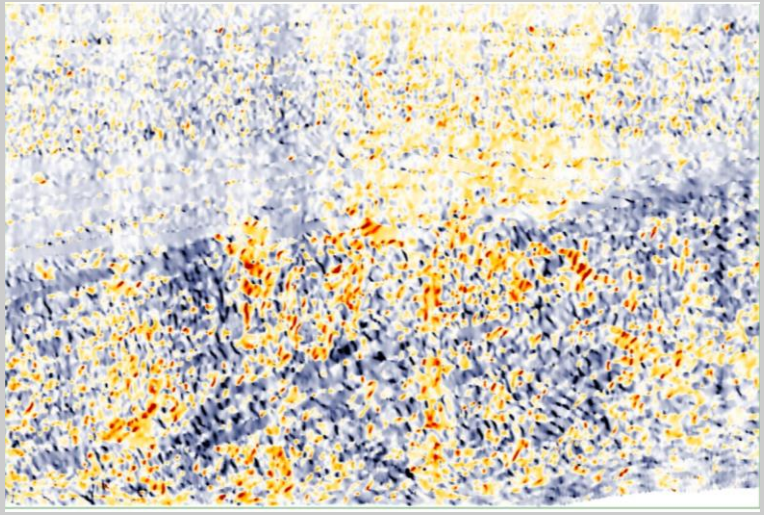
BG3 raw



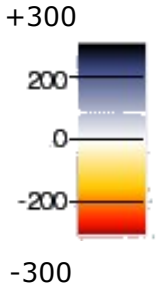
BG3 med002



BG3 med111



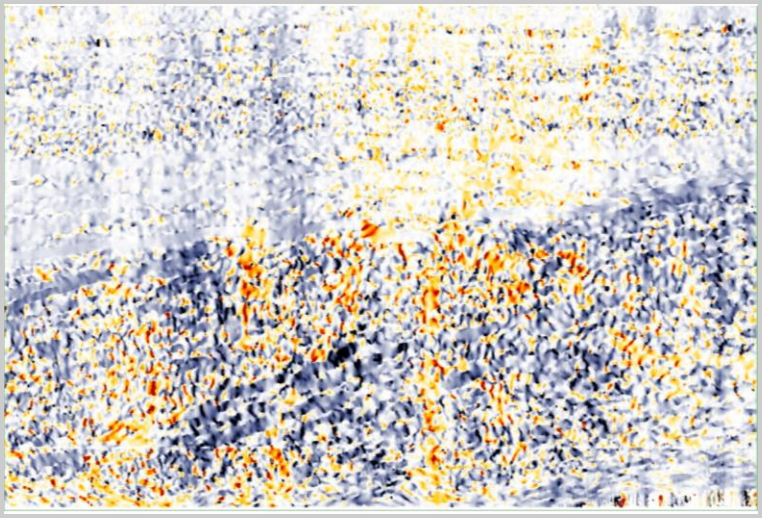
BG3 med113



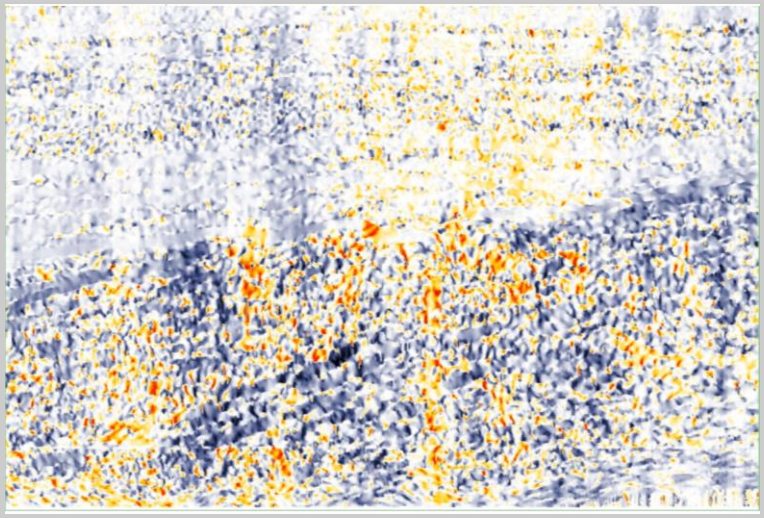
Conclusions #2

- No lateral filtering and only a small vertical filtering gives the best result with the FFT algorithm.
- The BG steering has to be filtered in the lateral and vertical direction. The best results were obtained with med113.
- After filtering, the outputs FFT7+ med002 and BG3 med113 are very similar in accuracy, but this latter is produced 10 times faster!

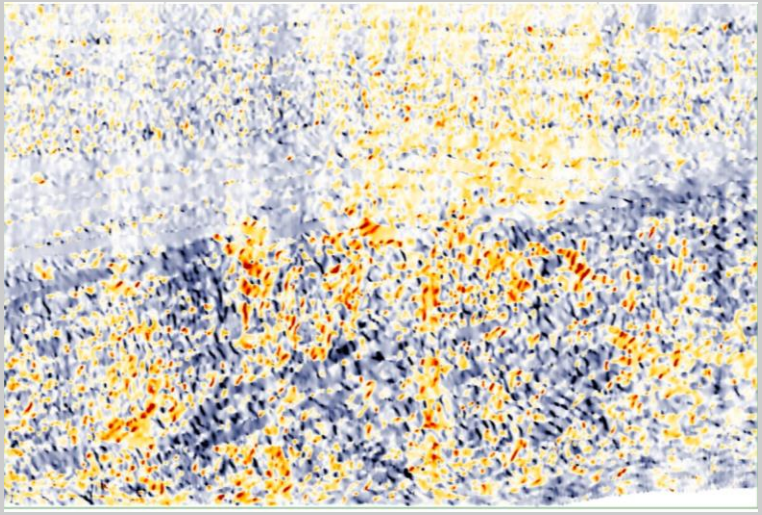
Raw Crossline Dip, with or without a dip limit



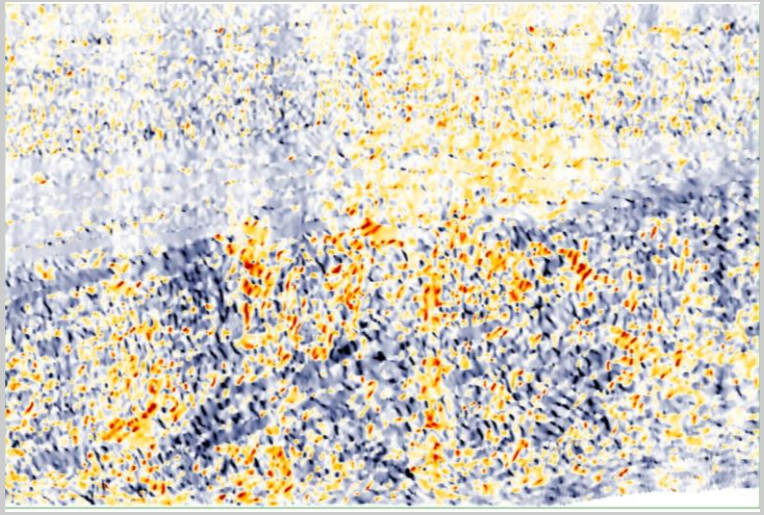
FFT7+ med002



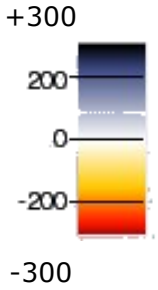
FFT7+ maxdip300 med002



BG3 med113



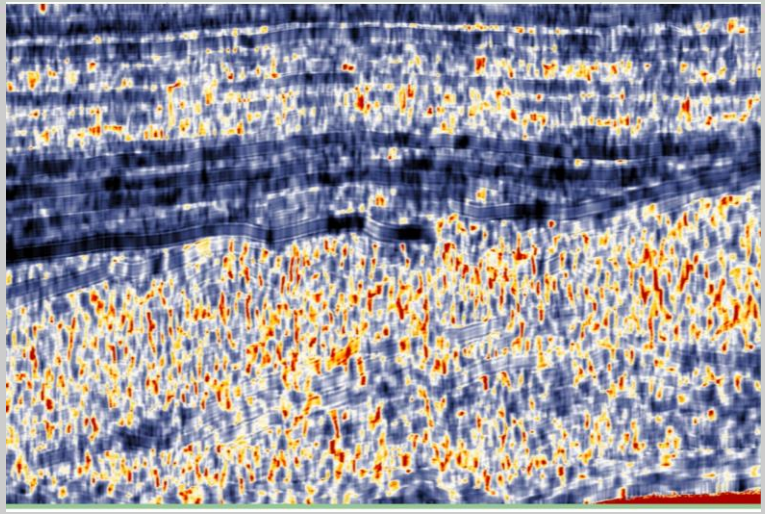
BG3 maxdip300 med113



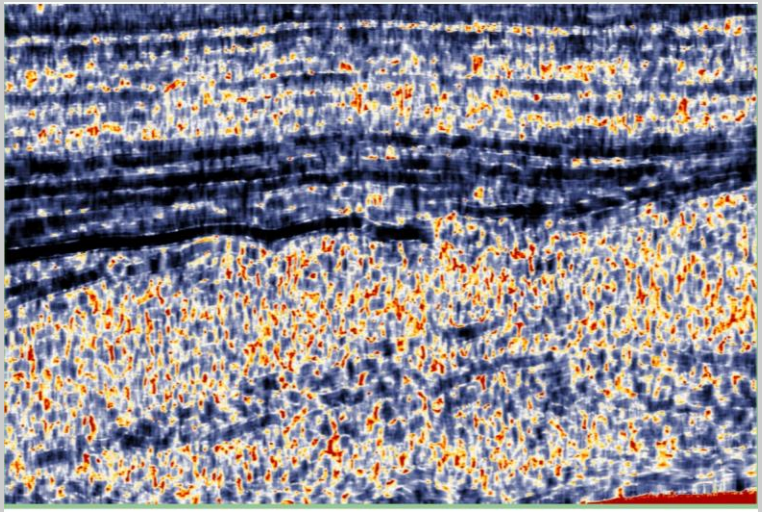
Conclusions #3

- Adding a dip limit during the processing does not affect the speed of the algorithms.
- The final result is strictly identical when the dip is lower than the limit L, and the extreme values are rounded toward L.
- Using a limit requires a priori knowledge and at the end remains the choice of the interpreter.

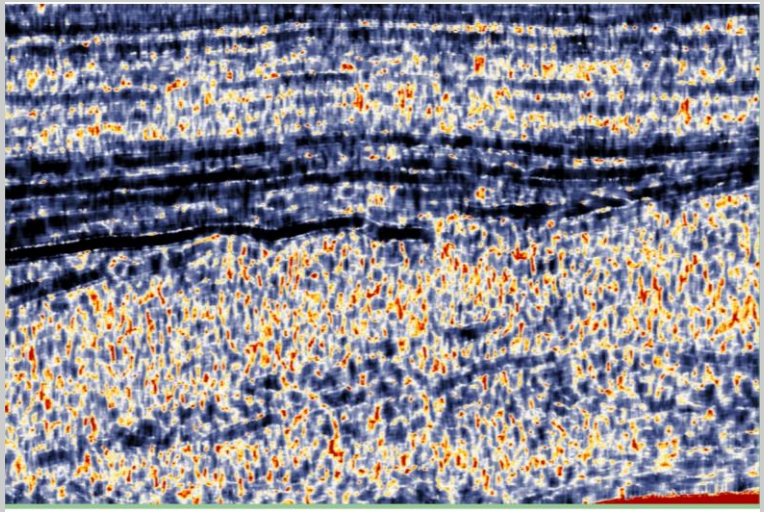
Steering in an attribute



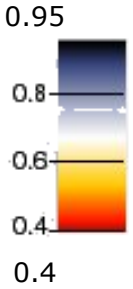
Non Steered Similarity [-32,32]ms



Full Steered Similarity [-32,32] ms
on FFT7+ med002



Full Steered Similarity [-32,32] ms
on BG3 med113



Conclusions and recommendations

At the end, a filtered steering using BG's fast algorithm seems to fit 95% of the situations as well as the FFT precise algorithm.

If for some reason it is not the case, you can still go back and spend much more time in producing a steering cube using the FFT precise algorithm.