Advances in seismic interpretation using new volume visualization techniques

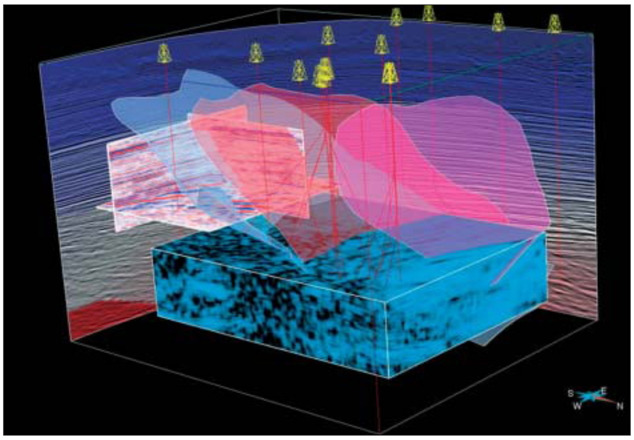
Laurent Castanie[1,2], Fabien Bosquet[1], and Bruno Levy[3] discuss new techniques in volume rendering to improve the visualization of 3D seismic structures.

As the use of 3D seismic interpretation continues to become part of the main stream work process with the industry, visualization techniques also continue to evolve as software and hardware improves. In the past 10 years, volume rendering tools have been progressively adopted by the geophysical community as the emergence of high-end graphics workstations with 3D texture capabilities made real-time volume rendering possible. Many interactive volume rendering packages are now available for seismic interpretation. However, interpretation is still mostly done in 2D. Using classical volume rendering with high spatial frequencies of seismic data make it very difficult to produce meaningful volume images and often results in cluttered useless images.

Volume rendering tools are now part of most seismic interpretation packages. However, most do not provide better insight into the 3D structures of seismic data because the noise and high spatial frequencies in the data prevent classical volume rendering from capturing relevant information. As an improvement, we propose more suitable high quality volume rendering algorithms based on a pre-integration of the transfer function (i.e. colour map) that use the capabilities of the recent programmable graphics processing units (GPUs) of new graphics cards. This results in a versatile multimodal (multi attributes) volume rendering system. This system is dedicated to the efficient combined visualization of several volumes. Coupled with high quality volume rendering algorithms, it makes it possible to visualize isosurfaces interactively and paint them with another attribute. This is done without explicit extraction of the surface. By this way, isosurfaces of distance maps to faults or well paths can be interactively extracted and painted with the seismic data.

In this article, we adapt high quality volume rendering algorithms from the computer graphics industry to improve the imaging. We have found these algorithms more suitable for seismic data analysis than classical ones. They use the capabilities of the recent programmable graphics hardware. In addition, we will present a versatile multimodal volume rendering system that enables the efficient co-visualization of several volumes.

[1] *Earth Decision Sciences*
[2] *Gocad Research Group*
[3] *INRIA Lorraine (Project ALICE)*



Sample of probes and arbitrary sub volumes for interpretation.

Volume Rendering in a Nutshell

In volume rendering, the volume is considered as a semi-transparent medium. For each pixel of the screen, the computation of a volume rendering integral accumulates the voxels contribution along a virtual viewing ray. Assuming that the viewing ray $\mathbf{x}(\lambda)$ is parametrized by the distance λ to the viewpoint, and that $c(\mathbf{x})$ and $\zeta(\mathbf{x})$ respectively define the color and extinction coefficients for any point $\mathbf{x}$ in space, the volume rendering integral is defined as:

$$I = \int_0^D c(\mathbf{x}(\lambda)) \cdot \exp\left(- \int_0^\lambda \zeta(\mathbf{x}(\lambda'))d\lambda' \right) d\lambda$$

where D is the distance to the point where the viewing ray leaves the data set and I the pixel's final color intensity. This integral simulates a local production of color in space attenuated by extinction coefficients between the point of emission and the viewpoint. Typically, colour and extinction coefficients are assigned by a transfer function that maps the scalar property space to a **RGBA** {Red, Green, Blue, Alpha} colour space where **A** is the opacity or extinction coefficient. Moreover, the gradient of the scalar property stored in the volume can be used as a normal vector to produce lighting effects and improve the perception of the 3D structures.

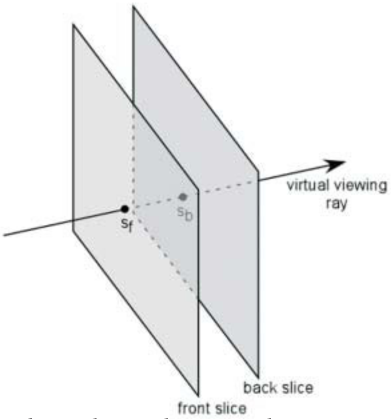


Figure 1 *High quality volume rendering considers a slab between each couple of successive sampling slices. The intersection of the virtual viewing ray with a slab is a segment - sf and sb are the scalar values respectively sampled on the front and back slices of the slab.*

High quality volume rendering

In classical 3D texture based volume rendering, the volume rendering integral is computed with a series of view-aligned sampling slices composited in a back-to-front order. Considering the virtual viewing ray leaving from each pixel of the screen in the view direction, the rendering slices decompose this ray in a series of segments composited in a back-to-front order. Classical slicing-based volume rendering approximates the colour and opacity of the segment with the colour and opacity sampled on the front slice. Used this way, the contribution of matter between each slice is neglected, and increasing the number of slices is the only way of accurately capturing relevant structures. However, even with recent powerful graphics hardware, increasing the number of slices (to increase accuracy) severely affects performance.

Today, high quality volume rendering introduced by Engel et al. (2001) uses slabs of data instead of slices. It is based on a pre-integration of the volume rendering integral for all possible couples of front and back slice sample values, respectively sf and sb in Figure 1. This pre-integration is done in the transfer function space in a pre-processing step. The result is stored in a 2D texture and fetched at rendering time from the values sf and sb stored on the front and back sample slices. Texture access is done in a fragment program running on the GPU. The values on the front and back sample slices are fetched from a 3D texture, then the pre-integration is obtained from the pre-computed 2D texture to produce the final image.

Pre-integrated volume rendering has been further adapted to produce high quality lighting effects by Lum et al. (2004). To implement high quality volume rendering for seismic data, we need to consider the specificity of the data. As shown in Figure 3, data values in a seismic volume usually have a Gaussian distribution. According to Brown (2004), the relevant information for the structural interpretation of seismic data is contained in the moderate amplitudes, i.e. between the mean and the extrema of the distribution. Extrema are local anomalies corresponding to fluids and mean values are the most subject to noise.

As shown in Figure 3, most interpretive time is therefore spent on these moderate amplitudes in a narrow range of property values. This requires specific transfer functions, most of the time containing a series of spikes (i.e. transfer functions with few scalar property values mapped to opaque colours). In this case, classical slicing-based volume rendering fails to capture the relevant structures in the volume as it misses the high frequencies in the transfer function. Note that neither super-sampling (Figure 2-A) nor lighting computations (Figure 2B) improve the result.

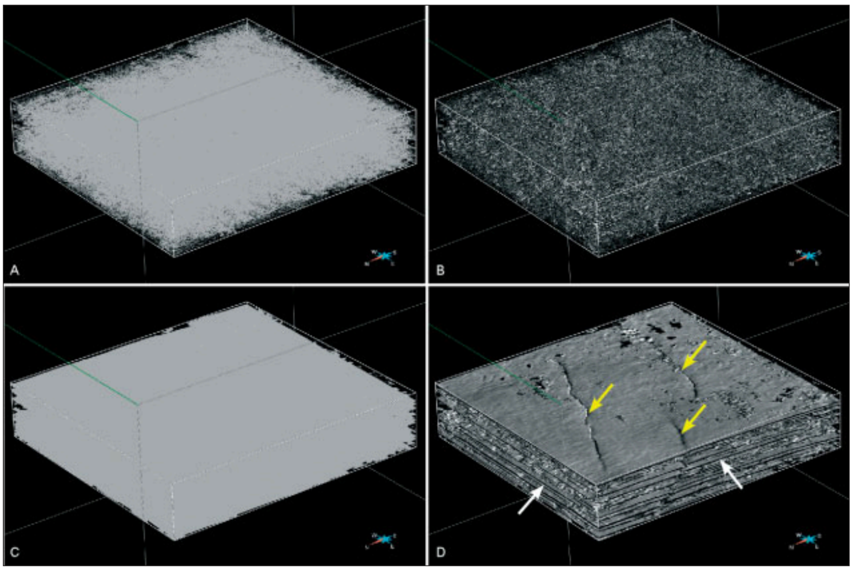


Figure 2 *Volume rendering of seismic amplitudes in a 3D probe of size 480x480x510 (roaming within a volume of size 2608x661x811, 32 bits per voxel) with a transfer function containing a series of spikes in the moderate amplitudes. (A) and (B) are oversampled with 1000 slices. (C) and (D) are rendered with 600 slices. (A) and (B) show classical slicing-based volume rendering respectively without and with lighting. (C) and (D) show pre-integrated volume rendering respectively without and with lighting. (Yellow and white arrows respectively show faults and horizons). Note how the 'layer-cake' structure is revealed in (D), whereas it is completely invisible in the other pictures.*

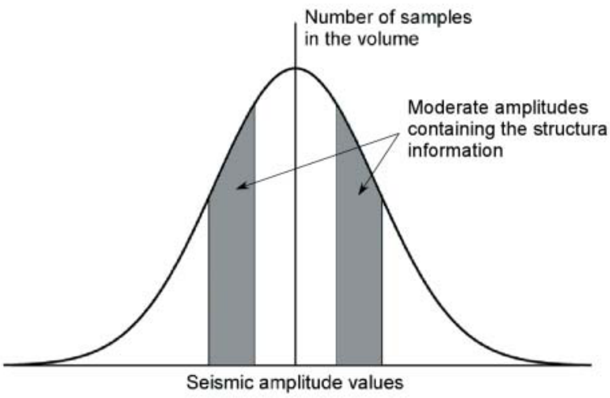


Figure 3 *Typical Gaussian distribution of seismic values. Moderate amplitudes correspond to relevant structural information (Modified from Brown).*

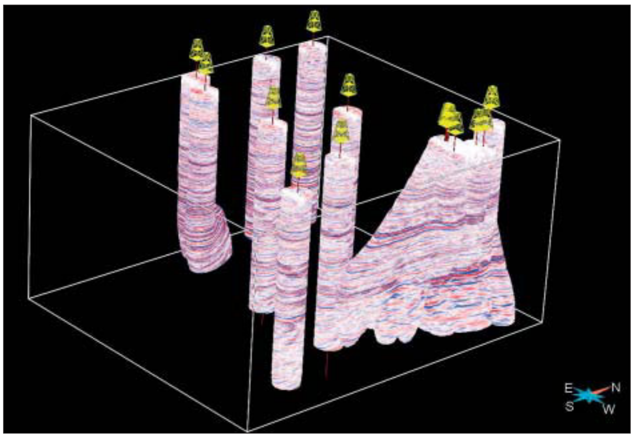


Figure 4 *Visualization of seismic amplitudes in the neighbourhood of a set of well paths thanks to volumetric clipping.*

In contrast, using pre-integrated volume rendering, as shown in Figure 2C, does not suffer from such singular transfer functions. Combined with lighting, it perfectly reproduces the structural information (Figure 2D). Faults traces are visible on the top face of the probe (yellow arrows), clearly affecting the horizons visible on the front faces (white arrows). Up until now, typical visualization workstations dedicated to seismic interpretation were limited to classical slicing-based volume rendering. They achieved good rendering quality at the price of being limited to smooth transfer functions and requiring high sampling rates (i.e. low performance).

Today, off-the-shelf PCs equipped with powerful GPUs expose new functionalities to supply high quality rendering techniques, such as pre-integrated volume rendering.

Multimodal volume rendering

As discussed in Marsh et al. (2000), seismic interpretation often generates multiple volumes of data. Such volumes can be either additional seismic attributes or structural information such as distance maps to a series of extracted surfaces (faults and/or horizons), or perhaps, to a series wells. This results in multimodal data sets that can be rendered efficiently by combining multiple volumes at the same time.

To continue, we need to introduce the idea of 'layer'. Each volume needed to be combined defines a layer with an associated 1D transfer function. Three types of layers are defined and determine three possible ways for the user to interact with the transfer function:

- *Colour:* **RGBA** colour channels are freely edited in the transfer function.
- *Opacity:* **RGB** colour channels are fixed to one (white colour), while **A** can be modified.
- *Intensity:* A given colour is defined for the layer and its intensity is modulated with either a white or a black background along the transfer function. Opacity **A** is fixed to one.

At rendering time, each sampling point contains multiple scalar values corresponding to each attribute stored in the volume. For each defined layer, we get the corresponding scalar value and map it to the **RGBA** color channels with the associated transfer function. Layers are then combined by multiplying the **RGBA** color channels.

A typical application of this technique is volumetric clipping as introduced by Weiskopf et al. (2003). The neighbourhood of a horizon, a fault, or a set of well paths often contains meaningful information. Assuming that a distance map to this particular object has been computed, it can be bound to an opacity layer. It is then possible to make all distance values above a given threshold transparent and visualize an attribute in the neighbourhood of the object. In Figure 4, seismic amplitudes are visualized in the neighbourhood of a set of well paths using the volumetric clipping technique.

Another application use is the interactive display of painted isosurfaces. Classical isosurface extraction algorithms explicitly compute isosurfaces from a volume and store them as a new, independent representation. Using texturing hardware, texture-based isosurfaces are rendered in real-time on a per-pixel basis. They are visualized independently on the resolution of the dataset and are only dependent on the resolution of the display. Typically, pre-integrated volume rendering with specific transfer functions interactively renders such isosurfaces. The isovalue can be edited in real time as the time-consuming explicit extraction of the surface is not necessary using this technique. In the same way as distance maps can play the role of clipping volumes, they can now be used to render texture-based isosurfaces. Combined with using layers in multimodal volume rendering, we can interactively paint isosurfaces of a distance map to a specific object with seismic amplitudes. This provides new ways of visualizing seismic data. For example, semblance can be displayed at a given distance to

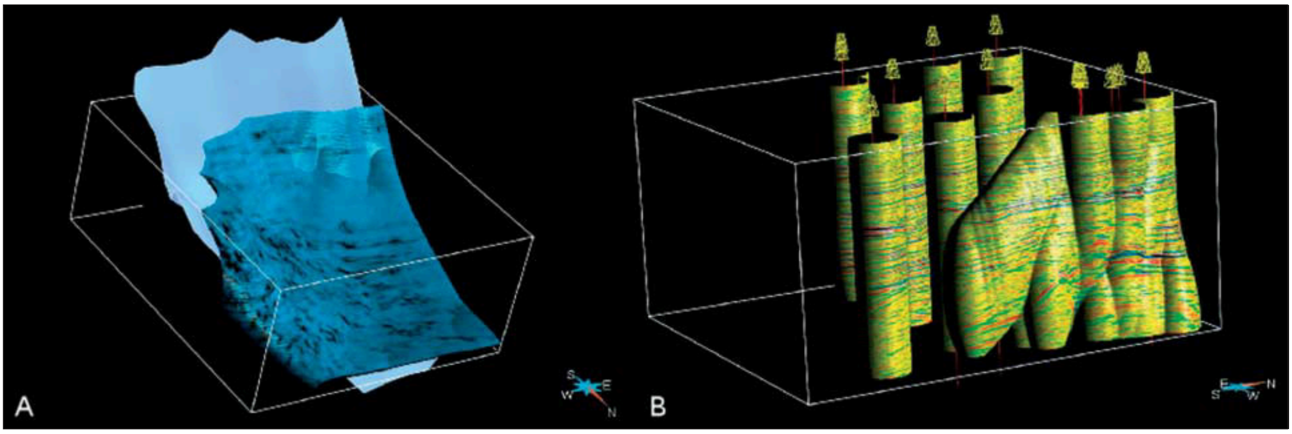


Figure 5 *Combination of structural information and seismic information with texture-based isosurfaces of distance maps painted with seismic data. (A) Isosurface of a distance map to a fault surface painted with semblance. (B) Isosurface of a distance map to a set of well paths painted with seismic amplitudes.*

a fault surface (Figure 5A), or seismic amplitude at a given distance to a set of well paths (Figure 5B).

Conclusions

Volume interpretation evolved out of visualization by looking for horizons and faults using opacity and colour with volume rendering. However, with new applications and hardware, volume interpretation is becoming an increasingly important tool for exploration. In this article, new, high quality volume rendering algorithms have been presented. Contrary to classical slicing-based volume rendering, they accurately extract meaningful information from the seismic data. They have been integrated in a generic multimodal volume rendering system to provide versatile visualization tools for seismic interpretation. For instance, we interactively render texture-based isosurfaces painted with seismic data. This makes it possible to combine structural information such as distance maps with seismic information such as seismic amplitude or semblance.

Acknowledgements

All images are courtesy of Earth Decision. The authors would like to thank Earth Decision for permission to publish this work.

References

Brown A.R. [2004] *Interpretation of Three-Dimensional Seismic Data*, Sixth Edition. AAPG-SEG, Tulsa.

Engel K., Kraus M., and Ertl T. [2001] High Quality Pre-Integrated Volume Rendering Using Hardware Accelerated Pixel Shading. *Proceedings of Eurographics/SIGGRAPH Workshop on Graphics Hardware*, 9-16.

Lum E.B., Wilson B., and Ma K.L. [2004] High-Quality Lighting and Efficient Pre-Integration for Volume Rendering. *Proceedings of Eurographics/IEEE Symposium on Visualization.*

Marsh A.J., Kidd G.D., and Furniss A. [2000] 3D Seismic Visualization Using Multiple Volume Data Sets. *Proceedings of the Indonesian Petroleum Association 27th Annual Meeting*, 599-612.

Weiskopf D., Engel K., and Ertl T. [2003] Interactive Clipping Techniques for Texture-Based Volume Visualization and Volume Shading. *IEEE TVCG*, **9**, 3, 298-312.