Geophysical Visualization: Industry Trends, Ingredients to Consider, and Future Expectations in building a Visualization Environment.

Tracy J. Stark, STARK Research, tstark3@attglobal.net

Abstract

Geophysical visualization is becoming an important and even required portion of a companyÕs arsenal of tools for finding and risking oil and gas accumulations. Its value and usage without a doubt will increase in the next 10 years.

Many companies now have at least one visualization center; a few have many. For example, bp built its first visualization center in early 1999. They now have 15 working centers worldwide, with 3 more either being built or planned. Rumor has it two other companies are playing Ôfollow the leaderÕ and plan on building 10+ centers in the next two years.

The goal of building and using a visualization center is to quickly and accurately gain insight on data without missing something important, and then being able to easily convey to others what you have learned. The ability to do this will depend upon your particular visualization center and the software you have running in it. Your ability to utilize these centers better than your competitors will depend upon: 1) your initial center design and implementation, 2) your plan for future modifications to the center and 3) how well and how quickly you adjust to the changes that will come.

Below I provide what I think are some industry trends concerning geophysical visualization and major ingredients that need to be considered when building such a center. I conclude with predictions of where we will be in 2011. Today it is hard to get a well drilled without interpreting a 3D data volume. I think the technology will have advanced such that in 2011 it will be hard to get a well drilled without a local collaborative visualization session of the drill location.

My presentation will also include a brief glimpse of software features I expect future visualization centers to utilize.

Lessons Learned and Industry Trends

Planning a visualization center should be done with the lessons learned and current industry trends in mind. Below are some of the lessons and trends that I think are important to consider. It will be an individual choice whether one should follow or buck these trends. A few companies will hopefully create new trends that others will follow.

<u>Level of Realism</u> As a Òlevel of realismÓ is added to a display, the easier it becomes to understand the data. Perspective, lighting, motion, and stereo are all on the upward path of improving the realism and improving our ability to understand complex 3D data. Stark et al. (2000, The Leading Edge, Vol. 19 no 8), provides examples of how adding a level of realism improves ones ability to understand complex data.

Today, head tracked stereo images provide the highest level of realism. Although commercially available versions of these programs are immature, they are developing quickly. A single, properly generated, static stereo image will provide an audience with much more useful information than either a movie loop of the same data around the same view point or even interactively moving the volume about that viewpoint. Stereo displays are equivalent to the queen in a game of chess - the most powerful piece in your arsenal. Moving stereo images are even stronger.

<u>Computer power</u> The changes in visualization have been strongly influenced by ÒMooreÕs LawÓ computer power will double about every 18 months. This ÒlawÓ implies a factor of 10 change in compute power every 5 years, and a factor of 100 in ten years. Staggering. The computers of today are 100 times faster than what was available in 1991 and should be 100 times slower than what will be available in 2011.

<u>Graphics power</u> — There probably exists a MooreÕs Law equivalent for graphics power. From my experience, on a certain Unix box, the graphics power tends to follow a square root of MooreÕs law and doubles about every three years. PC graphics power is growing much faster than on the Unix side, but it started at a lower level. High-end PC graphics are still behind the high end Unix graphics, but the gap is narrowing. If the gap closes — it will bring major changes to our industry.

A coming technology to watch is parallel graphics processing; doing with graphics what we have been doing with data processing. Splitting the problem up into many small pieces, solving them on cheaper boxes, and then combining the results, could provide super graphics performance at a significantly reduced price. Parallel graphics capabilities, combined with level of detail algorithms and better display culling algorithms, might give us better than

MooreÕs law improvement in functionality in the future.

Texture Memory Today most high-end graphics cards have texture memory. Texture memory first became available around late 1991. Today we can perform real time volume rendering of data that fits entirely in the texture memory. The largest amount of memory per graphic card is around 256 MB whereas in 1991 it was maybe 2 MB. Again we see a factor on 100 improvement in 10 years. Therefore in 2011 we might expect to have high-end machines with 25 gigs of texture memory.

Seismic data volumes Available data volumes have increased significantly in the past 10 years. Actual data collected in 2000 was not 100 time larger than what we collected in 1990. Given we have gone from single streamer to 8 or more streamer boats, a factor of 10 is a reasonable estimate. However, if you add in the number of new attribute volumes and our ability to begin to perform pre-stack interpretation, then the multiplier could easily be 100. If instrument oil fields become common place, then another 100-fold increase in 10 years is also a reasonable assumption.

Number of interpreters The world wide number of interpreters has probably decreased in the last 10 years. SEG membership grew 13% between 1990 (14,964 members) and 2000 (16,894). This growth however most likely reflects the growing internationalization of the society instead of new individuals joining the geophysical profession. This steady, or reduced state, will probably continue for the next ten years.

Software improvements New algorithms and techniques have emerged to change the way we look at and use seismic data. I expect this trend to continue in the future. New software will be key to enabling you to get the most out of a visualization environment. Your legacy software most likely was not designed to use these systems, and probably will not provide you with optimum performance.

To be sure, new software is springing up. New companies such as Continuum Resources, Inside Reality, and Magic Earth are providing us new ways and tools at looking at our data. There are also new companies which provide us new data volumes such as Coherence Technology (who probably started the attribute generation trend and is now part of Core Labs), Chroma Energy, dGB, eSeis, and Rock Solid Images to name a few.

Today instead of mapping many horizons, we generate many attribute volumes and then scan these volumes for large anomalies using volume visualization tools. Some companies even propose to never map events — just detect and drill geologic looking anomalies. It is interesting to note that even though computer speed has increased 100 fold in the last 10 years, the ability to map a significant structural horizon has only improved slightly. We are interpreting data differently in order to keep up with the rapid growth in the amount of data collected. We have developed new tools and techniques to handle the large data volumes instead of significantly improving our old techniques.

Number of Environments Today major companies are building visualization centers by the dozens. For example, bp built their first visualization center in early 1999. By June of 2001 they had 15 operational centers in 10 different cities around the world, with three more either planned or under construction. They also have provided their interpreters with 40 deskside sgi Onyx2 machines with MagicEarthÕs GeoProbe software for deskside visualization and interpretation work. In addition to their internal usage, they donated the ARCO visualization center (hardware, software and 3 years of funding) to the University of Colorado to help insure there will be new immersive technology in the future.

Bp is quite proud of their usage and implementation of visualization technology. Dave Roberts of bp in a recent email wrote: Ò. the technology has spread dramatically.....and the value we have already harvested from 'big screen teamwork' is staggering.Ó

Screen configurations Probably most of the environments built today contain front projected curved screens. These environments hide the image distortion when you are not in the sweet spot. You can move to the sweet spot to get an undistorted view. These environments provide immediate payoff but will only be able to partially utilize the fully immersive software applications that are becoming available.

Geophysical Visualization: Trends, Ingredients, and Future Expectations

Factors to Consider in Building a Visualization Environment

Overview

Based on the above trends, if you build a visualization environment today, I believe that in three years from now the following will be true. Will you be prepared for them?

1. Many of your competitors are using such environments to reduce their exploration and production risk.
2. The data volumes you need to handle per year have increased by a factor of 4.
3. The are fewer available interpreters.
4. New software is available that promises great advancement, but most is from new companies and therefore will not work seamlessly with your existing software.
5. There will be a selection of head tracked software applications available.
6. The new computers your competitors are buying are 4 times faster than the ones you got with your system.
7. You need consultants that live in a different state or country.
8. Some of your partners want remote collaboration sessions.

Major Visualization Center Ingredients

Intended room usage, available space, and budgets are the over-riding considerations in the building of a visualization center. With existing technology, available space can rule out some of the options such as fully immersive head tracked environments, the same can be said for budget. On the other hand, if the intended room usage is for fully immersive head tracking, the required space will be found and/or built and the budget will be made available to meet the need.

In additon to the above considerations, there are other ÒingredientsÓ that need to be considered. Below I provide brief discussions of the ones I think are most important.

Intended Room usage How you plan on using your visualization room is the first and most important decision to make. Both a primary and secondary usage should have heavy consideration in the design phase. A center that is mainly for show and tell sessions with clients, investors and board members will be different than one that is mainly for individual or small team sessions. Desired room usage will limit some of your choices on the options. For example, a center designed for fully immersive head-tracked applications can only use rear projected flat screens.

As part of your intended room usage, you need to decide on the amount and kind of immersion you want. Do you want to move images with a mouse on a large screen or are you interested in full immersive with head-tracking. This dictates screen design, room size, projector type, computer hardware and available software. Do you want to be able to work in a fully lit room, or does one with dimmed lights meet your needs? Working with the lights on or off will dictate what type of projectors you purchase, and can have ramifications for screen design and your ability to do stereo within your budget.

Audience and Room Size What is the maximum expected audience size? How many will be active users versus passive observers? How might this mix change? Is the room large enough to handle both your preferred screen design and this many people? Is the ceiling high enough? Are the other facilities such as power, heating, cooling, and restrooms adequate for the load? How are the lights controlled? Is there enough light to see or take notes but not interfere with the screens? What about air currents? Are they going to cause your screen to move every time the air conditioning comes on, or someone opens the door? Will opening the door disturb the audience?

Available Budget The available money will be proportional to your managementÕs belief in the ability of visualization to both reduce risk and provide new opportunities.

Software capabilities Are you going to use software designed for the desktop or software specifically designed to work in a visualization environment? How much of an improvement from a new software package will you demand before you are willing to change? A factor of 2? 4? More? What are your lost opportunity costs for waiting to switch to the new software? What will the early adopters gain over their competitors?

Stereo image displays Building a visualization center without including the ability to display stereo is like starting a chess game without your queen. It is very hard to compete and win that way. Using a head tracked fully immersive center will be like playing chess with many queens — almost impossible to beat! (Not quite the case today because of the immaturity of the software, but should be in the near future.)

Geophysical Visualization: Trends, Ingredients, and Future Expectations

Computer Hardware This could be your greatest cost. The computer hardware you select should be based on very current information and should take into account your desired room usage, software applications, volume and surface sizes today, and expected in three years. Knowing that the machines are going to improve, do you start with the high end today, and upgrade slowly, or do you start at the mid-grade and upgrade often?

Screen Configuration There are many different types of screen configuration: flat, single curved, multi-curved, moveable, etc. You should choose the one that best fits the intended use of the environment and your planned future expansions. Your screen configuration will determine the type of image distortion you have. Some people will make this decision based on the type of image distortion they prefer. There are advantages and disadvantages to each of the screen configurations.

Projector Location Placement of the projectors determines how close you can get to the screen without casting a shadow. If space is an issue, you will have to place the projectors in front of the screen. This will limit your ability to use head tracked truly immersive technology.

Type of Projectors — CRT, LCD, DLT — again, what are you going to use the room for? How important is: stereo? cost? working with the lights on? A lights on environment mean using either LCD or DLT projectors, and significantly increases the cost of stereo. Currently all fully immersive environments that I know of use CRT projectors and require working in a darken room.

Head tracking capabilities — A requirement for full immersion, however most standard software packages do not support it. You should expect to use it in the future. Some tracking systems can dictate the type of building material you use (non-ferrous). Continuum Resources, Gocad, and Inside Reality are the only three companies that I know of who support head tracking today.

Self versus contractor Do you do everything yourself, or hire a company to build a system for you? My recommendation is to use a known player that has experience in the oil and gas industry. The few I am familiar with include: Fakespace, MechDyne, Panoram, sgi, TAN, and Trimension. I have my favorite and least favorite, but that is something I had best not put in print.

Predictions for 2011

Based on the above trends, I expect the following to be ÒsafeÓpredictions for 2011.

- It will be hard to get a well drilled without a local collaborative visualization session of the drill location.
- Some companies will routinely use their centers for remote collaboration since there are visualization centers ÒeverywhereÓand the internet has taken its next major step.
- New software and interpretation methods will be available that radically change the way we use and interpret 3D seismic data by taking advantage of these new centers. They will result in significant improvement in our ability to risk-manage drilling portfolios. Many of these step change improvements will come from very young companies.
- Computers will be 100 times better than they are today and texture memory will be measured in gigabytes instead of megabytes. Computers still will not be fast enough to handle all of our needs, much less our desires.
- Data volumes will continue to grow faster than we can handle them.
- A second growth wave of truly immersive environments will be starting.
- In the past, a seismic computer was a human doing calculations; today it is a machine. Today, a seismic interpreter is a human following seismic events and picking well locations using a computer; in 2011 a seismic interpreter will beÉVery different.

Bottom Line

In the past 10 years interpreters have seen changes in the way they work and interpret 3D data. I expect the next 10 years will produce significantly more changes. A trend has started, and I believe it is now accelerating. We are at the beginning of a visualization center building boom. New software is going to come along that will allow the explorationist to take better advantage of these new centers, the vast amount of data that is being collected, and the available computer systems. Hopefully I will have something to add to this mix of new technology that will help change the way we view and use 3D seismic data.

Seismic Data Segmentation via Hyperstack Region Growing

Marcos Machado[(1)], André Gerhardt[(1)] and Marcelo Gattass[(2)]

[1] PETROBRAS S.A., Rio de Janeiro, Brazil (marcoscm, agerhardt @petrobras.com.br)
[2] Tecgraf / PUC-Rio, Rio de Janeiro, Brazil (gattass@tecgraf.puc-rio.br)

Abstract

This work presents a new multiscale segmentation method for multidimensional images, called Hyperstack Region Growing. It incorporates Region Growing strategies into the Hyperstack algorithm to increase user control over the segmentation process in complex images. This paper also presents examples based on 2D and 3D seismic data, illustrating the application of the proposed method on the segmentation of elongated structures.

Introduction

There is a great variety of data segmentation methods. In many of them, the data smoothing issue is of paramount importance to reduce noise effects and work with different levels of detail. Segmentation algorithms that deal with smoothing and different scales are called *hierarchical* or *multiscale* [1]. In these algorithms, all computations are performed at multiple resolution levels. The goal is to assure that, for each part of the image, computations are executed at the appropriate resolution level.

A hierarchical algorithm based on the Scale Space theory [2] is Hyperstack [3]. It seeks to automatically identify all homogeneous regions of an image. User intervention is restricted to defining global and generic initial parameters.

There are, however, many situations in which the user is interested in viewing a predefined region of an image and has little interest in other areas. Moreover, selecting the Hyperstack parameters is not an intuitive task. A given set of parameters may not be adequate to the specific image or to isolate a particular segment in which the user is interested. Elongated structures, which traverse large portions of an image, may also present potential problems. The Hyperstack algorithm might treat all parts of such structures as one segment in a high scale, in which leaking may occur and the segmentation process may include spurious regions. Proper identification of elongated structures is essential for seismic studies.

This paper presents a variation of the Hyperstack algorithm, here called Hyperstack Region Growing [4], which addresses all of these questions. It seeks to identify a single segment of the image from one or more seed voxels provided by the user. It can be generally described as an application of the Region Growing [5] strategy to the data structure generated by Hyperstack.

Hyperstack

This section presents a brief description of the Hyperstack segmentation algorithm. More details can be found in [3].

Hyperstack (Hnp) is a segmentation algorithm that uses Scale Space, taking advantage of the global information present at higher scale levels. The basic idea is to build links between voxels at adjacent levels in Scale Space, followed by a root selection to find the voxels that represent segments in the original image at all scales (Figure 1). The data structure representing the links is a forest where each tree represents a segment.

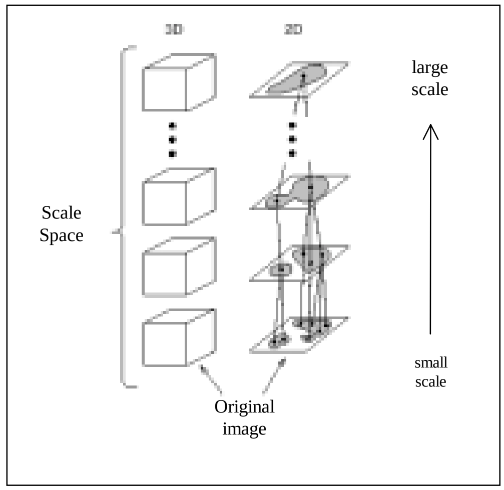


Figure 1: Hyperstack Scheme in 3D and 2D [3].

The method consists of four consecutive steps: (i) **blurring**, which builds the image's Scale Space; (ii) **linking**, which establishes the connections between voxels at adjacent scale levels; (iii) **root labeling**, which defines which voxels in Scale Space represent an entire segment; and (iv) **downward projection**, which forms the segments in the original image. The blurring step has the effect of a low-pass filter: low frequencies are preserved, while higher frequencies (the details) are smoothed out. The linking step is based on heuristic criteria called affection (or attractiveness), and the voxels involved in this hierarchical relationship are called child and parent.

The affection criteria is based not only on the proximity of voxel values but also on the volume of other children already linked to a candidate parent (convergence is encouraged to ever fewer parents) and the mean value of such children. Voxels that do not possess any children are considered inactive and are ignored in the continued connection process.

The root-labeling step is based on an “adultness” criterion that measures the lack of affection between a voxel and all its possible parents. A minimum adultness is needed for a child to be labeled as root and to form a new segment by means of the downward projection step.

Probabilistic Hyperstack (Hp) is a variation of Hyperstack with a more tolerant linking step: a child can be connected to more than one parent – that is, the linking step does not define the parent, but rather indicates the most probable parents. The goal here is to minimize the risk of losing important connections due to noise. With such connections preserved and the noise being progressively reduced, it is expected that compound probabilities will indicate the right connections as the algorithm progresses to higher levels.

The probability of a relationship can be obtained by normalizing all acceptable affection values of a child so that their sum is one. These relationships, treated as arcs, yield a graph that is the general case of the forest in Hnp. The probability for a voxel in the original image to belong to a given segment (root probability) can be computed by composing the adjacent-level probabilities from the original image to the root that represents this segment. The root-probability values can be used to define each voxel in a single segment that corresponds to the maximum probability, or to interpret the result as if the voxel were composed of sub-voxels (partial volume).

Hyperstack Region Growing

The variation presented in this paper, called Hyperstack Region Growing, aims at identifying a single segment in the original image. In this method, the user identifies one or more seed voxels in the original image and an upper limit scale. The seed choice can be aided by some automatic method, such as thresholding. The upper limit controls the segment’s definition, as described below.

There are also two versions of the Hyperstack Region Growing method: a non-probabilistic (HRGnp) and a probabilistic (HRGp) version. They use Hyperstack’s forest and graph, respectively. The basic idea of our method is to follow either the forest (HRGnp) or the graph (HRGp), identifying which voxels belong to the same segment as the seed voxels.

In HRGnp, the strategy is first to find the ancestors to the seed voxels in the upper limit level defined by the user. These ancestors are treated as pseudo-roots of the desired segment. The voxels of the segment in the original image are then obtained by performing the downward projection of these pseudo-roots. One could say that this strategy is similar to the Region Growing algorithm with the neighborhood being defined in the Scale Space.

As the upper level is increased, so is the number of voxels in the segment. That is, the voxels in the previous level are preserved and new voxels may enter the segment. Note that the algorithm can be implemented with the upper level being defined interactively by the user, who can initially provide a lower value and increase this limit as the desired response is produced.

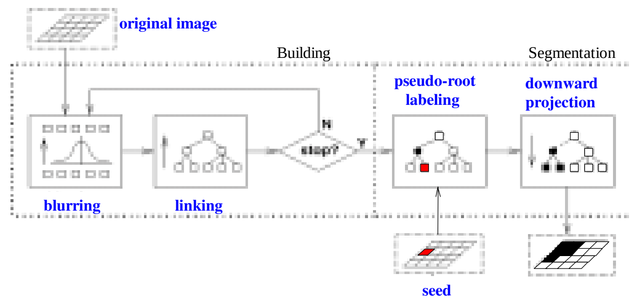


Figure 2: Schematic of HRGnp segmentation process.

In the probabilistic version of our method, the strategy is very similar to the one presented above. The main difference is that more pseudo-roots are found, as both the pseudo-root labeling and the downward projection steps follow the deterministic-graph data structure.

HRGp presents two different behaviors as the upper level is increased. If the projection is performed using the maximum probability, the behavior is similar to the one in HRGnp. If the partial-volume criterion is used, voxels are preserved as the upper scale level is increased, but their probability of belonging to the segment may change.

The algorithm consists of the following steps (Figure 2):

1. **Blurring**: identical to traditional Hyperstack;
2. **Linking**: also identical to traditional Hyperstack, can be simple (Hnp) or probabilistic (Hp);
3. **Seed identification**: labeling, by the user, of one or more seed voxels;
4. **Pseudo-root labeling**: in each scale level, all voxels reached by any path from the seed voxels are considered as belonging to the same segment (as if they were children of the same root);
5. **Downward projection**: identical to traditional Hyperstack;
6. **Upper level selection**: the user identifies an upper scale level in which the generated segment is considered appropriate.

Steps 3-6 can be repeated changing the seeds or the upper level until a satisfactory result is obtained.

Note that HRGnp is not sensitive to replacing the initial seeds by other voxels that belong to the same region. In HRGp, this happens only among voxels with a 100% probability of belonging to the segment.

An important variation of the proposed algorithm is to restrict the inclusion of a voxel in a segment to voxels connected to the seeds. In this case, the projection considers for inclusion into a segment, among all obtained voxels, only those that present a path to a seed. This condition is here called **connectivity condition**.

Results

A - 2D Seismic Image

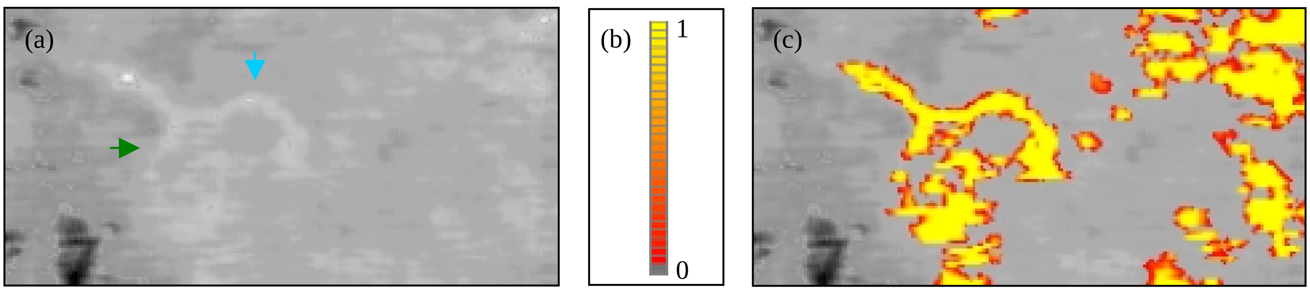


Figure 3: (a) A time-slice with a channel divided in 2 branches (blue and green arrows); (b) probability color scale; (c) Hp segementation, level 14.

The image shown in Figure 3(a) corresponds to the horizontal section of a real 3D seismic data. On the northwestern quadrant it presents a complex elongated structure representing a buried channel of an old fluvial system. The channel is divided into 2 branches: the upper branch (indicated by the blue arrow in Figure 3(a))

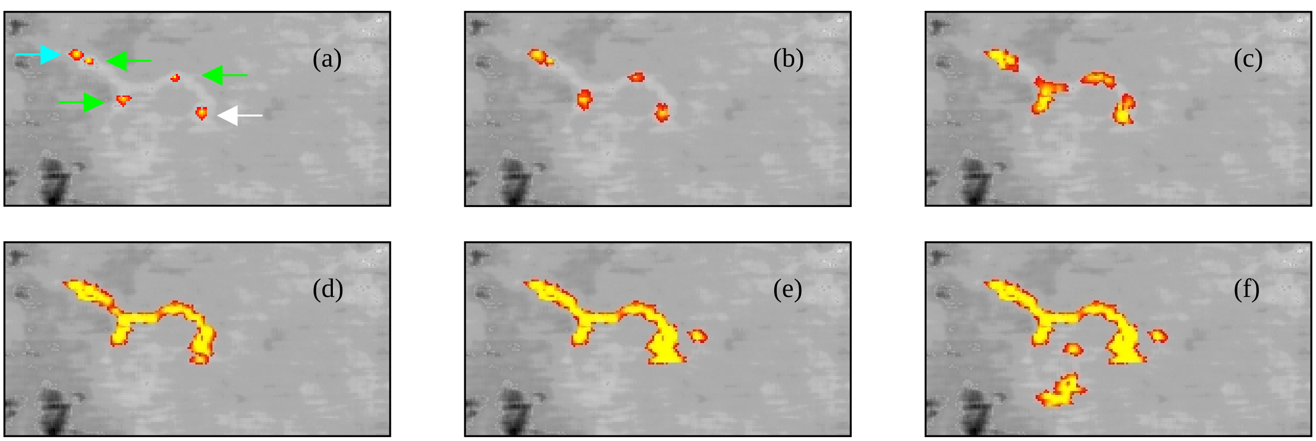


Figure 4: HRGp segmentation from 5 seeds on levels 2 (a), 3 (b), 6 (c), 8 (d), 9 (e) e 10 (f).

stretches towards the center of the image; the lower one (green arrow) does not present a well-defined continuity on the image plane.

Figure 3(c) shows the channel's segmentation with Hp. The result was obtained at level 14. Its probability of belonging to the segment is presented according to a color scale (Figure 3(b)). Pixels with a 0% probability are shown with the original image's gray tones. Similar results were obtained with Hnp.

Figure 4 shows how we can explore the option of providing more than one seed for HRGp. Five seeds were provided (see (a)), three of them related to the higher-intensity points in the upper part of the channel and its two branches (green arrows). A fourth seed was included to cover the extremity of the upper branch (white arrow), and another one to better include the central portion of the channel (blue arrow). This last seed was needed because the region growth around the higher-intensity point was blocked along several scales. Note that the channel is segmented on level 8 or 9 (depending on the user's strictness), which is quite sooner than level 14, as was obtained with only one seed or by labeling Hp's roots. Furthermore, even using HRGp without a connectivity test with the seeds, the generated segmentation has none (level 8) or little (level 9) leaking.

B - 3D Seismic Image

To evaluate this algorithm in three dimensions, a complete seismic dataset was used which includes the 2D image from the previous section. The same channel was segmented, now in 3D, using HRGp with connectivity conditions. Figure 5 shows the segment obtained in level 14, using only one seed. Figure 6 presents the segmentation results starting with 6 seeds, identified in level 12. Note the smaller amount of leaking in relation to the previous figure. Figures 5 and 6 have been trimmed to better show the inner channel details.

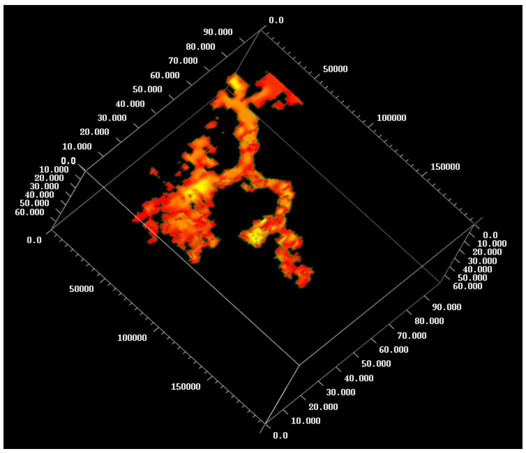


Figure 5: 3D channel segmented by HRGp with 1 seed.

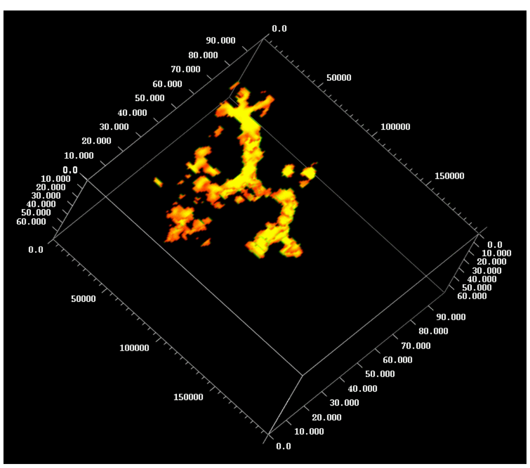


Figure 6: 3D channel segmented by HRGp with 6 seeds.

Conclusions

Our modification to the Hyperstack algorithm has yielded a semi-automatic segmentation method with some advantages:

- There is no need for an heuristic to find the root voxels;
- By observing the images generated in each scale during the evolution of the algorithm, the user can choose an appropriate scale;
- More than one seed can be provided, therefore complex elongated structures can be recognized at lower scale levels, thus decreasing the possibility of leaking.

In the case of the seismic images tested, both in 2D and 3D, we obtained a segmentation of the channel in a scale level prior to the one identified by traditional Hyperstack, thus reducing the resulting leaking. This shows how useful this variation can be, especially in the case of elongated structures. Notice that the segmentation of the same channel using the conventional Region Growing technique is effortful, demanding several tests with different statistical criteria on the voxels until a good result is obtained.

Acknowledgments

We would like to thank PETROBRAS for the permission to publish this work and for supporting the thesis of the first author. Jonas Miranda Gomes co-advised the first author and provided countless excellent suggestions.

References

[1] *P.J. Burt, T.H. Hong and A. Rosenfeld, "Segmentation and Estimation of Image Region Properties Through Cooperative Hierarchical Computation", IEEE Transactions on Systems, Man and Cybernetics, 11(12), pp. 802-809, 1981.*

[2] *A.P. Witkin, "Scale Space Filtering", Proc. International Joint Conference on Artificial Intelligence, Karlsruhe, W. Germany, 1983.*

[3] *K. Vincken, "Probabilistic Multiscale Image Segmentation by Hyperstack", Ph.D. Thesis, Utrecht University, The Netherlands, 1995.*

[4] *M. Machado, "Seismic Data Segmentation by Hyperstack for Visualization", M.Sc. dissertation, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ, 2000 (in portuguese).*

[5] *R. Adams and L. Bischof, "Seeded Region Growing", IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(6), pp. 641-647, 1994.*

Two-Dimensional Opacity Functions for Improved Volume Rendering of Seismic Data

André Gerhardt (1), Marcos Machado (1), Pedro Mário Silva (2), and Marcelo Gattass (2)

[1] PETROBRAS S.A., Rio de Janeiro, Brazil (agerhardt, marcoscm@petrobras.com.br)
[2] Tecgraf / PUC-Rio, Rio de Janeiro, Brazil (pmario, gattass@tecgraf.puc-rio.br)

Abstract

Although Volume Rendering has gained reputation as a powerful tool for visualizing complex structural and stratigraphic features embedded in 3-D seismic data, the complexity of the parameters involved still requires a lot of work in order to produce an informative image. One of such parameters assigns opacity levels to data-amplitude values to highlight the features of interest. The design of the opacity function usually follows a non-intuitive trial-an-error approach. The strong dependence of the rendered image on other optical parameters also contributes to make it a time-consuming task. Furthermore, some particularities of seismic data contribute to make such approach more suitable for the visualization of high-amplitude anomalies. This work describes a method for generating two-valued (amplitude and gradient) opacity functions that is able to better discriminate mid-amplitude events without obscuring them with other seismic features. Preliminary results using synthetic data are presented. Examples using real datasets will be presented at the conference.

Introduction

Direct Volume Rendering (DVR) has become a widely used tool for visualization and interpretation of 3-D seismic data. Its ability to help interpreters grasp the whole data structure in one single image, when compared to standard (2-D) interpretation techniques, has attracted industry attention.

DVR techniques rely on the definition of *transfer functions* to highlight specific features of interest embedded in the volumetric dataset by mapping amplitude values to optical properties – usually opacity, color and shading. The present work tackles the problem of setting the subset comprised by the *opacity functions* only.

One of DVR's greatest limitations is that opacity functions do not take geometry into account. During the rendering process, the data are assigned to opacity values irrespective of their spatial distribution and coherence across the whole volume, therefore lacking the power to isolate geological features of interest. Furthermore, the oscillating nature of seismic data results in overlapping ranges of data values, which makes it impossible to separate events other than the ones with the largest absolute amplitudes. These characteristics contribute to make seismic data perhaps one of the most challenging and time-consuming targets for DVR technology (Gerhardt, 1998).

The design of opacity functions is the most demanding task for obtaining an informative rendering from most types of data. Medical visualization benefits from some particularities of the data, specially the knowledge of a priori models, which allows the use of pre-designed standard transfer functions in some cases. However, this is not possible in the case of seismic data due to the strong variability of the data. Analyzing amplitude values using sample slices throughout the dataset can help defining relative opacity levels, but there is no easy way to predict how these individual samples stack up three-dimensionally and contribute to the final rendered image.

Typical user interfaces are restricted to editing a graph of the opacity function based on the histogram of the dataset values. Unfortunately, this conveys little useful information, as the histogram is also an entity that lacks spatial information. Thus, finding a useful opacity function is usually a non-intuitive, labor-intensive trial-and-error task. Furthermore, small changes in the opacity function can lead to great changes in the final rendered image, which combined to other optical parameters – specially the viewpoint – adds considerably to the complexity of the process. These difficulties can be arguably some of the reasons why volumetric interpretation has not achieved an even larger level of acceptance and usage amongst the geophysical community.

Related Works

Most of the current research and work on DVR focuses on making the rendering algorithms faster. However, very little effort has been made to improve the heart of the technology (i.e., the opacity function) aiming at obtaining more correct and informative renderings in a more intuitive and appropriate way.

Levoy (1988) was the first to introduce opacity functions with two variables (data value and gradient). The method is able to render only a single boundary at a time, and requires a lot of parameter experimentation from the user.

Gerhardt et al. (1999) developed an approach that incorporates a region-growing algorithm type into the DVR pipeline in order to add geometry information to the method. Although simple and efficient for isolating events, this approach still relies on 1-D opacity functions and therefore suffers from their limitations to correctly handle seismic data.

Kindlmann (1999) developed a method for semi-automatic generation of 1-D and 2-D opacity functions. Using algebraic properties of the Gaussian function, the user defines only a weighting window function that describes the behavior of all boundaries present in the data. Though remarkably simple, the approach assumes a data model incompatible with the oscillatory nature of seismic data.

Reflection Model

The current work adapts Kindlmann's model (1999) to accommodate the particularities of seismic data. The goal here is to automatically generate an approximate opacity function, which is able to render all seismic events present in the dataset and can serve as a basis for further editions by the user.

This work assumes as a model that a seismic trace (a vertical sequence of amplitude values) can be represented by a series of Gaussian functions, of either positive or negative values, defined by:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}},$$

with σ being the standard deviation and *x* the relative position to the center of the Gaussian. The first derivative is given by:

$$f'(x) = -\frac{x}{\sqrt{2\pi}\sigma^3} e^{-\frac{x^2}{2\sigma^2}}.$$

The Gaussian function has inflection points at ±σ, where *f '(x)* reaches its extrema, which can be regarded as the "thickness" (2σ) of the corresponding seismic event. The value of σ can be estimated from the values of *f* and *f'*:

$$\frac{f(0)}{f'(-\sigma)} = \sigma\sqrt{e}.$$

Once σ has been determined, the relative position *x* can be recovered based only on the values of *f* and *f'*:

$$\frac{f'(x)}{f(x)} = -\frac{x}{\sigma^2}.$$

When using real data, σ can be estimated from a histogram of *f versus f'* (discussed next). The trimmed weighted mean first-derivative function, *G(v)*, of *f* over all *x* positions in which *f(x)= v*, can be recovered from that histogram and used to obtain σ according to:

$$\sigma = \frac{\max(v)}{\max_v(|G(v)|)\sqrt{e}}.$$

With this information it is possible to define a mapping from data value, which is an approximate position along the event, as follows:

$$p(v,g) = \frac{-\sigma^2 G(v)}{v} \approx x.$$

The opacity value can then be obtained with a window function, *b(v)*, which controls thickness, sharpness and proximity to the maxima of the rendered events:

$$\alpha(v,g) = b(p(v,g)).$$

This opacity function renders all events present in the dataset. The user can eliminate non-interesting events from the final image by zeroing out portions of it.

The Histogram Panel (HP)

The relationship between one trace of the data and its gradient can be analyzed in a 3-D graph as a function of position (Figure 1).

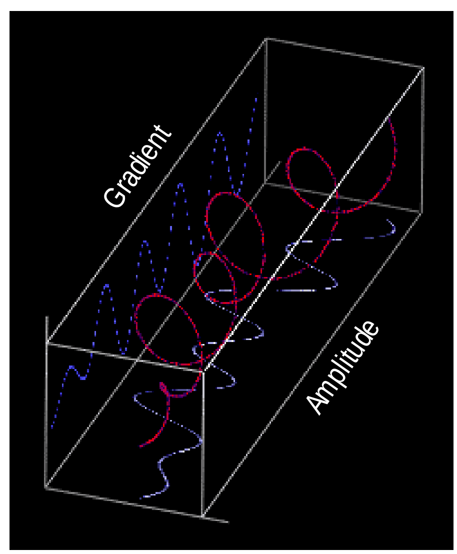


Figure 1: Amplitude versus gradient relationship.

One can observe that the gradient can be regarded as a rotated version of the data. Each individual seismic event tends to concentrate on limited combinations of amplitude and gradient pairs. As both *f* and *f'* are functions of position, they can be projected along the position axis and plotted as a 2-D image. As more events are considered down the trace, and quantization is necessary to create an image, some pixels in this image may be assigned more occurrences, creating a 2-D histogram (the Histogram Panel) that depicts the structure of the data events irrespective of their spatial positions (Figure 2).

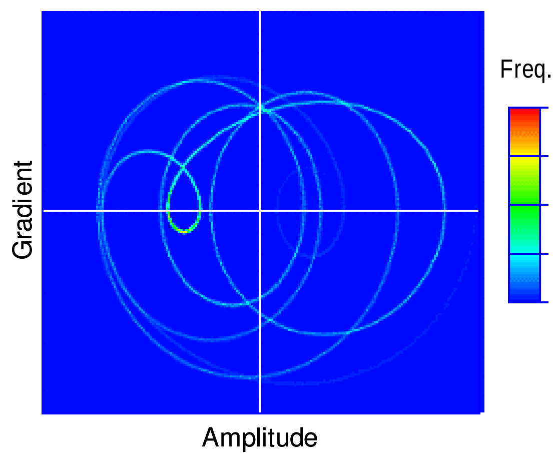


Figure 2: The Histogram Panel (HP).

In this representation, absolute peak values are concentrated close to the amplitude axis. The relative position (left or right) to a vertical symmetry plane of each individual event in this space (mid-point between two successive amplitude axis crossings) provides information on whether the point is in a peak or a trough.

The approach taken in this work is to measure *f* and *f'* more than once per voxel, at interpolated sample points of the dataset, since amplitude/gradient pairs taken from laterally monotonous geological areas may tend to accumulate in a few pixels of the HP, failing to reveal the oscillatory structure of the data. Furthermore, the gradient information is calculated only along the vertical axis of the dataset, therefore keeping information about its sign and contributing to make the HP less cluttered.

Design of the 2-D Opacity Function

The HP provides a basis for defining 2-D opacity functions and further refinements made by the user. Based on the reflection model, $\alpha(v,g)$ can be mapped onto HP as a preliminary semi-automatic opacity function. A region-selection tool can then be used to select different regions of the HP in order to render only the events of interest (Figure 3). As there is a potential inaccuracy in the estimation of σ, the user may need to play with different apertures of the region-selection tool for different events.

Although it is more complex, this data representation allows the user to selectively render only the events of interest. Considering a narrow amplitude range, selecting only the very low gradient values results in having only a single event rendered, ignoring other events with higher absolute values which contain those amplitudes as components.

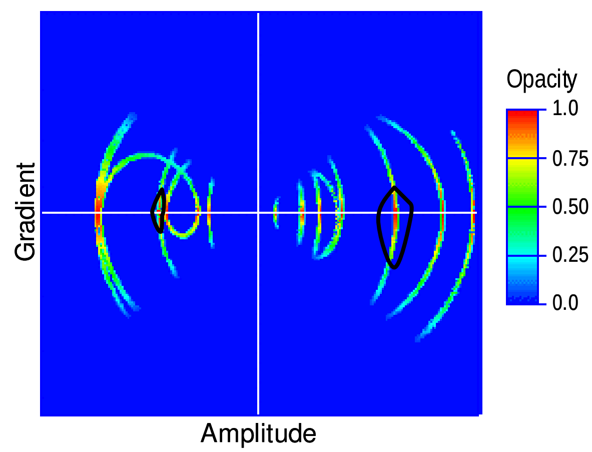


Figure 3: 2-D opacity function interface. The image is zoomed on the region close to the amplitude axis. The black curves indicate the selections made by the user.

Results

A synthetic model is used to demonstrate the advantages of the technique described in this work. The model consists of a real trace spatially interpolated to create an anticline structure. Figure 4a shows the kind of results obtained when using the traditional approach of 1-D opacity functions. The events of interest (in this case the intermediate absolute amplitude values) could not be completely isolated. Figure 4b shows the result of using an edited 2-D opacity function. Both events (with positive and negative values) are correctly isolated.

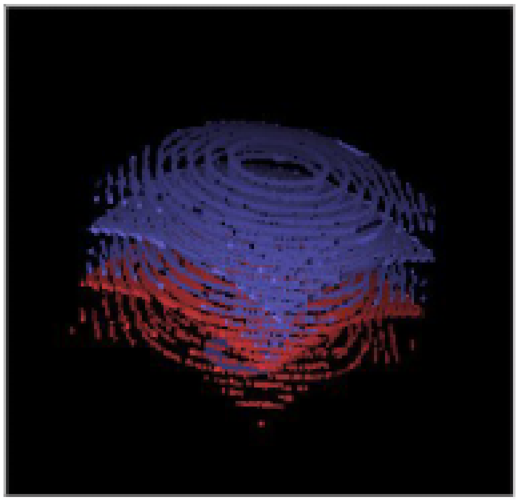


Figure 4a: Synthetic model rendered using a 1-D opacity function.

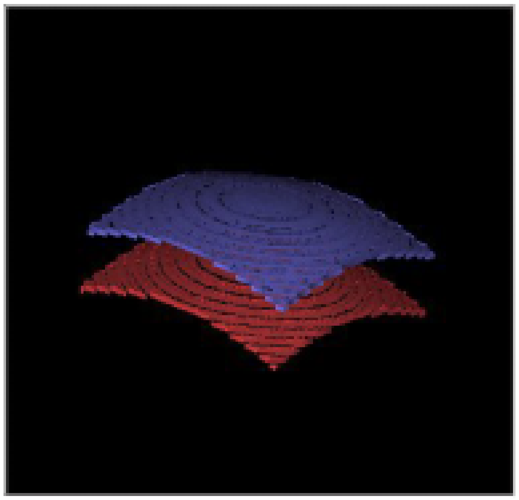


Figure 4b: Synthetic model rendered using our approach.

Conclusions

This work presents a new approach to the design of opacity functions for the Direct Volume Rendering of seismic datasets. This approach is able to distinguish events that have overlapping ranges of values. The extension of the traditional opacity function definition interface to 2-D, incorporating gradient information, allows the user to select only the events of his/her interest for rendering. The use of a reflection model helps the user to create more homogeneous renderings and can be regarded as a semi-automatic opacity function generator.

References

Gerhardt, A.; 1998. Aspects of Seismic Data Volume Rendering. M.Sc. Thesis, PUC-Rio, Rio de Janeiro, RJ. (In Portuguese).

Gerhardt, A.; Machado, M.; and Gattass, M.; 1999. A Combined Approach to 3D Seismic Data Segmentation and Rendering. 6th International Congress of the Brazilian Geophysical Society, Rio de Janeiro, Brazil.

Kindlmann, G.; 1999. Semi-Automatic Generation of Transfer Functions for Direct Volume Rendering. M.Sc. Thesis, Cornell University, Ithaca, NY.

Levoy, M.; 1988. Display of Surfaces from Volume Data. IEEE Computer Graphics & Applications 8(5):29-37.

Acknowledgments

We would like to thank PETROBRAS for permission to publish this work.

Voxel Visualization: Putting the 3-D into 3-D Interpretation

John Prutzman, VoxelVision, LLC.

Summary

Voxel visualization permits 3-D volume rendering and display, rather than the more typical 3-D surface rendering and display. This permits viewing structure prior to interpretation, applying true 3-D tracking, calculating volumes without horizons, and planning wellbore tracks within the voxel volume. Other more novel uses are also possible.

Introduction

The last twenty-five years have witnessed at least two major changes in subsurface seismic interpretation. The first shift was from 2-D to 3-D data in an attempt to better image the subsurface. The second change was away from paper sections to computer workstations for its ease and speed of interpretation. This latter change was a response to the former change aided by the increasing power and declining costs of computer hardware.

Early software programs implemented the familiar 2-D interpretation process allowing the interpreter to build up a 3-D subsurface model. Newer software continued the 2-D interpretation process while adding 3-D tools such as horizon and fault tracking along with 3-D visualization of surfaces, faults, wells, etc. Today, interpreters still like to think in terms of 2-D sections when studying 3-D problems. In part this is because of the limitation of interpretation and visualization software. Voxel visualization software holds out the promise of moving the interpretation process out of the 2-D realm into the 3-D realm.

Voxel Technology

Geoscientists are familiar with the concept of the pixel, which stands for picture element. They understand that the number and size of pixels on a display screen determine the resolution of displayed data. Smaller and more numerous pixels give higher quality displays.

When displaying an interpreted horizon, data points on the horizon surface are mapped via a perspective or orthographic view to a flat surface that can be displayed by pixels. This transformation from a 3-D surface to a 2-D surface for display is called 3-D surface rendering. The software need only consider a small subset of the total data volume when rendering since only 3-D surface data is displayed. Internal data within the volume cannot be seen. Thus, 3-D surface rendering is relatively fast and the resulting 2-D surface lends itself to display as a series of connected polygons, Figure 1.

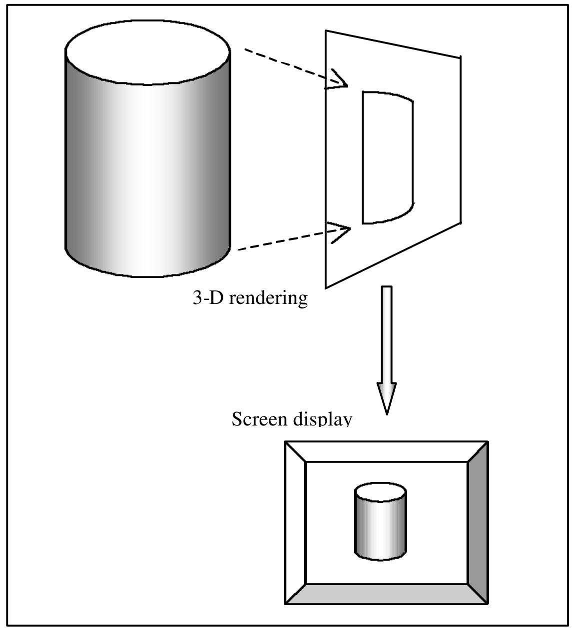


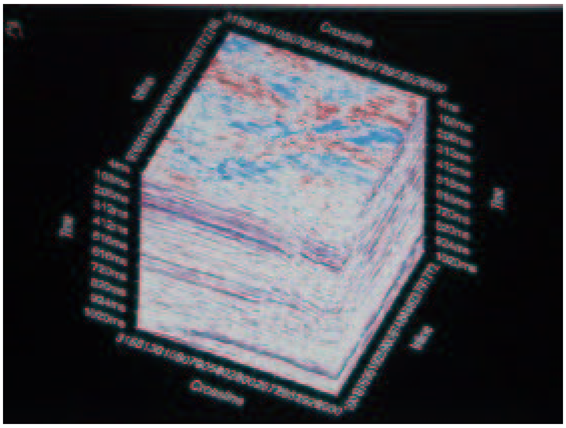
Figure 1: 3-D surface rendering and polygon display.

Less familiar is the concept of the voxel, which stands for volume element. Every data point is considered to occupy a small cube of space whose physical dimensions correspond to the bin size. Since all data cubes are available for display and multiple data cubes can contribute to a single pixel on the screen display, it makes sense to refer to these small cubes of data as volume elements, or voxels.

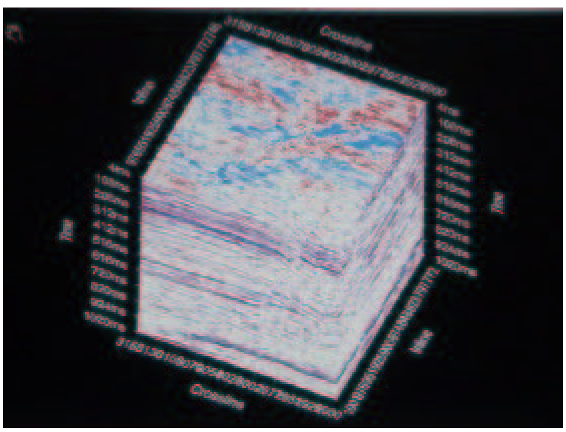
Multiple voxels can contribute to a single pixel on the screen display because each voxel is assigned an opacity as well as color and location. Voxels can be opaque, translucent, or transparent. When all the voxels are assigned total opacity, then the resulting display appears identical to 3-D surface rendering. When opacity is less than 100%, the 3-D volume rendering clearly shows a volume of data is present, not just a surface, Figure 2.

Voxel Visualization

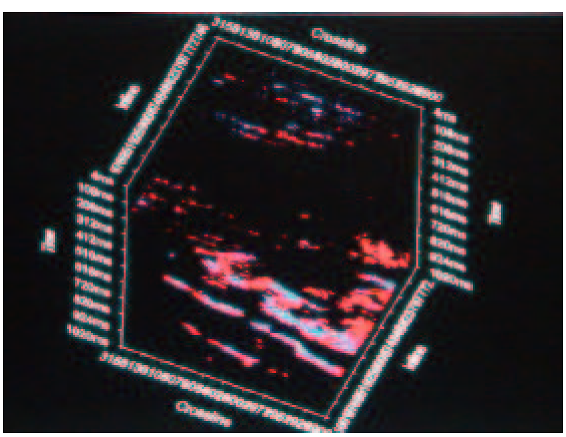
The presence of a volume of data available as voxels suggests that important structures can be displayed in 3-D directly without any interpretation. This is impossible with



3-D surface rendering



3-D volume rendering ,opaque



3-D volume rendering, transparent

Figure 2: Surface rendering uses a subset of the data. Volume rendering uses all the data.

traditional visualization software that can only display 2-D data in a movie type display. The 3-D structure can only be displayed after one or more horizons are interpreted. The ability to directly see 3-D structure with voxels prior to interpretation helps drive the interpretation process to arrive more quickly at the correct interpretation.

Voxel visualization can be applied to data other than seismic amplitude. Attributes, including AVO attributes can be displayed also without interpretation in 3-D voxel space. For example, a far offset cube minus a near offset cube results in a difference cube that can be used to highlight interesting anomalies quickly. This can lead the interpreter to focus on specific places within the dataset that appear to have some potential as prospects.

Voxel rendering allows the interpreter to look at large volumes of data, isolating significant anomalies quickly prior to interpretation. This is a significant advantage to the traditional approach of displaying 2-D sections out of a 3-D volume.

Voxel visualization also permits true 3-D tracking. Traditional horizon tracking starts at seed points and proceeds from trace to trace in a 2-D fashion. This mimics how an interpreter tracks a horizon on paper sections. While this approach is faster than using paper sections, it fails to exploit the three-dimensional nature of 3-D data. It is a relatively slow and inaccurate way to track. By contrast, tracking on voxels is much quicker and more accurate since the tracker examines data in a true 3-D perspective. Upon first seeing voxel tracking, interpreters may assume the display shows a previously completed track because of the tracking speed.

Voxel tracking can also be applied to volume tracking since voxels have volume. Multiple attributes can be applied to defining a volume to be tracked. The resulting tracking identifies similar, connected voxels meeting the specified criteria. Volume tracking is very helpful in identify stratigraphically important units.

Voxel visualization can also be used to directly calculate volumes of isolated zones. This can be done without the necessity of interpreting two or more horizons to define the volume limits. Instead, opacity values are set so the zone of interest appears within the rendered volume or previously tracked volumes can be used. Volume determination is then a simple procedure of counting up the visible voxels while using the appropriate voxel dimensions in the calculation.

Another 3-D application for voxel visualization is wellbore planning. The traditional 3-D surface rendering can include a wellbore track, as can 3-D volume rendering. However, wellbore planning requires going from the screen display into 3-D space, which can only be done with voxels that have three-dimensional extent. Voxel visualization allows interpretation and well drilling to be more closely tied together in an integrated environment.

Voxel visualization can also be applied to the problem of defining the velocity model for prestack depth migration. The velocity field can be displayed translucent with the velocity gathers superimposed. The gathers can be both the uncorrected and velocity-corrected gathers side-by-side. Picked imaging points can be saved as seed points for voxel tracking in the final migrated volume. This makes prestack velocity picking the start of prestack interpretation, which carries on into the final interpretation process

Price/Performance Considerations

The previous section outlines some of the ways voxel visualization brings the power of 3-D to the interpretation process. Voxel visualization has been readily available in the oil industry for less than five years. However, it is not currently widespread because of cost considerations. Hardware costs particularly have been high, requiring high-end Unix workstations. High cost has effectively put the advantages of voxel visualization out of reach for the average interpreter. But this need not be the case today.

Five years ago Unix workstations easily outperformed the personal computer for graphics display. However, the large demand for games on the PC has guaranteed that PC graphics would improve more rapidly than the graphics on workstations. Today PC graphics performance is comparable to or better than Unix workstations. Figure 4 shows recent test results comparing PC performance to a Unix workstation. The PC graphics performance is up to 2-3 times faster than a comparable workstation.

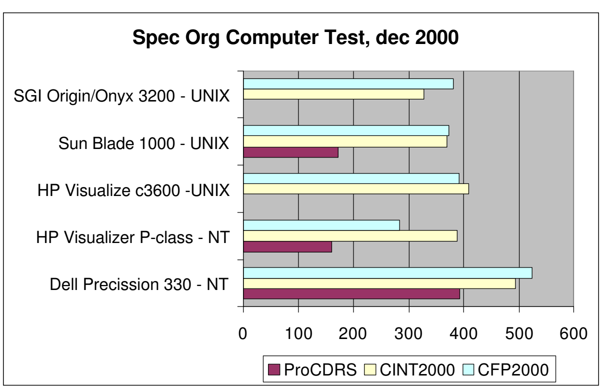


Figure 3: PC perfomance versus Unix workstation performance.

The PC hardware platform also is more cost effective. Pricing the PC hardware and Unix workstations (used in Figure 3) shows a definite price advantage for the PC. Therefore, on a dollar basis, the PC has a much higher price/performance than the Unix workstation. This is shown in Figure 4 with the SUN Blade 1000 assigned a value of one. Current PC configurations can show a price/performance advantage of ten or more over Unix workstations.

SUN Blade 1000	=	1X
HP Visualize P-class	=	2-3X
Dell Precision 330	=	7-8X

Figure 4: PC price/perfomance versus Unix workstation price/performance.

With the price/performance of the PC so much better than the Unix workstation, it is now possible to use voxel visualization on a PC with its much lower cost. Advances in computer programming allow voxel rendering in the CPU, further lowering costs. This progress has made possible inexpensive parallel processing on the PC platform yielding high-end voxel visualization at a fraction of the cost of Unix workstations. The result is a fully scalable visualization system based on inexpensive, easily supported PC hardware.

Today voxel visualization can be done on either a desktop or laptop PC with performance as good as or better than current Unix workstations, Figure 5. It can also be done on massively parallel PC-based systems suitable for visionariums. The ability to display and interpret in true three dimensional space is now readily available to the average interpreter.

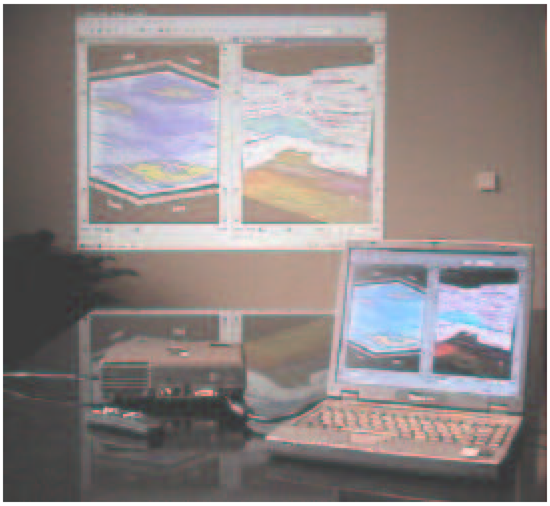


Figure 5: Voxel visualization on a laptop PC for the average interpreter.

Conclusions

Voxel visualization provides several 3-D interpretation advantages over traditional 3-D surface rendered displays. These include 3-D visualization of structure prior to interpretation, true 3-D tracking, volume tracking, volume calculation accurate as the data, and wellbore planning. Novel applications include applying it to prestack depth migration with the ability to capture interpretation information during velocity picking. Other 3-D uses will certainly emerge as voxel visualization becomes commonplace.

References

Dorn, G. A., Cole, M. J. and Tubman, K. M., 1995, Visualization in 3-D seismic interpretation: The Leading Edge, **14,** no. 10, 1045-1049.

Kao, J. C., Schneider, W. A. and Whitman, W. W., 1990, Automated interpolation of two-dimensional seismic grids into three-dimensional data volume: Geophysics, **55,** no. 04, 433-442.

Kidd, G. D., 1999, Fundamentals of 3-D seismic volume visualization: The Leading Edge, **18,** no. 6, 702, 704-709, 712.

Lees, J. A., 1999, Constructing faults from seed picks by voxel tracking: The Leading Edge, **18,** no. 3, 338,340.

Lieber, R. B., 1997, A proposed common scale for 3-D visualization and modeling: Annual Meeting Abstracts, Society Of Exploration Geophysicists, 570.

Nelson, H. R., Jr., Mastoris, S. and Huxohl, C., 1991, Visualization of map and seismic attributes: 61st Mtg. Eur. Assoc. Expl Geophys., Extended Abstracts, European Association Of Geophysical Exploration, 86-87.

Sengupta, M. K. and Peddy, C. P., 1995, AVO analysis for the high-impedance gas sands: Improved visualization, and rock-property estimation: Annual Meeting Abstracts, Society Of Exploration Geophysicists, 581-584.

Stark, T. J., 1995, Surface slices as a stratigraphic interpretation tool: Annual Meeting Abstracts, Society Of Exploration Geophysicists, 865-868.