

INTERACTIVE VISUALIZATION OF SUBSURFACE INFORMATION

Timo Ropinski, Klaus Hinrichs

Institut für Informatik, Westfälische Wilhelms-Universität Münster

ABSTRACT

In this paper we describe concepts, which support the interactive exploration of subsurface information extracted from seismic datasets. Since in general subsurface information is of volumetric nature, appropriate visualization techniques are needed to provide an insight view to special regions of interest. Usually clipping planes or surface extraction techniques are used for this purpose. We will present an approach, which allows the user to interactively change the visual representation for distinct regions of seismic datasets. Using this technique highlighting of regions of interest as well as clipping against volumetric regions can be realized. Volumetric clipping regions have the potential to assist the user when visually intruding into a 3D dataset by permitting an occlusion free view to inner regions of the dataset. During this process it is desirable to know where the current position is located relative to the whole dataset. We will introduce a 3D widget, which displays information concerning the location and orientation of the virtual camera during the exploration process.

1. INTRODUCTION

Geographic information systems (GIS) have been proven essential for the tasks of exploring and communicating spatiotemporal data and its associated information. Particularly the development of 2D GIS has advanced in the last decades, because there has been a great demand for these systems to support map-based planning tasks.

Since geographic data is inherently three-dimensional, a 2D representation is insufficient for many tasks, e.g. the visualization of terrain data. Therefore the idea of 3D GIS has been proposed [7]. Most 3D GIS adapt the concepts of polygonal representations for storing geographic information. One disadvantage of this form of representation is that geo-objects can only be represented by a polygonal approximation of their surface. This is sufficient as long as no information associated with volumetric subsurface data has to be represented. Since the visualization of subsurface information is essential in many application fields, e.g. locating oil or gas reservoirs

as well as earthquake prediction, appropriate forms of representation are needed.

For representing information associated with subsurface data volumetric techniques are adequate ([8], [18]), which have been already applied in medical visualization. Hence these techniques are highly developed. Especially for the use with large datasets, as found in geo-applications, many algorithms have been proposed, which allow interactive visualization on standard desktop computer systems.

In this paper we will propose techniques for enhancing interactive visualization and information retrieval of volumetric subsurface datasets.

One problem of visualizing information associated with volumetric data is that usually no insight view can be achieved. Especially when navigating through a visual representation the view of the camera will always be occluded. To avoid this problem we propose the application of 3D magic lenses ([13], [16]) to enhance visualization of subsurface information. By using 3D magic lenses it is possible to assign different visual representations to special regions of interest. Thus an occlusion-free view can be obtained by removing distracting information when intruding into the visual representation.

While navigating through the dataset it is desirable to know where the current view is located and how it is oriented in relation to the whole volume dataset. Therefore, we propose a 3D widget, called orientation widget, which shows the current position and the orientation of the virtual camera in relation to the 3D dataset.

This paper is divided into five sections. The next section will describe some related work concerning interactive visualization and exploration of subsurface information. In Section 3 we will describe some visualization techniques and give some examples, explaining how interactive visualization of subsurface information can benefit from the use of these techniques. Section 4 introduces the orientation widget, which assists the user while navigating through the visual representation. The paper concludes in Section 5 by giving an overview of the proposed concepts and future work.

2. RELATED WORK

Visualization techniques for representing volumetric phenomena have advanced in the past years. Most of the proposed techniques have been developed for medical applications. A good overview of visualization techniques used to visualize seismic datasets can be found in [9].

There are two kinds of geological data from which subsurface information can be retrieved: well data and seismic data [8]. Well data is directly extracted from a well, which contains precise information of the geological structure of the subsurface. Since it is very expensive to acquire well datasets, numerous techniques have been developed, which focus on information retrieval from seismic datasets [5]. Seismic data is acquired indirectly. Explosives are used to emit sound waves, which are propagated through the subsurface. The reflections of those sound waves are measured with special reflectors located at the surface. Thus geological layers, which reflect the sound waves, can be identified based on the time elapsed before a signal is received. The resulting information is encoded in a 3D volume composed of discrete samples, each representing the amplitude reflected beneath the surface. This 3D volume is usually stored as a SEG-Y dataset [14]. The acquired amplitudes can be visualized using different transfer functions to display the geological structure [15].

Since volumetric datasets, either of medical or geological type, are usually very large, many techniques have been proposed, which allow an interactive visualization of such datasets ([6], [12]). We do not refer directly to these kinds of techniques, but are focusing on visualization techniques, which reveal information usually hidden in the interior of seismic datasets. The described techniques are an extension to direct seismic visualization, i.e. visualizing without first performing an edge or surface detection.

Two different visualization techniques are widely accepted for displaying information encoded in seismic volume datasets. The slicing approach displays a sequence of successive slices of the 3D dataset [3]. By displaying these slices rapidly the user can construct a mental image containing the desired information. This concept can be implemented easily by applying clipping planes, which are supported by current graphics hardware. A disadvantage of this method is that it only leads to satisfying results when an image sequence is displayed. Hence showing a single image representing a single slice is less meaningful.

Wolfe and Liu [18] have proposed another approach, which allows to reveal information hidden inside the dataset. Their technique uses a threshold and visualizes only certain data values present in the volume by assigning transparency. Thus it is possible to reduce the information

being visualized and therefore to give the user a better overview of the whole dataset.

Both visualization concepts have been widely adapted in interactive visualization tools specialized for the interpretation of seismic volume datasets ([10], [11]).

Fröhlich et al. [4] have described the application of volumetric lenses to seismic datasets. They claim that the exploration of seismic datasets can benefit from the use of volumetric lenses, which can be freely positioned in the scene. However, they support only cuboid lenses. Since the surfaces of the cuboid represent orthogonal planes, this results in a visualization similar to the slicing approach when multiple slices are used. Therefore their technique can be considered as a generalization of the slicing approach, which uses more than one clipping plane.

3. VISUALIZATION TECHNIQUES

Due to the volumetric nature of seismic datasets it is difficult to provide mechanisms for enabling a sophisticated interactive visualization, since several problems may arise. First of all, because most commonly used graphics systems just provide a 2D projection of the data to be visualized, information hidden deep inside the dataset is usually occluded by the regions, which lie at the border. Currently most visualization techniques applied to seismic datasets use the slicing approach to deal with the occlusion problem, i.e. one or multiple clipping planes are used to reveal occluded regions of the dataset, and the user has to construct a mental image representing the information of interest.

3D magic lenses have the potential to deal with this shortcoming. By using the magic lens metaphor occlusions can be avoided without losing information, which is needed to visualize the 3D structure of a seismic dataset.

Interactive visualization of 3D magic lenses gives the user the ability to explore seismic datasets by providing full contextual information. The mechanism is based on the 2D magic lens metaphor, which has been introduced by Bier et al. [1]. In their work they describe Toolglass™ widgets as new interface tools for information visualization that can appear, as though on a transparent sheet of glass, between an application and a traditional cursor. Toolglass widgets can be positioned with one hand while the other positions the cursor. They may incorporate visual filters, known as magic lenses, which modify the visual representation of application objects, enhance data of interest or suppress distracting information in the region of interest determined by the shape of the lens. 3D Magic lenses [16] are represented by glass volumes associated with a visual representation, which is applied to the geometry intersecting the lens volume.

Image-based rendering of convex magic lenses [13] can be performed with three-pass rendering. The first pass renders the parts of the virtual environment which lie

behind and next to the lens volume (a), the second pass renders those parts which intersect the lens volume (b), and the third pass renders everything in front of the lens volume (c) (see Figure 1).

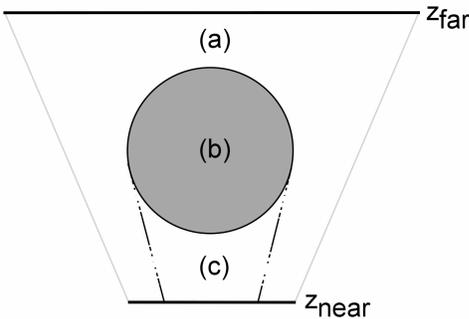


Figure 1 Rendering of the three regions (a-c).

During each pass a second depth test is used to check if the current sample of the volume dataset lies behind, inside or in front of the lens volume. Therefore the depth component of the current sample is compared with the corresponding depth value stored in a depth texture, which is part of the data structure representing the depth structure of the lens volume. Because the lens volume has to be convex, two different depth textures are used: dt_0 stores the depth values representing the front boundary of the lens volume and dt_1 stores the depth values representing the back boundary. Each depth texture is obtained during an extra rendering pass where just the lens geometry is rendered, before rendering the volume dataset. After rendering the lens geometry the resulting depth buffer information of each pass is stored in the corresponding depth texture. Thus the viewport-sized depth textures dt_0 and dt_1 obtained in this way store the depth structure of the lens volume. The depth textures are used in three subsequent rendering passes to determine whether a sample intersects one of the three described regions (a-c):

1. use depth texture dt_1 with its additional depth test set to greater (render region a)
2. use depth texture dt_1 with its additional depth test set to less and use depth texture dt_0 with its additional depth test set to greater (render region b)
3. use depth texture dt_0 with its additional depth test set to less (render region c)

We have adapted the described concepts to volume rendering techniques using texture mapping hardware [2] which we use to visualize seismic datasets. These volume rendering techniques visualize a volumetric dataset which

is represented as either several 2D textures or one 3D texture by rendering a stack of screen aligned polygons onto which the volume dataset is projected using appropriate texture coordinates. When using texture based volume rendering techniques it is important to ensure that the slices are rendered in a predefined order, either back to front or front to back, since the slices are blended based on the transparency values stored in the volume dataset to achieve a correct visualization.

To assign a special visual representation when rendering the parts of the volume dataset intersecting the lens volume, this representation must be applied during rendering pass 2 described above. The visual representation can be altered by using the methods provided by the magic lens interface (see Figure 2). We have designed this interface as an extension to a scene graph based 3D graphics system to allow easy use of the described concepts. There are three different groups of methods usable for altering the visual representation inside the lens volume: methods for accessing general representations, methods for accessing representations for particular shapes, and methods needed to exclude shapes from rendering.

```
interface MagicLens {
    void addRepresentationAttr(Attribute attr);
    void removeRepresentationAttr(Attribute attr);
    AttributeComposite getRepresentation();

    void addDSRepresentationAttr(string name,
                                Attribute attr);
    void removeDSRepresentationAttr(string name,
                                    Attribute attr);
    AttributeComposite getDSRepresentation
        (string name);

    void addInvalidDS(string name);
    void removeInvalidDS(string name);
}
```

Figure 2 Magic lens interface.

With the methods `addRepresentationAttr()` and `removeRepresentationAttr()` the visual representation used inside the lens volume can be specified. All those attributes are bundled in an object of the class `AttributeComposite`, which is returned by the method `getRepresentation()`. If several overlapping datasets have to be visualized, similar methods can be used to assign visual representations to these datasets. The datasets have to be identifiable by an object name. The object name together with the attribute representing the representation is passed to the methods `addDSRepresentationAttr()` and

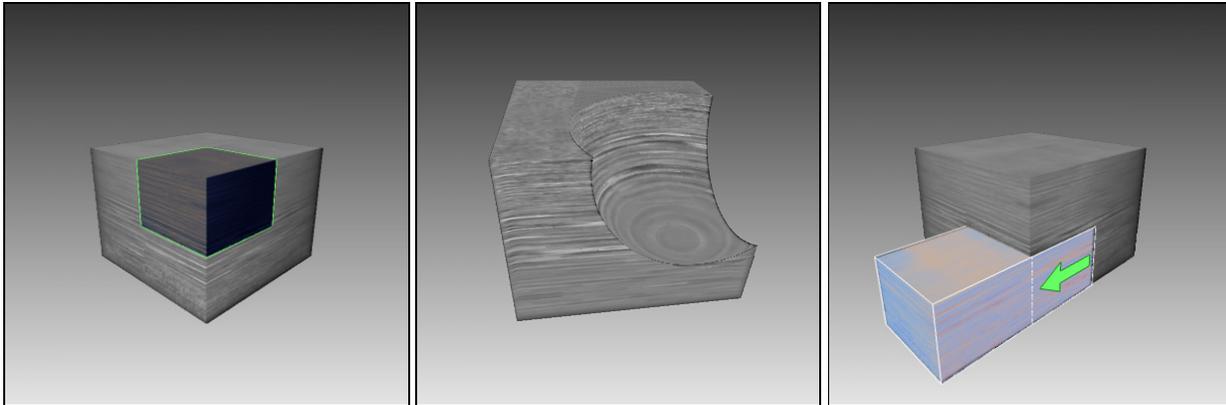


Figure 3 Magic lenses with different visual representations applied to a seismic volume dataset. Cuboid lens using different transfer function in combination with edge enhancement (left), spherical clipping lens with enhanced silhouette edges (middle), and interactive drawer metaphor with a transfer function highlighting extremely dense regions (right).

`removeDSRepresentationAttr()`. The representation used for a particular dataset can be queried by calling the method `getDSRepresentation()` with the corresponding object name. Furthermore a lens can be configured to exclude particular datasets from rendering by calling the methods `addInvalidDS()` and `removeInvalidDS()`.

By giving the user the ability to change the region of interest interactively, magic lenses have the potential to support users in the task of interactive information visualization by allowing 3D exploration. We have classified two types of 3D magic lenses: camera lenses and scene lenses. Camera lenses are positioned relative to the virtual camera, whereas scene lenses can be positioned anywhere in the virtual environment.

Camera lenses can assist the user while intruding in dense datasets like seismic datasets by removing or translucently rendering parts, which occlude the view of the camera. Thus it is possible to navigate through virtual environments containing dense data without removing context information. To assist the user in exploring seismic datasets we have implemented three visual representations bound to camera lenses: (1) semi-transparent rendering, (2) removing of partial or complete data intersecting the lens volume, and (3) highlighting data inside the lens volume by applying extra light sources. These representations can be combined in a single lens.

In contrast to camera lenses, scene lenses are particularly suitable for accentuating special regions of interest interactively. By using the magic lens metaphor it is assured that the context information does not change, while it is possible to alter the visual representation inside the lens volume. In order to assist the user in exploring the

data different visual representations can be applied inside lenses. One example is changing the transfer function used when rendering the dataset. Although in most cases the same transfer function is used for the whole dataset, it may be desirable to use different transfer functions in different parts of the dataset, e.g. to identify special regions of interest [15]. Thus it is possible to reveal structures within the data without changing the context, since in the parts not intersecting the lens volume the original transfer function is retained (see Figure 3 (left)). Magic lenses can also be used as clipping volumes as shown in Figure 3 (middle), where a spherical clipping lens is used to reveal information hidden inside the dataset. For an easier identification of the clipping region its silhouette edges are enhanced. Figure 3 (right) shows the usage of the interactive *drawer metaphor*. With this metaphor the user can interactively cut differently shaped subvolumes out of the whole dataset. These volumes are visualized using the representation assigned to the magic lens. To better display the information hidden inside these subvolumes, they are duplicated and pulled out in a drawer like manner to be positioned outside the dataset. Thus the user can explore even the sides otherwise occluded. The green arrow shown in Figure 3 (right) indicates in which direction the subvolume has been pulled out.

Furthermore it is possible to combine the well known techniques described in Section 2 with the magic lens metaphor. For example when only displaying certain data values present in the volume [18], this can be done exclusively inside the lens volume by applying threshold during rendering (see Figure 8 (top right)). In Figure 4 a magic lens is used to combine volumetric clipping with the slicing approach. The seismic dataset is volumetrically clipped by a sphere shaped lens, but to give a better

overview, two slices representing the right and the bottom outer face of the dataset are visualized.

Alternatively the interactions needed for the slicing approach can be adopted by the magic lens technique. Because the size and the shape of the lens volume can be altered interactively, a resizable volumetric clipping region can be realized [17]. When resizing a volumetric clipping region, the information shown on the border changes with the size of the volume. Similar to moving a single clipping plane, the resulting information can be used to reveal information hidden within the dataset.

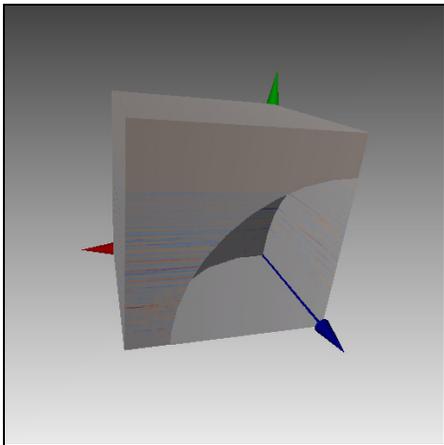


Figure 4 Clipping lens combined with slice.

4. NAVIGATION ASPECTS

The use of the 3D magic lens metaphor as described in Section 3 gives the user the ability to intrude into the dataset. Thus a fly-through metaphor can be used for navigation.

In aboveground navigation the user usually has a set of landmarks he can use for position sensing. In contrast the user does not have any such static reference points when navigating beneath the surface. Therefore a tool is needed for determining the position and the orientation while navigating through dense datasets.

We propose to use an orientation widget to display the current orientation and position relative to the dataset. The widget is composed of a bounding volume, representing the boundaries of the seismic volume dataset, and an arrow displaying the current orientation and position of the virtual camera (see Figure 5).

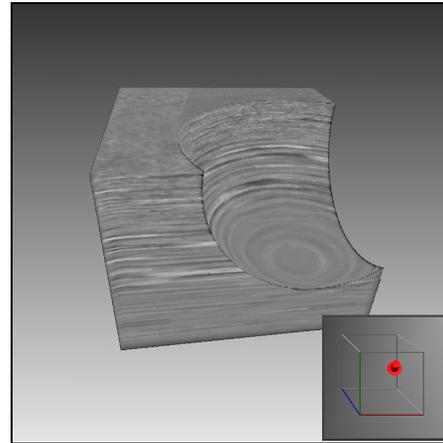


Figure 5 The orientation widget showing the current position of the camera.

The position as well as the orientation of the arrow is derived from the current orientation matrix determined by the virtual camera. Although the concepts described in this paper can be easily realized with different kinds of graphics systems, we are going to explain our implementation, which is based on a high-level rendering system using a scene graph to define the scene content.

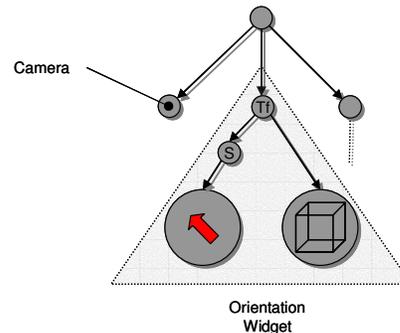


Figure 6 Scene graph with orientation widget.

The structure of the orientation widget as well as its integration in the scene graph is shown in Figure 6. The orientation widget consists of two primitives, the bounding box and the arrow. The orientation of the arrow depends on the virtual camera. Both elements are transformed by the scaling transformation S . This scaling transformation defines the size of the widget relative to the size of the seismic volume.

It directly influences the size of the orientation widget in screen space, since the front face of the bounding box is always parallel to the image plane and has a fixed distance to the camera. The matrix for this transformation can be either assigned automatically based on screen coordinates

and the dimensions of the seismic dataset, or it can be set by the user. T_f denotes the camera dependent transformation used to orient the arrow. It is updated during each redisplay, when the scene graph is evaluated. To perform this update, the transform needs to have a reference to the virtual camera to read its current orientation matrix.

Using the orientation widget the user can visually intrude into the dataset without losing orientation (see Figure 7). Hence most navigation techniques, which have been established in 3D virtual environments can be used subsurface, e.g. fly-through metaphors which may be combined with exponential acceleration algorithms.

The primitives used for the orientation widget, i.e. the bounding box and the arrow, can be altered for different purposes. For example to enable a more accurate prediction of the position inside the dataset, auxiliary lines can be visualized, each as long as the corresponding dimension of the bounding box. By using these kind of auxiliary lines the starting point of each line at a side of the bounding box gives the user an indication of the camera's orientation and location inside the lens volume. Another extension of the orientation widget could allow interactive positioning of the virtual camera by providing handles integrated into the widget.

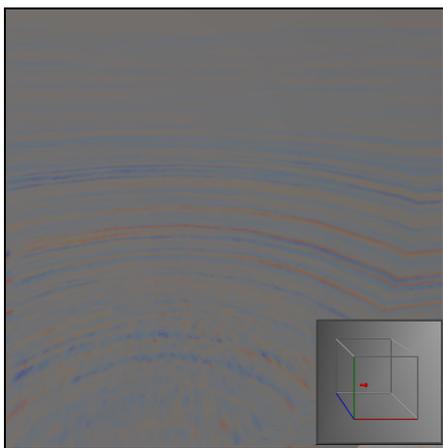


Figure 7 Orientation widget during intrusion.

Furthermore the orientation widget can be positioned freely by applying appropriate transformations. Thus it is possible to provide a zoomed view of the widget by scaling and centering it on the screen. This visualization of the orientation widget may be helpful for a more accurate positioning.

5. CONCLUSION AND FUTURE WORK

In this paper we have presented techniques for enhancing interactive visualization of subsurface information extracted from seismic datasets. All presented visualization techniques perform on off-the-shelf graphics hardware at interactive frame rates. We have proposed the use of 3D magic lenses to allow intruding into seismic datasets without occluding the view of the camera. Using the 3D magic lens metaphor different visual representations can be applied inside the lens volume. Those representations can be used to highlight regions of interest or filter the kind of information visualized for the desired region. Furthermore different transfer functions as well as threshold based transparency techniques can be applied inside the lens volume to improve information visualization.

For orientation purposes a 3D orientation widget has been introduced, which displays the current position and the orientation of the virtual camera. This widget gives the user the ability to plan navigation tasks by providing an immediate feedback.

The proposed techniques can be combined with existing information visualization techniques used to display information associated with volumetric seismic and well datasets. Since volumetric data is used in many geo-related fields as well as in medical visualization, there is a bright spectrum of applications, which can benefit from the described techniques. For example the oil and petroleum industry can apply the concepts for well planning. Furthermore archeological applications may benefit from the described concepts when analyzing strata.

Currently we are working on an easy to use library, which implements the described concepts. Since the proposed visualization techniques can be extended easily, i.e. new visual representations can be defined to be applied inside the lens volume, geo-scientists can develop their own lenses for their preferred graphics system using the library.

6. REFERENCES

- [1] E.A. Bier, M.C. Stone, K. Pier, W. Buxton, and T. DeRose, "Toolglass and Magic Lenses: The See-Through Interface", In Proceedings of SIGGRAPH'93, ACM Press, pp. 73–80, 1993.
- [2] B. Cabral, N. Cam, and J. Foran, "Accelerated Volume Rendering and Tomographic Reconstruction using Texture Mapping Hardware", In Proceedings of Symposium on Volume Visualization, ACM Press, pp. 91–98, 1994.
- [3] M.P. Curtis, A.C. Gerhardstein, and R.E. Howard, "Interpretation of Large 3-D Data Volumes", Paper 510.5, Expanded Abstracts of the Society of Exploration Geophysicists 56th Annual Meeting, pp. 497-499, 1986.

- [4] B. Fröhlich, S. Barrass, B. Zehner, J. Plate, and M. Göbel, "Exploring Geo-Scientific Data in Virtual Environments", *Proceedings of IEEE Visualization*, pp. 169-174, 1999.
- [5] D. Gao, "Volume Texture Extraction for 3D Seismic Visualization and Interpretation", *In Geophysics*, Vol. 68, No. 4, pp. 1294-1302, 2003.
- [6] S. Guthe, M. Wand, J. Gonser, and W. Straßer, "Interactive rendering of large volume data sets", *In Proceedings of IEEE Visualization*, pp. 53-59, 2002.
- [7] D. Koller, P. Lindstrom, W. Ribarsky, L.F. Hodges, N. Faust, and G. Turner, "Virtual GIS: A Real-Time 3D Geographic Information System", *In Proceedings of IEEE Visualization*, pp. 94-100, 2003.
- [8] L.A. Lima, and R. Bastos, "Seismic Data Volume Rendering", Technical Report TR 98-004, Department of Computer Science, University of North Carolina at Chapel Hill, 1998.
- [9] C. Ma, and J. Rokne, "3D Seismic Volume Visualization", *In Integrated Image and Graphics Technologies*, Kluwer Academic Publishers, pp. 241-262, 2004.
- [10] Magic Earth: GeoProbe. (<http://www.magic-earth.com/geo.asp>).
- [11] Paradigm: Voxelgeo. (<http://www.paradigmgeo.com/products/voxelgeo.php>).
- [12] S. Roettger, S. Guthe, D. Weiskopf, and T. Ertl, "Smart Hardware-Accelerated Volume Rendering", *In Proceedings of Symposium on Visualization*, IEEE, pp. 231-238, 2003.
- [13] T. Ropinski, and K. Hinrichs, "Real-Time Rendering of 3D Magic Lenses having arbitrary convex Shapes", *International Winter School of Computer Graphics*, Journal of WSCG, 12(2), pp. 379-386, 2004.
- [14] SEG Technical Standards Committee, "SEG Y rev 1 Data Exchange Format". Society of Exploration Geophysicist, Editors Michael W. Norris and Alan K. Faichney, 2002.
- [15] T.M. Sheffield, D. Meyer, J. Lees, G. Kahle, B. Payne, and M. J. Zeitlin, "Geovolume visualization interpretation: Color in 3-D volumes", *The Leading Edge*, pp. 668-674, 1999.
- [16] J. Viega, M. Conway, G. Williams, and R. Pausch, "3D Magic Lenses". *In Proceedings of UIST*, pp. 51-58, 1996.
- [17] D. Weiskopf, K. Engel, and T. Ertl, "Interactive Clipping Techniques for Texture-Based Volume Visualization and Volume Shading", *In Transactions on Visualization and Computer Graphics*, Vol. 9, No. 3, pp. 298-312, 2003.
- [18] R.H. Wolfe, and C.N. Liu, "Interactive Visualization of 3D Seismic Data: A Volumetric Method", *In Computer Graphics and Applications*, Volume 8, Issue 4, pp. 24-30, 1988.

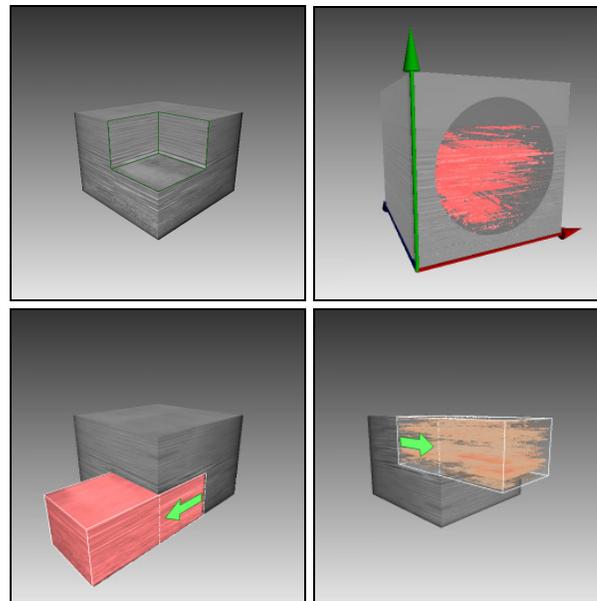


Figure 8 Different types of magic lenses. Cuboid lens with enhanced clipping plane edges (top left), a spherical lens showing different transfer function and threshold (top right), visualization of the drawer metaphor using a *highlight* transfer function (bottom left), and the drawer metaphor combined with threshold (bottom right).